

UNIVERZITET U NOVOM SADU
PRIRODNO-MATEMATIČKI FAKULTET
DEPARTMAN ZA
MATEMATIKU I INFORMATIKU



Milinko Mandić

Razvoj platforme za standardizaciju obrazovanja nastavnika informatike

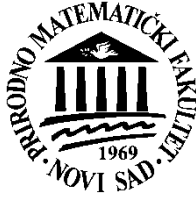
– doktorska disertacija –

Mentori:

Prof. dr Mirjana Ivanović

Prof. dr Zora Konjović

Novi Sad, 2015.



UNIVERZITET U NOVOM SADU
PRIRODNO-MATEMATIČKI FAKULTET
DEPARTMAN ZA
MATEMATIKU I INFORMATIKU



Milinko Mandić

Razvoj platforme za standardizaciju obrazovanja nastavnika informatike

– doktorska disertacija –

Mentori:

Prof. dr Mirjana Ivanović

Prof. dr Zora Konjović

Novi Sad, 2015.

Predgovor

U disertaciji je razvijena softverska platforma namenjena standardizaciji obrazovanja nastavnika informatike. Softverska platforma treba da olakša definisanje kompetencija nastavnika informatike u skladu sa aktuelnim srednjoškolskim informatičkim standardima i savremenim kurikulumima za nastavnike informatike. Na osnovu istraživanja stavova nastavnika informatike, preporuka međunarodnih organizacija i analize reprezentativnih nastavničkih kurikuluma kreiran je računarski model referentnog nastavničkog kurikuluma. Računarski model srednjoškolskog informatičkog kurikuluma zasnovan je na ACM K12 standardu, dok je model konkretnog nastavničkog kurikuluma kreiran na osnovu izabranog kurikuluma iz Republike Srbije. Razvijena softverska platforma je verifikovana utvrđivanjem usaglašenosti kreiranih računarskih modela. Ona je zasnovana na tehnologijama Semantičkog Weba i implementirana u programskom jeziku Java.

Disertacija sadrži sledeća poglavlja:

1. Uvod
2. Istraživanje kurikuluma srednjoškolskih nastavnika informatike
3. Istraživanje stavova nastavnika informatike o kurikulumu
4. Ontološki model kurikuluma informatike
5. Platforma za sinhronizaciju informatičkih kurikuluma
6. Zaključak

U **prvom**, odnosno uvodnom poglavlju opisani su sadržaj i tema rada, definisani su predmet i cilj istraživanja, kao i postavljene hipoteze i postignuti rezultati istraživanja. Na kraju poglavlja definisani su pojmovi informatike, računarstva i računarskih nauka u kontekstu srednjoškolskih i nastavničkih kurikuluma.

Pregled aktuelnog stanja u sferi obrazovanja nastavnika informatike dat je u **drugom poglavlju**. Opisane su potrebe za standardizacijom kurikuluma za nastavnike informatike i preporuke relevantnih međunarodnih organizacija o modalitetima obrazovanja nastavnika informatike. Takođe je dat prikaz generalnih oblasti koje svaki kurikulum za nastavnike informatike treba da sadrži. Detaljno su prikazani aktuelni predlozi standarda za srednjoškolsko informatičko obrazovanje. Takođe, izabrani su i analizirani reprezentativni svetski kurikulumi za nastavnike informatike, kao i kurikulumi iz Republike Srbije. Odabrani kurikulumi su upoređeni sa srednjoškolskim informatičkim standardom.

Rezultati istraživanja stavova nastavnika informatike u AP Vojvodina o preporučenim sadržajima nastavničkih informatičkih kurikuluma prikazani su u **trećem** poglavlju. Istraženi su stavovi nastavnika o zastupljenosti, značaju i poznavanju izabranih tema i oblasti, dobijenih

analizom kurikuluma i preporuka, kao i o modalitetu obrazovanja nastavnika informatike. Takođe, istraživanjem je utvrđeno akademsko i dodatno obrazovanje ispitanih nastavnika.

U **četvrtom** poglavlju opisan je ontološki model kurikuluma zasnovan na IEEE RCD standardu za definisanje kompetencija. Najpre je definisan srednjoškolski informatički ontološki model na osnovu aktuelnog ACM K12 CS predloga kurikuluma. Nakon toga, kreiran je model referentnog nastavničkog kurikuluma zasnovan na analiziranim kurikulumima i preporukama. Takođe, kreiran je i model nastavničkog kurikuluma definisan na osnovu jednog odabranog kurikuluma za nastavnike informatike iz Republike Srbije.

U **petom** poglavlju opisana je specifikacija (funkcije i softverska arhitektura) i implementacija softvera za utvrđivanje usaglašenosti kurikuluma. Prilikom implementacije korišćen je skup biblioteka otvorenog koda pisanih u Java programskom jeziku. Softverska aplikacija zasnovana je na algoritmima ontološkog uparivanja, koji su detaljno opisani u ovom poglavlju. Na kraju poglavlja prikazana je verifikacija softverskog modela. Primenom razvijene softverske aplikacije analizirani su rezultati uparivanja svih kreiranih ontoloških modela kurikuluma.

Na kraju disertacije dat je **zaključak** rada i navedeni su mogući pravci daljeg istraživanja.

Želim da izrazim najveću zahvalnost svojim mentorkama prof. **dr Mirjani Ivanović** i prof. **dr Zori Konjović**.

Zahvaljujem se prof. **dr Mirjani Ivanović** na ukazanom poverenju, dragocenim savetima, posvećenom vremenu, velikom strpljenju i nesebičnoj pomoći tokom doktorskih studija kao i tokom pisanja i izrade doktorske disertacije.

Veliku zahvalnost dugujem prof. **dr Zori Konjović** na korisnim idejama, brojnim ljudskim i profesionalnim savetima, i, uopšte, nemerljivoj pomoći i razumevanju tokom svih godina naše saradnje.

Takođe se zahvaljujem članovima komisije na interesovanju za moj rad, kao i na korisnim sugestijama i preporukama.

Neizmernu zahvalnost za ljubav i podršku koju su mi pružili dugujem mojim roditeljima, **Dragici i Milanu**, sestri **Milani** i njenoj porodici, naročito velikoj radosti, maloj **Petri**. Hvala svim mojim prijateljima koji su bili uz mene tokom rada na disertaciji, a ogromno hvala za sve mojim kumovima **Emiliji** i **Milošu Svirčev**.

Sadržaj

1. UVOD	12
1.1. Sadržaj i tema rada	14
1.2. Definicije najvažnijih pojmova (računarskih nauka, računarstva, informatike) u kontekstu srednjoškolskih i nastavničkih kurikuluma	16
2. ISTRAŽIVANJE KURIKULUMA SREDNJOŠKOLSKIH NASTAVNIKA INFORMATIKE.....	18
2.1. Postojeći standardi u srednjoškolskom informatičkom obrazovanju	18
2.1.1. ACM model K12 kurikuluma računarskih nauka	19
2.2. Aktuelno stanje u oblasti obrazovanja nastavnika informatike...	19
2.3. Potreba za standardizacijom.....	21
2.4. Preporuke međunarodnih organizacija.....	22
2.5. Rezultati naučnih radova i iskustva iz različitih zemalja	24
2.6. Klasifikacija oblasti koje svaki program za pripremu nastavnika informatike treba da sadrži	27
2.7. Analiza kurikuluma za nastavnike informatike.....	29
2.7.1. Usaglašenost domenskih znanja u nastavničkim kurikuluma sa ACM K12 modelom.....	30
2.7.1.1. Analiza svetskih kurikuluma	32
2.7.1.2. Analiza kurikuluma iz Republike Srbije	39
2.7.2. Usaglašenost kurikuluma sa nedomenskim znanjima	41
2.7.3. Zaključna razmatranja	43
3. ISTRAŽIVANJE STAVOVA NASTAVNIKA INFORMATIKE O KURIKULUMU	47
3.1. Cilj istraživanja.....	47
3.2. Uzorak	48
3.3. Istraživački instrumenti	48
3.3.1. Specifikacija upitnika	49
3.4. Rezultati istraživanja stavova nastavnika.....	50
3.4.1. Akademsko obrazovanje	50
3.4.2. Nastavničko iskustvo.....	52
3.4.3. Ocena osposobljenosti nastavnika u domenskoj (informatičkoj) oblasti i pedagoško-metodičkom korpusu	52
3.4.4. Permanentno obrazovanje	55
3.4.5. Stavovi nastavnika o značaju domenskog i pedagoško-metodičkog korpusa u kurikulumu obrazovanja nastavnika informatike	57

3.4.6. Stavovi nastavnika o zastupljenosti domenskih znanja u kurikulumu srednjoškolskog obrazovanja.....	59
3.4.7. Stavovi o modalitetima obrazovanja nastavnika informatike	61
3.4.8. Stavovi nastavnika o standardizaciji, dostupnosti i usaglašenosti kurikuluma	62
3.4.8.1. Standardizacija kurikuluma.....	62
3.4.8.2. Poznavanje kurikuluma	64
3.4.8.3. Stavovi nastavnika o dostupnosti kurikuluma.....	65
3.4.8.4. Stavovi nastavnika o usaglašenosti kurikuluma.....	67
3.4.9. Korisnost softverskih alata za realizovanje nastave	70
3.5. Zaključci i implikacije istraživanja	71
4. ONTOLOŠKI MODEL KURIKULUMA INFORMATIKE ...	73
4.1. Pojam ontologije.....	73
4.2. Pregled literature iz oblasti primene ontologija u obrazovanju...	75
4.2.1. Primena ontologija za reprezentaciju kurikuluma.....	77
4.3. Ontološki model kurikuluma.....	79
4.3.1. Kompetencija kao osnovna klasa ontološkog modela.....	79
4.3.2. IEEE RCD standard.....	80
4.3.3. Ontološka reprezentacija kompetencija zasnovana na IEEE RCD standardu	82
4.3.4. WordNet leksička baza podataka za engleski jezik.....	87
4.3.5. Ontološki model srednjoškolskog informatičkog kurikuluma	88
4.3.6. Ontološki model referentnog kurikuluma za nastavnike informatike	95
4.3.7. Ontološki model kurikuluma za nastavnike informatike iz Republike Srbije	98
5. PLATFORMA ZA SINHRONIZACIJU INFORMATIČKIH KURIKULUMA	104
5.1. Pregled literature u oblasti usaglašavanja ontologija	104
5.1.1. Ontološko uparivanje	104
5.1.2. Tehnike uparivanja ontologija.....	105
5.1.2.1. Tehnika zasnovana na sličnosti naziva (Terminološka sličnost)	105
5.1.2.1.1. Metode zasnovane na sličnosti stringova	105
5.1.2.1.2. Metode zasnovane na jeziku.....	107
5.1.2.2. Strukturne metode	110
5.1.2.2.1. Interna struktura	110

5.1.2.2.2. Relaciona struktura.....	110
5.1.2.3. Ekstenzione tehnike.....	111
5.1.2.4. Semantičke metode.....	111
5.1.3. Strategije za kombinovanje različitih tehnika	112
5.1.4. Učešće korisnika u usaglašavanju ontologija.....	113
5.1.5. Evaluacija rezultata uparivanja	114
5.2. Opis izabranih tehnika za usaglašavanje ontoloških reprezentacija kurikuluma.....	114
5.2.1. Određivanje “uparenih entiteta” iz matrice sličnosti.....	116
5.2.2. Sličnost zasnovana na poređenju naziva (Terminološka sličnost)	118
5.2.3. Taksonomijska strukturalna sličnost	119
5.2.3.1. Izračunavanje sličnosti roditeljskih klasa.....	119
5.2.3.2. Izračunavanje sličnosti „list“ klasa uparenih roditelja	120
5.2.3.3. Izračunavanje sličnosti „list“ klasa neuparenih roditelja	121
5.2.4. Relaciona sličnost.....	122
5.2.5. Superklasa - potklasa relacije između klasa dve ontologije	124
5.2.6. Izmene rezultata uparivanja ontologija od strane korisnika sistema	124
5.2.7. Način prikazivanja rezultata	125
5.3. Implementacija softverske platforme za sinhronizaciju informatičkih kurikuluma.....	127
5.3.1. Specifikacija zahteva.....	127
5.3.2. Arhitektura sistema.....	131
5.3.2.1. Konvertor ontologija	131
5.3.2.2. Sistem za uparivanje.....	132
5.3.2.3. Eksport rezultata.....	132
5.3.2.4. Grafički korisnički interfejs.....	132
5.4. Evaluacija sistema za usaglašavanje nastavničkog referentnog i srednjoškolskog informatičkog kurikuluma.....	134
5.4.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa.....	134
5.4.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja.....	136
5.4.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja	137
5.4.4. Primena algoritma za izračunavanje relacione sličnosti.....	138
5.4.5. Primena 1:N algoritma	139
5.4.6. Analiza rezultata.....	140

5.5. Evaluacija sistema za usaglašavanje odabranog nastavničkog kurikuluma iz Republike Srbije i srednjoškolskog informatičkog kurikuluma.....	141
5.5.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa.....	142
5.5.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja.....	144
5.5.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja.....	146
5.5.4. Primena algoritma za izračunavanje relacije sličnosti.....	148
5.5.5. Primena 1:N algoritma.....	151
5.5.6. Analiza rezultata.....	152
5.6. Evaluacija sistema za usaglašavanje referentnog nastavničkog kurikuluma i odabranog nastavničkog kurikuluma iz Republike Srbije.....	153
5.6.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa.....	154
5.6.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja.....	157
5.6.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja.....	159
5.6.4. Primena algoritma za izračunavanje relacije sličnosti.....	161
5.6.5. Primena 1:N algoritma.....	163
5.6.6. Analiza rezultata.....	168
5.7. Diskusija i budući rad.....	170
6. ZAKLJUČAK.....	175

Lista tabela

Tabela 1. Srednjoškolski informatički sadržaji po ACM K12 CS modelu kurikuluma.....	32
Tabela 2. Kurikulumi u Nemačkoj.....	34
Tabela 3. MSC kurikulumi za nastavnike informatike u Austriji i Estoniji.....	36
Tabela 4. BSC i dipl. kurikulumi u SAD i Škotskoj.....	37
Tabela 5. BSC i dipl. kurikulumi u Turskoj i Austriji.....	38
Tabela 6. Izabrani kurikulumi za nastavnike informatike u Republici Srbiji.....	40
Tabela 7. Kursevi nedomenskih oblasti.....	42
Tabela 8. Struktura nastavnika u odnosu na akademsko obrazovanje i tip škole u kojoj izvode nastavu.....	51
Tabela 9. Znanja i veštine iz oblasti nastavne prakse stečena tokom studija.....	51
Tabela 10. Srednje vrednosti samoevaluacije podeljene po oblastima i tipu završenih studija.....	54
Tabela 11. Stavovi nastavnika o potrebi za permanentnim obrazovanjem.....	55
Tabela 12. Stavovi nastavnika o učestalosti potrebe za permanentnim obrazovanjem.....	55
Tabela 13. Način sticanja znanja nakon formalnog obrazovanja.....	56
Tabela 14. Značaj tematskih oblasti u kurikulumu obrazovanja nastavnika informatike.....	58
Tabela 15. Zastupljenost domenskih znanja u kurikulumu srednjoškolskog obrazovanja.....	60
Tabela 16. Najpodesniji koncept obrazovanja nastavnika.....	61
Tabela 17. Poznavanje procesa standardizacije kurikuluma.....	62
Tabela 18. Poznavanje ACM K12 CS standarda.....	62
Tabela 19. Potreba usaglašavanja informatičkog obrazovanja sa ACM K12 CS standardom.....	63
Tabela 20. Poznavanje ECDL programa.....	63
Tabela 21. Potreba usaglašavanja srednjoškolskog informatičkog obrazovanja sa ECDL sertifikacijom.....	63
Tabela 22. Poznavanje kurikuluma svih nivoa obrazovanja.....	64
Tabela 23. Dostupnost kurikuluma.....	65
Tabela 24. Potreba dostupnosti kurikuluma.....	66
Tabela 25. Mogućnost uticaja na kurikulum.....	66
Tabela 26. Korisnost softvera za rad sa kurikulumima.....	66
Tabela 27. Stepen struktuiranosti i sadržajnosti kurikuluma.....	67

Tabela 28. Aktuelnost kurikuluma	67
Tabela 29. Prilagođenost kurikuluma dinamici promena u oblasti	68
Tabela 30. Prilagođenost kurikuluma primeni novih tehnologija	68
Tabela 31. Usaglašenost kurikuluma različitih nivoa obrazovanja	68
Tabela 32. Korisnost softvera za proveru usaglašenosti kurikuluma različitih nivoa obrazovanja	69
Tabela 33. Način provere realizovanosti propisanog kurikuluma.....	69
Tabela 34. Korisnost softvera za proveru usaglašenosti realizovanog i propisanog kurikuluma.....	69
Tabela 35. Stavovi nastavnika o korisnosti softverskih alata za izvođenje nastave	70
Tabela 36. Revidirana Blumova taksonomija	86
Tabela 37. Primeri kategorisanja <i>Skills</i> potklasa srednjoškolskog modela kurikuluma.....	92
Tabela 38. Primeri kategorisanja <i>Skills</i> potklasa referentnog nastavničkog kurikuluma	98
Tabela 39. Ishodi i sadržaji predmeta 'Računarske mreže i komunikacije'	101

Lista slika

Slika 1. Semantički Web stek.....	74
Slika 2. Struktura klase <i>Competence</i>	80
Slika 3. IEEE RCD Data Model.....	81
Slika 4. Ontološka reprezentacija IEEE RCD standarda.....	83
Slika 5. Kardinalnost svojstava klase <i>Competence</i>	84
Slika 6. Potklase klase <i>Skills</i>	86
Slika 7. Tema 'Programming languages'.....	89
Slika 8. Potklase klase <i>Knowledge</i> nakon prve faze kreiranja srednjoškolskog ontološkog modela.....	90
Slika 9. Struktura klase <i>Problem_design</i>	91
Slika 10. Hijerarhijska struktura klase <i>Programming_Languages</i>	93
Slika 11. Hijerarhijska struktura klase <i>Careers_in_Computing</i>	94
Slika 12. Potklase klase <i>Knowledge</i> srednjoškolskog modela.....	94
Slika 13. Potklase klase <i>Knowledge</i> nastavničkog kurikuluma.....	96
Slika 14. Deo ontološke reprezentacije referentnog nastavničkog kurikuluma.....	97
Slika 15. Više klase informatičkih (domenskih) znanja nastavničkog kurikuluma iz Republike Srbije.....	100
Slika 16. Potklase klase <i>Computer_Networks_and_Communications</i>	102
Slika 17. Postupak uparivanja kurikuluma.....	115
Slika 18. Primer utvrđivanja najboljih parova iz matrice sličnosti ..	117
Slika 19. Primer opravdanosti primene Greedy selection metode za usaglašavanje ontologija.....	117
Slika 20. Grupe korisnika softverske platforme za sinhronizaciju informatičkih kurikuluma.....	127
Slika 21. Dijagram slučajeva korišćenja.....	127
Slika 23. Početni ekran softverske platforme.....	132
Slika 22. Arhitektura sistema za sinhronizaciju kurikuluma.....	132
Slika 24. Prikaz klase <i>Programming_Fundamentals</i>	133
Slika 25. Tabela prikaz rezultata uparivanja.....	133
Slika 26. Deo uparenih klasa nakon primene algoritma za utvrđivanje sličnosti roditeljskih klasa.....	135
Slika 27. Deo hijerarhijske strukture srednjoškolskog modela i referentnog nastavničkog modela.....	135
Slika 28. Primer uparenih klasa nakon primene druge faze taksonomijskog strukturnog algoritma.....	136
Slika 29. Uparene „list“ klase čiji roditelji nisu upareni.....	137

Slika 30. Deo uparenih veština srednjoškolskog i referentnog nastavničkog kurikuluma	139
Slika 31. Uparene klase u „Superclass“ relaciji	140
Slika 32. Deo uparenih roditeljskih klasa.....	142
Slika 33. Deo hijerarhijske strukture srednjoškolskog i nastavničkog modela	143
Slika 34. Parovi „list“ klasa uparenih roditelja	144
Slika 35. Deo ontološke strukture poređenih modela.....	145
Slika 36. Uparene „list“ klase neuparenih roditelja	147
Slika 37. Hijerarhijske strukture klasa “neuparenih roditelja”	147
Slika 38. Primeri uparenih <i>Skills</i> potklasa modela srednjoškolskog kurikuluma i odabranog nastavničkog kurikuluma	148
Slika 39. Primeri sva tri tipa relacija	151
Slika 40. Deo uparenih roditeljskih klasa modela nastavničkih kurikuluma.....	155
Slika 41. Deo strukture klasa modela nastavničkih kurikuluma	156
Slika 42. Uparene „list“ klase modela nastavničkih kurikuluma	157
Slika 43. Deo struktura klasa koje reprezentuju operativne sisteme i školsku praksu	158
Slika 44. Parovi „list“ klasa neuparenih roditelja modela nastavničkih kurikuluma.....	159
Slika 45. Međusobno neuparene klasne strukture nastavničkih modela	160
Slika 46. Međusobno neuparene klasne strukture modela nastavničkih kurikuluma.....	160
Slika 47. Uparene <i>Skills</i> potklase modela nastavničkih kurikuluma	161
Slika 48. Primeri parova klasa nastavničkih modela u 1:N relaciji..	164

Poglavlje I

1. Uvod

Brojna aktuelna literatura (Armoni, 2011; Ericson et al., 2008; Gal-Ezer & Stephenson, 2010; Hazzan et al., 2010; Ragonis et al., 2010; Wilson et al., 2010) ističe neophodnost dizajniranja savremenih nastavničkih kurikuluma iz oblasti računarskih nauka. U (Ericson et al., 2008; Wilson et al., 2010) se navodi da, nesrazmerno važnosti ove naučne discipline, u mnogim državama ne postoje programi za pripremu nastavnika računarskih nauka. Takođe, autori ističu da nisu jasno definisani uslovi (sertifikacija) koje je potrebno da ispune nastavnici iz obe oblasti. U (Stephenson et al., 2005) se naglašava da nedostatak konzistentnih i dostupnih informacija o uslovima sertifikacije čini skoro nemogućim određivanje kako se pripremiti za tu profesiju. Takođe, različite države realizuju različite srednjoškolske računarske kurikulume (čak i u okviru USA) i ne postoje internacionalni ili nacionalni standardi za pripremu nastavnika računarskih nauka (Ragonis et al., 2010). Kao posledica toga srednjoškolsku nastavu neretko realizuju nastavnici koji ne poseduju adekvatne kompetencije i formalno obrazovanje (Ericson et al., 2008).

Za rešenje takvog stanja, u (Ericson et al., 2008; Wilson et al., 2010) se preporučuje definisanje standarda za srednjoškolsko računarsko obrazovanje i obrazovanje nastavnika računarskih nauka, kao i standardizacija programa za sertifikaciju nastavnika. Ipak, aktuelna literatura koja se bavi tom tematikom najčešće sadrži samo nabrojane oblasti koje programi za pripremu nastavnika računarstva treba da sadrže. Tako, reference (Ericson et al., 2008; Gal-Ezer & Stephenson, 2010) prikazuju samo generalne preporuke prilagođene nastavnicima sa različitim predznanjem (sadašnjim nastavnicima sa ili bez informatičkog i/ili metodičkog znanja, budućim nastavnicima itd.); u (Hazzan et al., 2010) prikazane su osnovne komponente programa za pripremu nastavnika računarskih nauka. Slično, iako radovi (East et al., 2011; ISTE, 2011) predstavljaju važan doprinos u standardizaciji računarskih nastavničkih kurikuluma, ovi dokumenti sadrže samo nabrojane veštine koje nastavnici računarskih nauka treba da poseduju.

S druge strane, za srednjoškolsko informatičko obrazovanje postoje sveobuhvatni predlozi standarda. Najznačajniji su ACM model K12 kurikuluma računarskih nauka za sve nivoje obrazovanja od osnovnog do visokoškolskog (CSTA Standards Task Force, 2011) i UNESCO/IFIP kurikulum informacione i komunikacione tehnologije u srednjoškolskom obrazovanju (UNESCO, 2000).

Među brojnim primenama ontologija u obrazovanju (Harchay et al., 2012; Hsu, 2012; Knight et al., 2006; Pathmeswaran & Ahmed, 2009; Romero et al., 2014; Vesin et al., 2012; Xin-juan et al., 2007; Yaghmaie & Bahreininejad, 2011; Wang, 2008), posebno se mogu izdvojiti mogućnosti korišćenja ontologija za reprezentaciju kurikuluma kojima se bave radovi (Chi, 2009; Demartini et al., 2013;

Dexter & Davies, 2009; Elsayed, 2009; Fernández-Breis et al, 2012; Libbrecht et al., 2008; Memon & Khoja, 2009). U njima se ističe podesnost primene ontologija za dizajniranje, održavanje, izmene i praćenje promena kurikuluma od strane ministarstava, akreditacionih tela, eksperata i sl. Svojstva dostupnosti i deljivosti kurikuluma, mogućnosti jednostavnog i brzog prilagođavanja dinamici u oblasti su posebno značajne pri razvoju i održavanju kurikuluma iz oblasti računarskih nauka. Tako je, na primer, u (Cassel et al., 2008) ontološki pristup primenjen za reprezentovanje kurikuluma računarstva.

Ontološko uparivanje predstavlja rešenje problema semantičke heterogenosti pronalaženjem korespondencija između semantički povezanih entiteta ontologija (Shvaiko & Euzenat, 2011). Ono omogućava interoperabilnost podataka i znanja reprezentovanih pomoću ontologija (Euzenat & Shvaiko, 2007; Shvaiko & Euzenat, 2011). Postoji veliki broj tehnika i strategija za uparivanje ontologija, koje su implementirane u brojnim sistemima poput (Cruz et al., 2009; Hu et al., 2008; Huber et al., 2011; Jiménez-Ruiz & Cuenca Grau, 2011; Lambrix & Tan, 2006; Li et al., 2009; Jean-Mary et al., 2009; Ngo & Bellahsene, 2012).

Dobijeni rezultati uparivanja (ontološko usaglašavanje) se mogu koristiti za različite namene, poput semantičkog web pretraživanja, spajanja ontologija iz više domena (Jean-Mary et al., 2009), integraciju i prevođenje podataka (Shvaiko & Euzenat, 2011). U (Faria et al., 2013) se navodi da su ontologije postale integralni deo mnogih domena, a pre svih, biomedicine i geografije. U skladu sa tim i sistemi za usaglašavanje ontologija su često namenjeni usaglašavanju i integraciji biomedicinskih informacija (Jean-Mary et al., 2009; Lambrix & Tan, 2006) ili geoprostornih podataka (Cruz & Sunna, 2008; Vaccari et al., 2009). Ipak, u savremenoj literaturi se retko mogu pronaći primeri implementiranih sistema za usaglašavanje ontoloških reprezentacija kurikuluma (različitih ili istih nivoa obrazovanja). U (Anohina-Naumeca et al., 2012) autori navode važnost postojanja sistema za usaglašavanje studijskih programa modelovanih pomoću ontologija. Kreirane su konceptualne mape koje opisuju kurikulume prevedene u ontologiju, pri čemu nisu ni opisani ni realizovani algoritmi za usaglašavanje studijskih programa. Autori (Gluga et al., 2013) su predstavili novi pristup za dizajn univerzitetskih kurikuluma tako da niz predviđenih nastavnih aktivnosti, sadržaja i ocenjivanja tokom tri do pet godina obezbeđuje učenicima progresivan napredak u sticanju generičkih veština, ciljeva učenja specifičnih za određenu disciplinu i kompetencija predviđenih akreditacijom. Međutim, model predviđa trivijalno semantičko mapiranje koje nije zasnovano na principima ontološkog uparivanja.

Detaljniji prikaz navedene literature dat je u poglavljima koja odgovaraju oblastima obuhvaćenim radovima.

1.1. Sadržaj i tema rada

Istraživanja u ovoj disertaciji imaju za cilj da predlože softversku platformu koja bi omogućila utvrđivanje usaglašenosti računarski razumljivih (ontoloških) modela srednjoškolskih informatičkih kurikuluma i kurikuluma za nastavnike informatike zasnovanih na aktuelnim standardima i postojećim kurikulumima. Na taj način bi bilo omogućeno da informatičko obrazovanje na nižim nivoima (srednjoškolskom, pre svega) sprovodi nastavnički kadar sposoban da odgovori izazovima izuzetno dinamičnih promena u oblasti informatike i računarstva.

Stoga, prvi korak je da se identifikuje stanje u oblasti obrazovanja nastavnika informatike. Stanje u oblasti obrazovanja nastavnika informatike obuhvata modalitete obrazovanja nastavnika i kompetencije koje se tim modalitetima stiču. U disertaciji se stanje identifikuje na osnovu pregleda literature, analize reprezentativnih svetskih i domaćih kurikuluma za nastavnike informatike i preliminarnog istraživanja nad populacijom sadašnjih nastavnika informatike na teritoriji Vojvodine.

Bitan aspekt koji se u disertaciji analizira je potreba i mogućnost standardizacije obrazovanja nastavnika informatike. Veoma je važno utvrditi koje osnovne uslove treba da ispunjavaju kurikulumi za obrazovanje nastavnika informatike čime bi se pružila osnova za definisanje standarda. Polazna pretpostavka je da kompetencije nastavnika informatike moraju biti definisane tako da zadovoljavaju zahteve aktuelnih informatičkih školskih kurikuluma i standarda.

Predmet istraživanja ove disertacije ima dva osnovna pravca.

Prvi pravac obuhvata istraživanje modela domenskih (informatičkih) znanja u nastavničkim kurikulumima sa ciljem njihovog prilagođavanja dinamičnim promenama u oblasti računarstva i standardizovanom modelu ACM K12 CS (Computer Science) kurikuluma (CSTA Standards Task Force, 2011). Takođe, pored domenskih znanja, neophodno je sagledati koje nedomenske oblasti nastavnički kurikulumi treba da sadrže.

Da bi se što kvalitetnije definisao model, pored pregleda savremene literature i svetskih i domaćih kurikuluma, važno je identifikovati i stavove sadašnjih nastavnika informatike o:

- njihovom poznavanju oblasti predloženog modela kurikuluma za nastavnike informatike,
- značaju predloženih oblasti kurikuluma za držanje nastave,
- zastupljenosti informatičkih oblasti u kurikulumima po kojima oni realizuju nastavu.

Međutim, jednom kreiran model ne može biti trajno rešenje. Brzina promena u oblasti informatike i računarstva i primene informacionih i komunikacionih tehnologija u drugim oblastima iziskuju često osavremenjivanje visokoškolskih kurikuluma za nastavnike informatike i permanentno usavršavanje nastavnika. Stoga je potrebno obezbediti reprezentaciju definisanog modela kurikuluma u takvoj formi da njegovo menjanje, održavanje i prilagođavanje promenama u

oblasti bude jednostavno i brzo i da omogućuje aktivno uključivanje akademske zajednice, nadležnih ministarstava, studenata i zaposlenih u informatičkom sektoru u njegovo održavanje i unapređivanje.

Stoga je drugi pravac istraživanja obezbeđivanje (tehničke) pretpostavke za objektivnije i efikasnije kreiranje, upoređivanje i razmenu kurikuluma. Taj pravac se odnosi na razvoj mašinski čitljive ontologije kurikuluma bazirane na već postojećim standardima u domenu obrazovanja i softverske podrške za kreiranje i korišćenje takve reprezentacije kurikuluma bazirane na tehnologijama Semantičkog Weba.

Nadalje, reprezentovanje srednjoškolskog informatičkog kurikuluma i kurikuluma za nastavnike informatike primenom ontologija omogućava primenu principa usaglašavanja ontologija tako da računar (polu)automatski određuje usaglašenost ontoloških reprezentacija kurikuluma. Na taj način je moguće obezbediti da akademska zajednica, nadležna ministarstva, obrazovne institucije i sl. imaju uvid u usaglašenost kurikuluma različitih nivoa obrazovanja, kao i da menjaju rezultate usaglašenosti u skladu sa svojim viđenjem sličnosti pojedinih oblasti i tema analiziranih kurikuluma.

Hipoteza istraživanja u ovoj disertaciji je bila da je moguće razviti mašinski čitljiv model informatičkog kurikuluma za obrazovanje nastavnika informatike, mašinski čitljiv model informatičkog kurikuluma za srednjoškolski nivo obrazovanja i odgovarajuću softversku aplikaciju tako da budu zadovoljeni sledeći zahtevi:

- Da bude omogućeno jednostavnije i brže menjanje, održavanje i prilagođavanje kurikuluma promenama u oblasti.
- Da bude omogućeno usaglašavanje informatičkih kurikuluma različitih nivoa obrazovanja, pre svega nastavničkih kurikuluma i kurikuluma srednjoškolskog nivoa obrazovanja.
- Da različite ciljne grupe (na primer, akademska zajednica, nadležna ministarstva, studenti, sadašnji zaposleni u informatičkom sektoru) mogu da budu aktivno uključene u održavanje i unapređivanje kurikuluma.

Da bi se potvrdila navedena hipoteza, osnovni cilj istraživanja u disertaciji je specifikacija mašinski čitljivog modela informatičkog kurikuluma za obrazovanje nastavnika informatike i mašinski čitljivog modela informatičkog kurikuluma za srednjoškolski nivo obrazovanja koji zadovoljavaju navedene zahteve. Pored toga, cilj je i razvoj prototipa softverske aplikacije koja će biti iskorišćena za verifikaciju predloženih modela.

Rezultati istraživanja u ovoj disertaciji su potvrdili navedene polazne hipoteze:

- Kreiran je mašinski čitljiv model kurikuluma primenom ontologija. Ontološki model je zasnovan na IEEE RCD standardu za definisanje kompetencija.
- Očekivana informatička znanja i ishodi učenika po završetku srednje škole definisana su na osnovu standardizovanog ACM K12 predloga modela kurikuluma.

- Za modelovanje sadržaja i ishoda nastavnčkog kurikuluma analizirani su reprezentativni svetski kurikulumi i aktuelni kurikulumi u Republici Srbiji. Takođe, istraženi su stavovi zaposlenih nastavnika informatike u AP Vojvodina o zastupljenosti, značaju i poznavanju izabranih tema i oblasti, dobijenih analizom kurikuluma i preporuka. Na osnovu istraživanja i analize kurikuluma kreiran je referentni ontološki model nastavnčkog kurikuluma.
- Kreiran je ontološki model nastavnčkog kurikuluma na osnovu odabranog kurikuluma iz Republike Srbije.
- Razvijena je softverska platforma koja omogućava poređenje nastavnčkih i srednjoškolskih kurikuluma, kao i poređenje nastavnčkih kurikuluma međusobno, tako da potencijalnim korisnicima sistema omogućava:
 - Uvid u stepen podudarnosti kurikuluma.
 - Uvid u koncepte koji nisu upareni.
 - Uvid u nekonzistentne ishode i sadržaje.
 - Mogućnost izmene rezultata uparivanja i slično.
- Razvijena softverska aplikacija je verifikovana, što je podrazumevalo korišćenje prototipskog softvera za utvrđivanje slaganja za sledeće kombinacije:
 - Model srednjoškolskog informatičkog kurikuluma, zasnovanog na ACM K12 modelu i model referentnog kurikuluma za nastavnike informatike.
 - Model referentnog kurikuluma za nastavnike informatike i model izabranog nastavnčkog kurikuluma u Republici Srbiji.
 - Model srednjoškolskog informatičkog kurikuluma zasnovanog na ACM K12 modelu i model izabranog nastavnčkog kurikuluma u Republici Srbiji.

1.2. Definicije najvažnijih pojmova (računarskih nauka, računarstva, informatike) u kontekstu srednjoškolskih i nastavnčkih kurikuluma

U obrazovnoj zajednici se značaj izučavanja računarstva u srednjim školama razmatra još od 1970tih (Armoni, 2011). Kontinuirano se menjaju shvatanja o tome šta su računarske nauke (engl. *Computer Science* – CS), koje oblasti obuhvataju, šta CS eksperti, a šta studenti treba da znaju i koje sposobnosti iz CS oblasti treba da imaju. Zbog relativno skorog nastanka oblasti računarstva, interdisciplinarne prirode, dinamike razvoja i dijapazona njenih primena, i najvećim ekspertima u oblasti je veoma teško da definišu sadržaj i granice naučne discipline koja bi u sebi objedinila sve ove aspekte. Ipak, postoji opšta saglasnost da računarske nauke na najbolji način trenutno predstavljaju naučni aspekt računarstva.

Definicija iz izvora WordNet Search - 3.1 (<http://wordnetweb.princeton.edu/perl/webwn>), u kojoj se termini računarske nauke i računarstvo smatraju sinonimima, kaže sledeće: „

Računarske nauke, računarstvo (oblast inženjerske nauke koja proučava (uz pomoć računara) sračunljive procese i strukture)". U ACM modelu K12 kurikulumu računarskih nauka (Tucker et al., 2006) data je sledeća definicija računarskih nauka namenjena srednjoškolskim nastavnicima CS: „*Računarske nauke (CS) su izučavanje računara i algoritamskih procesa, uključujući i njihove principe, njihov hardverski i softverski dizajn, njihove primene i njihov uticaj na društvo.*“ koja je prihvaćena i u (Gal-Ezer & Stephenson, 2010). U (Armoni, 2011) su računarske nauke definisane kao *naučna disciplina usmerena ka identifikovanju problema koji su računarski rešivi, a zatim i rešavanju problema, pre svega u domenu algoritamskog načina razmišljanja, i, ponekad, implementaciji rešenja u nekom od programskih jezika.*

Drugi termin koji se koristi da označi oblast koja je predmet interesovanja ovoga rada je *informatika* za koji se, takođe, mogu naći različite definicije. Ekvivalente termine predstavljaju reči *Informatik* i *Informatique* u nemačkom i francuskom jeziku, respektivno. Pojam informatika je uveo Dreyfus 1962. godine i po (Dreyfus, 1962): „***Informatika*** je oblast naučne, tehničke i industrijske aktivnosti namenjena automatskoj obradi informacije kroz izvršavanje računarskih programa pomoću mašina“. Na Univerzitetu u Edinburgu (http://www.ed.ac.uk/polopoly_fs/1.40075!/fileManager/What%20is%20Informatics.pdf) je dato sledeće značenje: „*Informatika proučava strukturu, ponašanje i interakcije prirodnih i inženjerskih računarskih sistema. Informatika izučava reprezentaciju, obradu i razmenu informacije u prirodnim i inženjerskim sistemima*“. U izvoru (UNESCO, 2000) Informatika (računarske nauke) se definiše kao *nauka koja se bavi dizajnom, realizacijom, evaluacijom, primenom i održavanjem informacionih sistema; uključujući hardver, softver, organizacione i ljudske aspekte, i industrijske, komercijalne, državne i političke implikacije.*

U ovoj disertaciji je opredeljenje da se ravnopravno koriste termini *računarske nauke*, *informatika* i *računarstvo* kao termini koji označavaju ono što je definisano kao računarske nauke u ACM modelu K12 kurikulumu.

Poglavlje II

2. Istraživanje kurikuluma srednjoškolskih nastavnika informatike

2.1. Postojeći standardi u srednjoškolskom informatičkom obrazovanju

Prema (Ericson et al., 2008) postoji jasna veza između izučavanja računarstva, odnosno računarskih nauka i ekonomskog razvoja društva. Potražnja na tržištu radne snage za kvalitetnim ekspertima u oblasti je izuzetno velika, ali i druge profesije zahtevaju odgovarajuće informatičko obrazovanje zbog uloge informaciono-komunikacionih tehnologija (IKT) u svim oblastima.

Srednjoškolsko obrazovanje moralo bi da obezbedi osnovu za efektivno i efikasno korišćenje IKT u budućoj profesiji i svakodnevnom životu, solidnu osnovu za više nivoe obrazovanja u IKT oblasti i, za određene profile srednjih škola, znanja za obavljanje jednostavnijih IKT poslova nakon završene srednje škole.

S obzirom na univerzalnost oblasti IKT, njen značaj za razvoj savremenog društva i shvatanje da premošćavanje „digitalnog jaza” može presudno da doprinese ujednačavanju stepena razvoja društvenih zajednica, jasno je prepoznata potreba kreiranja i implementacije globalno prihvaćenog standarda informatičkog obrazovanja i u srednjim školama.

Postoje dva glavna pokušaja standardizacije informatičkog obrazovanja u srednjim školama:

- UNESCO/IFIP kurikulum srednjoškolskog IKT obrazovanja (UNESCO, 2000) i
- K12 standardi računarskih nauka (CSTA Standards Task Force, 2011; Tucker et al., 2006).

UNESCO/IFIP kurikulum srednjoškolskog IKT obrazovanja strukturiran je u četiri nivoa kompetencija:

- IKT pismenost,
- domenske primene IKT-a,
- integracija domena putem IKT-a,
- IKT specijalizacija.

Detaljnijim uvidom u četiri oblasti IKT kompetencija koje predviđa ovaj standard u (Micheuz, 2008) je uočeno da prvi nivo - IKT pismenost (sa izuzetkom poslednja dva modula) i deo drugog nivoa obuhvataju u potpunosti jezgro ECDL (European Computer Driving License) kursa, i to: osnovne koncepte IKT-a; primenu računara i upravljanje fajlovima; obradu teksta; rad sa tabelama; baze podataka; kreiranje dokumenata i prezentacija; informisanje i komunikacija; društvena i etička pitanja; posao i IKT. Ipak, u (Micheuz, 2008) se ističe da neki autori izražavaju bojazan da preveliko potenciranje

integriranog pristupa IKT-a u okviru drugih disciplina, koje je prisutno u predlogu kurikuluma (UNESCO, 2000), može imati za posledicu izostanak posebnih obaveznih informatičkih predmeta u srednjim školama.

U UNESCO/IFIP kurikulumu srednjoškolskog IKT obrazovanja su posebno definisani informatika (računarske nauke) (data u uvodu ove disertacije), informacione i komunikacione tehnologije (IKT) i informacione tehnologije. Međutim, u istom izvoru se navodi da će u dokumentu sve definicije biti svedene na jednu, sveobuhvatnu definiciju IKT-a, „što podrazumeva da će IKT biti korišćen, primenjen i integrisan u radne aktivnosti i učenje na osnovu konceptualnog razumevanja i metoda informatike“ (UNESCO, 2000).

2.1.1. ACM model K12 kurikuluma računarskih nauka

Najkompletniji i, sudeći po analiziranoj literaturi, najčešće citiran predlog predstavlja ACM-ov K12 predlog standarda za sve nivoje obrazovanja, do visokoškolskog (CSTA Standards Task Force, 2011). Prema (Tucker et al., 2006) računarske nauke u srednjim školama uključuju: algoritamski način razmišljanja, matematičke osnove računarskih nauka, računarski hardver, programiranje i različite programske paradigme, operativne sisteme, računarske mreže, upravljanje podacima (baze podataka i nalaženje informacija), računarsku grafiku, Internet i web tehnologije (organizacija Interneta, dizajn web stranica), aplikativni softver, veštačku inteligenciju, etička i pravna pitanja (bezbednost, privatnost, pravo intelektualne svojine, prednosti i mane softvera na javnim domenima, pouzdanost informacija na Internetu), ograničenje računara, karijere u računarstvu.

ACM-ov K12 predlog smatra se savremenijim i sveobuhvatnijim od UNESCO/IFIP-ovog kurikuluma srednjoškolskog IKT obrazovanja zbog toga što je zasnovan na računarskim naukama, definisanim tako da predstavljaju širu, adekvatniju i savremeniju naučnu disciplinu od IKT-a, definisanog kao u (UNESCO, 2000). Takođe, ACM-ov K12 predlog je usklađen sa dinamikom promena u oblasti računarskih nauka, na šta ukazuju i broj revizija i godine izdanja aktuelnih verzija ovog standarda (CSTA Standards Task Force, 2011; Tucker et al., 2003; 2006).

2.2. Aktuelno stanje u oblasti obrazovanja nastavnika informatike

Da bi srednje škole ispunile zahteve koje propisuju pomenuti standardi kurikuluma srednjoškolskog informatičkog obrazovanja, potrebno je da raspolažu odgovarajućim nastavničkim kadrom. Iako literatura iz oblasti (Eurydice, 2004; Hazzan et al., 2010; Stephenson et al., 2005) ukazuje na važnost postojanja, kako srednjoškolskih tako i nastavničkih kurikuluma iz oblasti informatike, odnosno da „računarske nauke zaslužuju“ odgovarajuću pripremu nastavnika (Ragonis et al., 2010), u radovima (Armoni, 2011, Ericson et al., 2008; Ragonis et al., 2011) autori ocenjuju da obrazovni sistemi u

nekim zemljama čak ni ne predviđaju posebne studijske programe za nastavnike informatike. Takvo stanje uzrokuje da u mnogim srednjim školama informatiku predaju nastavnici koji nemaju adekvatno formalno obrazovanje, poput nastavnika matematike, tehničkog obrazovanja ili nastavnika drugih naučnih disciplina. Izvor (Ericson et al., 2008) analizira stanje kompetencija nastavnika računarstva u srednjim školama u SAD i ističe značajan nedostatak konzistentnosti u sertifikaciji nastavnika računarskih nauka u srednjim školama širom sveta, i posebno u SAD. Isti izvor na strani 12 iznosi tvrdjenje da u mnogim slučajevima nastavnici koji, po formalnom obrazovanju, poseduju znanja i veštine propisane ACM K12 kurikulumom (recimo, imaju nastavničku akreditaciju i postdiplomske studije iz oblasti računarskih nauka) ne mogu da budu sertifikovani kao nastavnici informatike, dok je to omogućeno nastavnicima koji imaju diplomu u oblasti biznisa ili druge struke bez osnove iz oblasti računarskih nauka.

Izvori (Ericson et al., 2008; Wilson et al., 2010) tvrde da postojeću krizu obrazovanja nastavnika za informatičke predmete uzrokuju dva ključna faktora:

- nedostatak jasnosti, razumevanja i konzistentnosti u važećim zahtevima sertifikacije, i
- nedostatak veze između postojećih zahteva sertifikacije i aktuelnog sadržaja same discipline računarskih nauka.

Na osnovu rezultata drugih istraživača i sopstvenih istraživanja, autor ove disertacije smatra da su faktori identifikovani u (Ericson et al., 2008; Wilson et al., 2010) posledica sledećih osnovnih uzroka krize u obrazovanju sadašnjih i budućih nastavnika informatike:

- složenosti i tradicionalne osnove oblasti,
- dinamike promena u oblasti, i
- specifičnog položaja oblasti u obrazovanju u odnosu na druge oblasti.

Složenost oblasti čini izuzetno teškim određivanje granica i sadržaja oblasti (na primer, biološki inspirisano računarstvo, obrada prirodnog jezika, tehnološki podržano učenje, itd.), a tradicionalne osnove (pre svega, elektrotehnika i matematika), tendencijama da je odrede kao svoj deo, otežavaju njeno prihvatanje kao posebne naučne oblasti.

Ekstremno visoka dinamika promena u oblasti otežava formiranje stabilnog i trajnog (nastavničkog) kurikulumuma čiji su ishodi usaglašeni sa promenama u oblasti.

Specifičan položaj oblasti ogleda se u njenoj trostrukoj ulozi u (srednjoškolskom) obrazovanju: kao osnovni predmet izučavanja; kao skup metoda i tehnika koje se primenjuju u drugim oblastima; kao skup pripadajućih tehnologija koje se široko primenjuju u svim oblastima i, posebno, u obrazovanju. Ovakav položaj zahteva formiranje nastavničkih kurikulumuma koji će da zadovolje zahteve različitih profila srednjoškolskog obrazovanja (što zahteva pokrivanje prve dve navedene uloge), da obezbede opšte IKT veštine i da

maksimalno iskoriste prednosti primene IKT u obrazovanju (što zahteva pokrivanje i treće navedene uloge).

2.3. Potreba za standardizacijom

Opšte je prihvaćen stav u literaturi da su jasni i specifični standardi koje treba da zadovolje nastavnički kurikulumi preduslov za izlazak iz ove krize koja može da ima izuzetno negativne šire posledice. U (Ericson et al., 2008) se navodi da bi nepostojanje jasnih i specifičnih standarda koje nastavnički kurikulumi treba da ispune moglo da dovede da smanjenja broja i snižavanja kvaliteta fakulteta koji obrazuju buduće nastavnike informatike. To, dalje, može dovesti do snižavanja opšte informatičke pismenosti učenika u osnovnim i srednjim školama i manjeg interesovanja učenika u srednjim školama za upis na studijske programe iz oblasti računarskih nauka, odnosno manjeg broja budućih stručnjaka iz te oblasti. U (Armoni, 2011) i (Wilson et al., 2010), navodi se da je upis novih studenata na CS orijentisane studijske programe neodgovarajući ako se posmatraju zahtevi tržišta za ovim stručnim profilom. Postojanje profesionalnih organizacija poput CSTA (Computer Science Teachers Association) predstavlja pokazatelj da je akademska i stručna zajednica svesna postojećih problema (Ragonis et al., 2011). U CSTA izveštaju (Ericson et al., 2008) detaljno se obrazlaže važnost kreiranja i usvajanja standarda za sertifikaciju nastavnika informatike. CSTA izveštaj posebno ističe važnost permanentnog obrazovanja nastavnika informatike koji već rade u školama. Računarstvo je specifična naučna disciplina čija dinamična priroda može da se ogleda u blažim promenama u nastavnim sadržajima, poput uvođenja nove tehnologije u okviru iste paradigme (prelazak sa jezika Pascal na C) ili u mnogo „dramatičnijim“, kao što je promena same paradigme programiranja (prelazak sa proceduralnog na objektno-orijentisano programiranje). To znači da nastavnici, za svaku od mogućih promena, moraju biti spremni za sopstveni profesionalni razvoj bilo „organizovano ili nezavisno“ (Armoni, 2011).

Imajući u vidu stanovište da nedostaju jasni i specifični standardi koji bi definisali znanja i veštine koje je potrebno da obezbede kurikulumi za nastavnike informatike, deo ovog rada je i prikaz i upoređivanje preporuka relevantnih udruženja i akreditacionih tela iz oblasti (East et al., 2011; Ericson et al., 2008), aktuelne literature (Grgurina, 2008; Niemi & Jakku-Sihvonen, 2009; Ragonis et al., 2011) i odabranih svetskih kurikuluma (Nemačka, Izrael, Turska, Škotska, SAD, Estonija, Austrija, Republika Srbija), da bi se kroz zajedničke karakteristike i razlike pružila osnova za definisanje standarda.

Kao što je već rečeno, u ovom radu je opredeljenje da se ravnopravno koriste termini *računarske nauke*, *informatika* i *računarstvo*, budući da, na primer, pojmu informatike u nemačkim kurikulumima za nastavnike informatike odgovara pojam računarske nauke iz američkih i izraelskih kurikuluma, odnosno pojam informatike i računarstva u kurikulumima u Republici Srbiji. Dakle, u

ovom radu su upoređivani, u zavisnosti od zemlje čiji je kurikulum izučavan, kurikulumi pod nazivima: Nastavnik informatike, Program za pripremu nastavnika računarskih nauka, Nastavnik informatike i računarstva i sl.

2.4. Preporuke međunarodnih organizacija

CSTA (Computer Science Teachers Association) je organizacija koja promovise i podržava predavanje računarskih nauka i drugih računarskih disciplina. Pod pokroviteljstvom CSTA je objavljen niz radova u oblasti CS edukacije, od kojih su za obrazovanje budućih CS nastavnika najznačajniji (Ericson et al., 2008) i (Gal-Ezer & Stephenson, 2010). U (Gal-Ezer & Stephenson, 2010) se elaborira postojeća kriza u sertifikaciji i programima za edukovanje CS nastavnika, dok je vrlo detaljan pregled aktuelnog stanja dat nešto ranije u (Ericson et al., 2008). U ovim radovima su predloženi i modeli obrazovanja nastavnika. Poseban kvalitet je što su predlozi modela obrazovanja nastavnika povezani sa ACM K12 modelom kurikulumu računarskih nauka (Tucker et al., 2006).

Dakle, u (Ericson et al., 2008) su date preporuke o potrebnim programima i kursovima za (do)školoavanje CS nastavnika namenjene različitim ciljnim grupama: budući nastavnici (aktuelni studenti odgovarajućih studijskih programa), nastavnici sa iskustvom koji poseduju sertifikaciju iz drugih oblasti i nikada nisu predavali CS, nastavnici koji imaju sertifikaciju iz drugih oblasti i imaju iskustva u predavanju CS kurseva, i pojedinci sa CS kompetencijama ali bez pedagoškog iskustva. Tako se, na primer, za buduće nastavnike zahtevaju sledeći uslovi:

- **Akademski zahtevi iz oblasti CS:**
 - Bečelor nivo iz CS ili, u slučaju drugih specijalnosti koje sadrže CS predmete, položeni kvalifikacioni ispiti iz sledećih oblasti: programiranje, objektno-orijentisani dizajn, strukture podataka i algoritmi, računarski hardver i organizacija, računarski modeli,
 - Kurs u vidu seminara koji bi uključivao teme istorije računarskih nauka, prirodu oblasti i vezu sa drugim disciplinama, poznavanje CS kurikulumu na srednjoškolskom i univerzitetskom/koledž nivou, različite aspekte vezane za probleme koji mogu nastati pri predavanju oblasti programiranja, kao i korišćenje alata i sredstava za predavanje CS,
 - Napisan istraživački rad iz oblasti CS obrazovanja.
- **Akademski zahtevi iz oblasti obrazovanja (nespecifični za CS):**
 - obrazovna psihologija,
 - primena tehnologija u učionici,
 - dizajn i razvoj kurikulumu.
- **Metodika nastave CS i iskustvo u oblasti:**
 - Kurs metodike nastave CS,

- Praksa koja obuhvata prisustvo nastavi i minimum 10-15 nedelja praktične nastave (predavanja).
- **Praktičan ispit iz generalne pedagoške osposobljenosti.**

Ukoliko je studijski program namenjen zaposlenim nastavnicima koji nemaju iskustva i u predavanju kurseva iz oblasti računarskih nauka, onda se u CSTA izveštaju navodi da je potrebno da takvi nastavnici pored metodičkih kurseva savladaju i oblasti: programiranja, objektno-orijentisanog dizajna, strukture podataka i algoritama, računarskog hardvera i organizacije. Međutim, bez obzira da li ciljnu grupu predstavljaju sadašnji ili budući nastavnici ili zaposleni bez pedagoškog iskustva, u (Ericson et al., 2008) se navodi da svaki program za pripremu CS nastavnika mora uključiti četiri osnovne komponente:

- Akademske zahteve iz oblasti računarskih nauka,
- Akademske zahteve iz oblasti obrazovanja,
- Metodološke (ili metodičke) kurseve i iskustvo u oblasti,
- Opšta pedagoška znanja.

National Council for Accreditation of Teacher Education (NCATE) je akreditaciono telo u USA za akreditaciju studijskih programa na kojima se obrazuju budući nastavnici. Od 1990. je razvijen niz standarda za pripremu srednjoškolskih CS nastavnika, pri čemu NCATE promovise programe koji se zasnivaju na K12 modelu kurikuluma. Standardi se odnose na sadržaje iz oblasti računarskih nauka, ali i na sposobnosti obrazovnih stručnjaka. Poslednja, (treća) revizija ovih standarda realizovana je u saradnji sa SIGCT (the Special Interest Group for Computing Teachers of ISTE, the Specialty Professional Association collaborator with NCATE) i CSTA 2011. godine. Najnoviji predlozi prikazani su u (East et al., 2011; ISTE, 2011) gde su nabrojana detaljno strukturirana znanja i sposobnosti koje nastavnici informatike treba da poseduju. Ova preporuka predstavlja ažuriranu verziju programa za inicijalnu pripremu nastavnika iz 2002. godine koja je detaljno opisana u (ISTE, 2002). Budući da su u (ISTE, 2002) „Secondary Computer Science Standards” predviđeni kao dodatno odobrenje na postojeću nastavničku licencu, preporučeni standardi za nastavnike informatike mahom ne sadrže opšte pedagoške oblasti i opšta znanja i veštine koja se stiču tokom osnovnog nastavničkog obrazovanja. Ipak, potrebno je da kandidati pre upisa ispune potrebne uslove iz poznavanja obrazovne tehnologije (ISTE National Educational Technology Standards for Teachers). Standardi, prikazani u (East et al., 2011), se sastoje od četiri principa:

- I. Poznavanje sadržaja.
 - I.A. Apstrakcija i reprezentacija podataka.
 - I.B. Dizajn algoritama, razvoj i testiranje.
 - I.C. Digitalni uređaji, sistemi i mreže.
 - I.D. Uloga i uticaj računarstva u savremenom okruženju.
- II. Efikasno podučavanje i strategije učenja.
 - II.A. Planiranje i predavanje CS lekcija/nastavnih jedinica.
- III. Efikasna učenička okruženja (u učionici i on-line).

- III.A. Okruženja za efikasno podučavanje i učenje u učionici i on-line učenje.
- IV. Profesionalna znanja i sposobnosti.
 - IV.A. Kontinuirani profesionalni razvoj i doživotno učenje.

Svaki od principa i njima pripadajuće podteme se dalje detaljnije dele pa se, na primer, I.B. dizajn algoritama, razvoj i testiranje sastoji od: znanje programskog jezika visokog nivoa; kompleksni logički izrazi primenom Bulovih operatora; dizajn i test algoritama i programerskih rešenja; selektovanje odgovarajuće strukture podataka radi rešavanja problema; znanje dve ili više programerske paradigme, poznavanje dva ili više integrisanih razvojnih okruženja itd.

2.5. Rezultati naučnih radova i iskustva iz različitih zemalja

Prirodna osnova na koju se oslanjaju savremena istraživanja u oblasti nastave informatike su generalne preporuke za nastavničke kurikulume. U toj oblasti osnovu mogu pružiti Shulman-ove preporuke (Shulman 1986, 1987), po kojima se razlikuje pet osnovnih tipova znanja potrebnih nastavniku: znanje domenskih sadržaja, poznavanje drugih tema koja ne predstavljaju osnovnu disciplinu, „poznavanje učenika“, „znanje obrazovnih ciljeva“ i „generalno pedagoško znanje“. U odnosu na tradicionalan pristup koji je do tada važio i u kome se kao osnova programa za pripremu nastavnika ističe pedagoško znanje, Shulman naglašava važnost, pre svega, znanja domena koji se izučava/predaje. Pri tome se, po istom autoru, znanje domenskih sadržaja dalje može podeliti na tri tipa: znanje glavnog predmeta; pedagoško znanje vezano za predmet izučavanja; poznavanje kurikuluma. I drugi autori (Zeidler, 2002) smatraju da znanja domena koji se izučava, pedagoška domenska znanja i opšta pedagoška znanja moraju biti deo svakog programa za pripremu nastavnika.

U (Shulman 1990) se zastupa stav da nastavnički kurikulumi treba da izučavaju opšte kurseve (filozofija, psihologija, sociologija) samo kao širi kontekst instrukcionog sadržaja. Ipak, u literaturi (Pietig, 1997) se mogu sresti i suprotna mišljenja po kojima je neophodno sociološke ili psihološke kurseve odvojeno izučavati, a ne kao deo metodičkih kurseva. Aktuelni kurikulumi najčešće pomiruju ove dve škole mišljenja; postoje odvojeni teorijski kursevi iz ove oblasti, a u metodičkim kursevima su ta znanja implementirana na način kao što Shulman sugeriše.

Analizom izvora koji se specifično bave uvođenjem standardizacije u srednjoškolsku nastavu informatike može se uočiti da ranijim radovima (Gal-Ezer, 1995; Poirot et al., 1985, 1988) nedostaju važne oblasti koje, sa današnjeg stanovišta, nastavnički informatički kurikulumi moraju da sadrže. Tako u (Poirot et al., 1988) nisu navedeni kursevi iz oblasti pedagoških znanja potrebnih za predavanje računarskih nauka, dok program prikazan u (Gal-Ezer, 1995) ne sadrži opšte obrazovne kurseve ili praksu.

Aktuelnim istraživačkim rezultatima u oblasti CS kurikuluma koji se odnose na značaj postojanja posebnih kurikuluma, sadržaje kurikuluma i modalitete obrazovanja nastavnika informatike se ističu izraelski autori Hazzan, Ragonis, Gal-Ezer i Lapidot (Gal-Ezer & Stephenson 2010; Gal-Ezer & Zur, 2007; Hazzan et al. 2008, 2010; Hazzan & Lapidot, 2004, 2006; Lapidot & Hazzan, 2005; Ragonis et al., 2010, 2011). U (Hazzan et al., 2010) se navodi da se izraelski model srednjoškolskog CS obrazovanja smatra za „jedan od vodećih“ svetskih srednjoškolskih CS kurikuluma. Po tom modelu „centralna uloga“ pripada programima za pripremu nastavnika. Po istom izvoru, programi za pripremu nastavnika informatike najčešće sadrže četiri osnovne oblasti:

- Disciplinarnе studije računarskih nauka,
- Generalne pedagoške studije, poput izučavanja psihologije i generalnih nastavnih veština,
- CS pedagoška domenska znanja, pre svega kurs metodike nastave računarskih nauka,
- Praktikum u držanju srednjoškolske CS nastave.

Uslovi koje srednjoškolski nastavnici treba da ispune su najmanje bečelor nivo iz računarskih nauka i CS nastavnički sertifikovan studijski program, kao i da poseduju licencu za izvođenje CS nastave (Hazzan et al., 2008). Master diploma je potrebna za predavanje bilo koje naučne discipline u Izraelu. Međutim, isti izvor navodi da, poput mnogih drugih zemalja, postoje izuzeci koji se najčešće odnose na starije nastavnike koji nemaju završene bečelor studije iz CS oblasti. Iako se očekuje da nastavnici već imaju CS bečelor nivo, u (Ragonis et al., 2010) se navodi da CS nastavnici moraju da poseduju i opšta CS znanja iz tema koja se uobičajeno ne izučavaju u CS specifičnim kursevima, poput istorije računarskih nauka, veze sa drugim disciplinama, definicija najvažnijih pojmova iz oblasti računarskih nauka i sl. Takođe, u predloženom modelu nastavničkog kurikuluma autori posebno naglašavaju važnost CS metodičkog kursa, koji je zasnovan na NCATE standardima, Shulman-ovom modelu nastavničkog znanja (Shulman, 1986), i detaljno opisan u radovima (Hazzan & Lapidot, 2006; Lapidot & Hazzan, 2003, 2005). Ovde se predlaže i obavezna praksa pod „dvojnim“ mentorstvom: školskog nastavnika i univerzitetskog profesora (Hazzan & Lapidot, 2004). Po modelu srednjoškolskog CS obrazovanja, detaljno prikazanom u (Hazzan et al., 2008), nastavnički i srednjoškolski kurikulum su i indirektno povezani preko važne oblasti istraživanja u CS obrazovanju koja uključuje čitanje i pretraživanje aktuelne literature, istraživačke metode, sprovođenje akcionog istraživanja, i samostalni profesionalni razvoj, uopšte.

Radovi (Grgurina & Tolboom, 2008; Van Diepen et al., 2011), opisuju informatičko obrazovanje u Holandiji, pre svega, na srednjoškolskom nivou, dok su u (Grgurina, 2008) opisani modaliteti obrazovanja nastavnika informatike. Iz tih radova se može videti da je u Holandiji 1998., zbog situacije u kojoj nisu postojali kompetentni nastavnici koji bi predavali informatičke kurseve u srednjoj školi,

osnovan konzorcijum od 12 univerziteta i visokoškolskih institucija – CODI (*Consortium Omscholing Docenten Informatica*). CODI program je bio namenjen zaposlenim nastavnicima u školama i obuhvatao je važne savremene informatičke aspekte (arhitektura računarskih sistema i operativne sisteme, vizualno programiranje u programskom jeziku Java, programske paradigme, softversko inženjerstvo, informacione sisteme, baze podataka i sl.), ali i didaktiku informatike i zadatke iz praktične nastave. Program je obustavljen 2005. godine, budući da je pet univerziteta u Holandiji (tek od tada) u svoje kurikulume inkorporiralo posebne (master) studije za nastavnike informatike u srednjim školama. Uslov za upis master studija (obrazovanja i komunikacija) na ovim univerzitetima je bečelor nivo iz informatike. Iskustvo u Holandiji (Grgurina & Tolboom, 2008) ukazuje na to da samo mali broj kandidata koji ispunjavaju predviđeni uslov nastavljaju studije za nastavnika informatike. Sa druge strane, mnogi koji bi želeli da postanu “Licencirani nastavnici Informatike” i poseduju zadovoljavajuća informatička znanja i nastavničko iskustvo nisu u mogućnosti da to ostvare, budući da nemaju odgovarajuću bečelor diplomu. Studijski program Univerziteta u Groningenu (University Center for Academic Learning and Teaching - UOCG) namenjen budućim nastavnicima informatike u srednjim školama prikazan je u (Grgurina, 2008). Uslov za upis ovih dvogodišnji master studija je, takođe, bečelor nivo iz informatike, ali se mogu upisati i kandidati koji ne ispunjavaju taj uslov ukoliko zadovolje proveru znanja iz oblasti softvera (programiranja i algoritama), informacionih sistema, hardvera (mašina i infrastrukture), poznavanja računarskih nauka kao naučne discipline. Studijski program, u okviru teorijske pripreme nastavnika, predviđa izučavanje najpre opštih metodičkih principa, a zatim specifične metodike računarskih nauka. U kurikulumu su sadržani i drugi opšti pedagoški kursevi poput “procesa učenja”, izbornih kurseva o dizajniranju i poznavanju studijskih programa, upravljanje kvalitetom u školama, multikulturalnog obrazovanja, IKT menadžmenta u školama i sl. Studenti realizuju nastavnu praksu u kolaborativnom okruženju u školi. Postoji trener - nastavnik koji predaje iste ili slične teme, i nastavnik edukator koji je zadužen za organizovanje sastanaka sa svim studentima, kao i ocenjivanje njihovog napretka.

Visok PISA rejting koji postižu finski učenici je, po (Niemi & Jakku-Sihvonen, 2009), posledica svrsishodne obrazovne politike i odgovarajućih kompetencija nastavnika. U istom izvoru su opisani nastavnički kurikulumi generalno, bez obzira na domen koji se izučava (informatika, matematika, itd). Nastavnički studijski programi traju 5 godina i nastavnici, po okončanju studija, stiču master diplomu. Kurikulum za obrazovanje nastavnika za srednje škole obuhvata: akademsku disciplinu (matematika, strani jezik, računarske nauke/računarske tehnologije, društvene nauke, itd.), pedagoške studije, komunikaciju, jezike i IKT, razvijanje samostalnog plana studiranja, istraživačke studije koje uključuju i završni rad, kao i niz izbornih kurseva. U opšte pedagoške studije spadaju kursevi didaktike, obrazovne psihologije, sociologije obrazovanja, a neki

kurikulumi sadrže i filozofiju obrazovanja. Deo didaktike se posebno odnosi na pedagoško domensko znanje koje je analogno onom definisanom u (Shulman, 1986). Naročito se ističe važnost nastavne prakse osmišljene tako da studenti najpre posmatraju rad škole i đaka iz obrazovne perspektive, a zatim izučavaju specifične tematske oblasti i proces učenja kod đaka. Na kraju realizacije prakse studenti preuzimaju odgovornost za održavanje nastave (pripremu i unapređivanje nastavnih lekcija). Ceo proces nastavne prakse odigrava se uz nadzor univerzitetskih nastavnika ili lokalnih školskih nastavnika u zavisnosti od faze prakse. Praksa se većinom realizuje u „normalnim“ školama (engl. *Normal school*) u kojima nastavnici imaju dualnu ulogu: supervizora/mentora i držanja regularne nastave. Slični modeli nastavne prakse mogu se pronaći i u relevantnoj literaturi poput (Schoon & Sandoval, 1997). Jedan od najvažnijih aspekata u finskom kurikulumu za nastavnike u srednjim školama je istraživački rad kao osnova nastavnikovog budućeg doživotnog učenja. Cilj je učenje studenata kako da pronađu i analiziraju probleme sa kojima se mogu susresti u budućem radu. Za podršku istraživanju je osnovan poseban LUMA centar (<http://www.luma.fi/centre/>) koji obezbeđuje interakciju među školama, univerzitetima, tržištem i industrijom, pruža nastavnicima poslednje istraživačke rezultate u oblasti, informacije o predstojećim konferencijama i radionicama, i mogućnosti organizovanja godišnjih konferencija i letnjih kurseva.

I u (Armoni, 2011) se navodi preporuka po kojoj je potrebno da programi za pripremu nastavnika razvijaju istraživačke sposobnosti kod budućih nastavnika. Važno je da nastavnici steknu znanja i veštine koje bi im omogućile samostalni profesionalni razvoj, kao i sprovođenje akcionog istraživanja u razredu radi unapređenja nastave i evaluacije nastavnog procesa.

2.6. Klasifikacija oblasti koje svaki program za pripremu nastavnika informatike treba da sadrži

Na osnovu analize preporuka relevantnih organizacija i naučnih radova iz predmetne oblasti, znanja koje treba da obezbede savremeni kurikulumi nastave informatike su u ovom radu svrstana u 5 (ne nužno disjunktne) oblasti:

- **Informatička (domenska) znanja** se odnose na užu domen informatike/računarskih nauka. Domenskim znanjima pripadaju predmeti poput: programskih jezika, baza podataka, računarskih mreža, kompjuterske etike, arhitekture računara i sl.
- **Opšta obrazovna i pedagoška znanja** čine predmeti iz psihologije (obrazovanja), filozofije, sociologije, pedagogije i opšte didaktike i odnose se na: logičko-gnoseološku strukturu znanja; uticaje javnih institucija i individualnih iskustava na obrazovanje i njegove ishode; svrhe, oblike, metode i rezultate obrazovanja; odnos saznavanja u nauci i nastavi; pedagoške teorije; strategije učenja (nastavne metode); instrukcije u

nastavi (koncepti za personalizovano učenje, kolaborativna nastava); tehnike valorizacije učeničkog postignuća i evaluacije nastavnog procesa (merenja i ocenjivanja) i sl.

- **Domenska metodičko-didaktička znanja** integrišu informatička (domenska) znanja, pedagoška domenska znanja (PCK po Shulman-u) i opšta pedagoška znanja (Armoni, 2011; Hubwiser, 2009; Lapidot & Hazzan, 2003; Nikitchenko, 2011; Tedre, 2007) i obuhvataju: logičko-gnoseološku strukturu znanja i razvoj pojmova u informatici; jezik informatike; specifičnosti saznavanja u nastavi informatike; nastavne strategije i instrukcije u nastavi informatike; oblike rada u nastavi informatike; tok nastavnog procesa u nastavi informatike; planiranje nastave i kurikulum informatike; valorizaciju učeničkog postignuća u nastavi informatike; usavršavanje nastavnika i samoevaluaciju u nastavi informatike.
- **Znanja nastavne prakse** obuhvataju znanja iz organizacije rada škole, upravljanja školom, delatnost odeljenskog starešine, metodičku praksu u realizovanju informatičkih sadržaja, istraživanje i sl.
- **Opšta znanja** obuhvataju strane jezike, matematiku, fiziku, hemiju, menadžment i sl.

Kako definisana klasifikacija uzima u obzir preporuke, kurseve i kategorizacije navedene u analiziranoj literaturi i kurikulumima, moguće je uspostaviti konzistentno semantičko mapiranje koncepata iz analiziranih radova na predloženu klasifikaciju. Na primer, Shulman-ova znanja glavnog predmeta odgovaraju domenskim znanjima, „poznavanje drugih tema“ koja ne predstavljaju osnovnu disciplinu pripadaju kategoriji opštih znanja; „poznavanje učenika“, „znanje obrazovnih ciljeva“ i „generalno pedagoško znanje“ spadaju u opšta obrazovna i pedagoška znanja; pedagoško znanje vezano za predmet izučavanja i „poznavanje kurikuluma“ se mogu definisati kao domenska metodička znanja. Slično, četiri osnovne komponente koje svaki kurikulum za pripremu nastavnika mora da sadrži po CSTA preporuci se, gotovo u potpunosti, podudaraju sa predloženom klasifikacijom: „Akademske zahteve iz oblasti računarskih nauka“ su ekvivalentni Informatičkim (domenskim) znanjima; „Akademske zahteve iz oblasti obrazovanja“ i „Opšta pedagoška znanja“ odgovaraju opštim obrazovnim i pedagoškim znanjima u definisanoj podeli; metodički kurs CSTA predloga odgovara domenskim metodičkim znanjima; „iskustvo u oblasti“ odgovara nastavnoj praksi. I preporuke prikazane u ostalim navedenim radovima saglasne su sa predloženom klasifikacijom; na primer, osnovne oblasti izraelskog predloga (disciplinarnе studije računarskih nauka, generalne pedagoške studije, CS pedagoška domenska znanja, praktikum u držanju srednjoškolske CS nastave) gotovo direktno odgovaraju klasifikaciji. Izuzetak je NCATE-ov predlog standarda gde podudaranje nije toliko očigledno. Međutim, detaljnijom analizom šire verzije dokumenta (ISTE, 2002) vidi se da kategorija „Efikasno

podučavanje i strategije učenja“ obuhvata niz tema iz nastavne prakse i domenskih metodičkih znanja, dok je posedovanje pedagoških znanja preduslov za upis opisanog programa za pripremu nastavnika. Sa druge strane, u prikazanim preporukama i literaturi često nisu obuhvaćeni kursevi koji bi se svrstali u opšta znanja. Ipak, po autoru ovog rada, ovu oblast je potrebno posebno izdvojiti budući da analizirani kurikulumi neretko sadrže kurseve (na primer, matematika, fizika, jezici) koje nije moguće svrstati ni u jednu od četiri preostale kategorije.

2.7. Analiza kurikuluma za nastavnike informatike

Pri analizi kurikuluma za nastavnike informatike primarni cilj u disertaciji je bio identifikovanje kompetencija koje budući nastavnik stiče okončanjem studijskog programa. Stoga su izabrani kurikulumi analizirani tako što su oblasti i kursevi koje sadrže svrstani u neku od pet predloženih kategorija znanja, opisanih u sekciji 2.6. Da bi se utvrdile oblasti obuhvaćene studijskim programima i izvršilo kategorisanje kurseva, detaljno je analiziran ishod i sadržaj celokupnog kurikuluma, kao i ishod i sadržaj svakog kursa u kurikulumu. Informacije su preuzete sa zvaničnih sajtova fakulteta. Posebno su opisani preduslovi za pohađanje određenog studijskog programa, ukoliko postoje, kao i nivo studija na kom se realizuje program.

U prvom delu ove sekcije analizirano je da li se završetkom studija stiču domenska znanja potrebna za držanje nastave oblasti predviđenih ACM-ovim standardom. Analizom je procenjen stepen zastupljenosti domenskih znanja predviđenih modelom kurikuluma ACM K12 CS u ishodima i sadržajima analiziranih kurikuluma. Budući da su u ovom radu analizirani visokoškolski studijski programi za nastavnike informatike u srednjim školama, poređenje je izvršeno, pre svega, sa sadržajima drugog i trećeg (3.A. i 3.B) nivoa poslednje revizije ACM K12 standarda (CSTA Standards Task Force, 2011). Drugi nivo “*Computer Science and Community*” je namenjen “mlađim srednjoškolskim učenicima”. Nivo “*Computer Science in the Modern World*” (3.A.) namenjen je učenicima 9. ili 10. stepena (razreda) i obavezan je za sve đake u srednjim školama. Ovaj nivo predstavlja uslov za sledeća dva nivoa: 3.B i 3.C. Nivo “*Computer Science Concepts and Practices*” (3.B.) preporučuje se za učenike viših razreda srednje škole. “*Topics in Computer Science*” (3.C.) obezbeđuje produbljivanje znanja u jednoj određenoj oblasti računarstva i to mogu biti projekat bazirani kursevi ili kursevi obezbeđeni od strane firmi iz CS oblasti koji vode ka profesionalnoj računarskoj sertifikaciji.

U drugom delu ove sekcije analizirani su odabrani nastavnički kurikulumi iz aspekta nedomenskih znanja. Ova analiza ne sadrži stepen zastupljenosti znanja, već daje samo listu kurseva u nastavničkom kurikulumu kategorisanih u odgovarajuću oblast. Razlog za to je nepostojanje standarda za nedomenska znanja

uporedivih po nivou detaljnosti sa domenskim standardom ACM K12 CS.

2.7.1. Usaglašenost domenskih znanja u nastavničkim kurikulumima sa ACM K12 modelom

U ovoj disertaciji su oblasti koje, po ACM K12 standardu, student može da izučava u srednjoj školi podeljene u 13 kategorija na osnovu nivoa opisanih u (CSTA Standards Task Force, 2011), i nivoa opisanih u (Frost et al., 2009; Madden et al., 2007; Verno et al., 2006). Oblasti i njima pripadajuće najvažnije teme prikazane su u tabeli 1.

<p>Principi organizacije računara</p> <ul style="list-style-type: none"> • Arhitektura računara (ulaz, izlaz, registri, ALU, CPU, memorija, portovi, magistrale, performanse računara) • Logička kola • Kriterijumi za kupovinu računara • Uobičajeni PC problemi i rešenja 	<p>Multimedija</p> <ul style="list-style-type: none"> • Vektorska i bit-mapirana grafika (razlike, prednosti i mane, kreiranje, snimanje i editovanje) • Konverzija formata slike i kompresija • Digitalno "slikanje" i importovanje multimedije u dokumente i na web strane • Fajl formati za video, audio i slike • Potreban hardver i softver za rad sa multimedijom
<p>Algoritmi i rešavanje problema</p> <ul style="list-style-type: none"> • Algoritmi (definicija, tipovi, algoritmi pretraživanja i sortiranja i sl.) • Proces rešavanja problema • Problem dizajn metodologija ("od vrha ka dnu", "od dna ka vrhu", kombinacija pristupa) • Alati za (vizuelni) dizajn problema (dijagrami toka, UML, pseudokod i sl.) • Računarska ograničenja i kompjuterski "intenzivni"/teški/nerešivi problem • Ocena efikasnosti, tačnosti i jasnosti algoritama • Pojam paralelnog procesiranja (deljenje procesa u niti) 	<p>Programiranje i principi softverskog inženjerstva</p> <ul style="list-style-type: none"> • Tipovi podataka (primitivni tipovi, vektori, nizovi, liste, stek, apstraktni tipovi podataka) • Karakteristike i razlike između jezika visokog nivoa i asemblerskih jezika • Strukturano programiranje • Objektno-orijentisano programiranje (primena API-ja, biblioteka klasa, interaktivno programiranje i sl.) • Razvojna okruženja (IDE) • Karakteristike i razlike byte-code generatora, kompajlera i interpretera • Principi softverskog inženjerstva (faze razvoja softvera, timski rad, primena softvera za modelovanje procesa i sl.)

<p>Operativni sistemi (OS)</p> <ul style="list-style-type: none"> • Uloga operativnih sistema • Fajl sistem • Upravljanje eksternim uređajima • Multi-tasking • Karakteristike i poređenje različitih OS • Svojstva OS koja omogućuju pristup hardveru (sistemske pozivi i sl.) 	<p>Aplikacije</p> <ul style="list-style-type: none"> • Standardan softver (tekst editori, softverski alati za rad sa tabelama, aplikacije za rad sa prezentacijama) • Aplikacije za upravljanje relacionim bazama podataka (kreiranje i održavanje) • Upiti nad bazama podataka i SQL • Modelovanje i simulacije • Obrazovanje i trening (tutorski softver, obrazovna okruženja i sl.) • Mobilne aplikacije
<p>Računarske mreže</p> <ul style="list-style-type: none"> • Komponente računarske mreže i hardverski i softverski zahtevi • Topologija mreže • Mrežni protokoli • Karakteristike koje utiču na funkcionalnost mreže (propusni opseg, kašnjenje, firewall) • LAN i WAN • Deljenje resursa i rad u mreži • Bezbednost računarskih mreža • Klijent-server i peer-to-peer strategija • Mobilno računarstvo 	<p>Veza između matematike i računarskih nauka</p> <ul style="list-style-type: none"> • Digitalna reprezentacija podataka i kodiranje • Brojni sistemi (i konverzija između sistema) • Binarni brojevi (i operacije nad njima) • Bulova algebra • Tablice istinitosti • Funkcije, skupovi i operacije sa skupovima • Koncepti grafova
<p>Internet koncepti</p> <ul style="list-style-type: none"> • WWW i URL • Internet servisi • Pretraživači i pretraživanje Interneta • Asinhrona i sinhrona komunikacija • Društvene mreže • Internet bezbednost • Ocena web izvora po njihovoj relevantnosti i pouzdanosti 	<p>Dizajn web stranica i razvoj</p> <ul style="list-style-type: none"> • HTML i CSS • Dizajniranje, kreiranje, objavljivanje i ocenjivanje "user-friendly" web site-a • Alati za web razvoj • Interaktivni i statički web sajtovi • Dizajn i razvoj mobilnih računarskih aplikacija • Osnove "interakcije čovek računar"(HCI)
<p>Primene veštačke inteligencije i modeli inteligentnog ponašanja</p> <ul style="list-style-type: none"> • Obrada prirodnog jezika • Sistemi zasnovani na znanju • Mašinsko učenje • Računarske igre • Primeri aplikacija za prepoznavanje govora i "Computer vision" sistema • Osnovne aplikacije veštačke inteligencije i osnove robotike 	<p>Etička pitanja i pitanja bezbednosti</p> <ul style="list-style-type: none"> • Virus i maliciozan softver • Bezbedno korišćenje računara i zaštita uključujući enkripciju, kriptografiju i autentifikaciju • Prava intelektualne svojine • Odgovorno korišćenje softvera i softverska piraterija • Privatnost i deljenje informacija • Komercijalni softver, softver na javnim domenima, "open source" i "shareware" softver

Računari u društvu (Interdisciplinarna priroda računarstva)

- Dinamičnost CS oblasti i (pozitivan i negativan) uticaj na obrazovanje, društvo i zaposlenje
- Uloga tehnologije u obrazovanju dece sa posebnim potrebama
- Primena računara pri donošenju odluka (primeri ekspertnih sistema, baza znanja i sl.)
- Uticaj računarstva na e-trgovinu
- Primeri "ugrađenih" sistema (automobili, avioni i sl.)
- Istorija računarstva
- Karijere u računarstvu

Tabela 1. Srednjoškolski informatički sadržaji po ACM K12 CS modelu kurikuluma (CSTA Standards Task Force, 2011; Frost et al., 2009; Madden et al., 2007; Verno et al., 2006)

Pri utvrđivanju da li studijski program sadrži sve kategorisane oblasti ACM-ovog standarda, uzete su u obzir moguće situacije po kojima se jedna oblast izučava u različitim kursevima ili više oblasti u istom kursu. Na osnovu podataka dostupnih na sajtovima institucija za svaku oblast procenjena je zastupljenost pripadajućih tema, jednostavnim prebrojavanjem navedenih tema i računanjem procentualne zastupljenosti. Pri tome treba, svakako, imati u vidu da nisu svi kurikulumi opisani sa istim nivoom detaljnosti, odnosno negde su precizno nabrojane sve teme koje se izučavaju, a ponegde samo opšte kompetencije.

U naredna dva odeljka prikazani su rezultati analize odabranih kurikuluma grupisanih po zemljama u kojima se studije realizuju i modalitetima studiranja. Obuhvaćene su sledeće zemlje: Nemačka, Izrael, Turska, Škotska, SAD (Texas), Estonija, Austrija, Republika Srbija i sledeći modaliteti studiranja: Bečelor i master studije (BSC+MSC), samo master studije (MSC) u trajanju od 2 ili 4 semestra, i samo bečelor studije (BSC). U većem delu rada fakulteti/univerziteti su označeni skraćenim oznakama definisanim u Prilogu 1. Kurikulumi su odabrani prvenstveno na osnovu dostupnosti na Internetu i detaljnosti opisa kurseva koje sadrže. To se, pre svega, odnosi na brojne studijske programe iz Nemačke koji su veoma detaljno opisani. Takođe, na pojedinim univerzitetima u Nemačkoj postoje odvojene studije za nastavnike informatike za svaki tip škole (detaljnije u sekciji 2.7.1.1.). Međutim, postojali su i drugi kriterijumi: država Texas je izabrana jer, po (Wilson et al., 2010), ima „dugu istoriju CS obrazovanja u srednjoj školi“ koje uvažava aktuelne CS standarde; s obzirom na brojnost i relevantnost naučnih radova izraelskih autora iz oblasti CS nastavničkih kurikuluma, bilo je važno proučiti i jedan kurikulum iz te zemlje; izabrani kurikulumi u Estoniji i Turskoj sadrže interesantne predloge za kurseve koji integrišu CS domen i obrazovnu primenu itd.

2.7.1.1. Analiza svetskih kurikuluma

U Nemačkoj postoji veliki broj univerziteta na kojima se realizuju studijski programi za nastavnike Informatike. Za potrebe ovog rada analizirani su kurikulumi sa sedam fakulteta/univerziteta: Fakultät für

Informatik, Technische Universität Dortmund (Fakultät für Informatik, 2013); Fakultät für Wirtschaftswissenschaften, Universität Duisburg-Essen (Fakultät für Wirtschaftswissenschaften, 2012); Naturwissenschaftlich-Technische Fakultät, Universität Siegen (Naturwissenschaftlich-Technische Fakultät, 2012); Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn (Fakultät für Elektrotechnik, Informatik und Mathematik, 2013); Goethe Universität (Goethe-Universität, 2013); Universität Rostock (Universität Rostock, 2013); Universität Potsdam (Universität Potsdam, 2012). Jedna važna specifičnost nemačkih nastavničkih kurikuluma je postojanje odvojenih programa za nastavnike informatike u gimnazijama i nastavnike informatike u osnovnim, opštim i „sveobuhvatnim“ školama. Postoje i posebni programi za nastavnike informatike u srednjim stručnim školama (GER1, GER4). U disertaciji su prikazani rezultati uporedne analize programa namenjenih budućim gimnazijskim nastavnicima. U tabeli 2, od 7 analiziranih studijskih programa iz Nemačke, prikazana je analiza tri kurikuluma koji sadrže najdetaljniji opis kurseva. Isti kriterijum je primenjen pri izboru kurikuluma za prikaz u tabelama 3, 4, 5 i 6.

Na analiziranim nemačkim fakultetima studenti već na bečelor nivou izučavaju paralelno domenske i pedagoške kurseve, a na master nivou se stečena znanja iz svih oblasti unapređuju kroz izučavanje naprednijih tema. Posebno se naglašavaju praktičan rad studenata u saradnji sa centrima za obrazovanje i metodologija naučnog istraživačkog rada. U tabelama 2, 3, 4, 5 i 6 prikazana je usaglašenost domenskih sadržaja sa oblastima i temama ACM K12 CS modela kurikuluma. Kolona „Oblast“ ispod odgovarajućeg studijskog programa, sadrži informaciju o tome da li kurikulum predviđa izučavanje oblasti (Da/ Ne), dok kolona *Teme (%)* prikazuje podatke o tome koji procenat tema određene oblasti, definisanih u tabeli 1, je naveden u kurikulumu. U analizi sadržaja kurikuluma uzeti su u obzir i izborni kursevi.

Oblast	Univerzitet/nivo studija					
	GER1/ BSC + MSC		GER2/ BSC + MSC		GER3/ BSC + MSC	
Oblast	Oblast	Teme (%)	Oblast	Teme (%)	Oblast	Teme (%)
Principi organizacije računara	Da	75%	Da	75%	Da	75%
Multimedija	Da	100%	Ne	0%	Ne	0%
Algoritmi i rešavanje problema	Da	100%	Da	100%	Da	86%
Programiranje i principi softverskog inženjerstva	Da	86%	Da	71%	Da	71%
Operativni sistemi	Da	100%	Da	100%	Da	83%
Aplikacije	Da	50%	Da	85%	Da	67%
Računarske mreže	Da	78%	Da	100%	Da	44%
Veze između matematike i računarskih nauka	Da	57%	Da	86%	Da	71%

Oblast	Univerzitet/nivo studija					
	GER1/ BSC + MSC		GER2/ BSC + MSC		GER3/ BSC + MSC	
	Oblast	Teme (%)	Oblast	Teme (%)	Oblast	Teme (%)
Internet koncepti	Ne	0%	Ne	0%	Ne	0%
Dizajn web stranica i razvoj	Da	83%	Da	67%	Ne	0%
Primene veštačke inteligencije i modeli inteligentnog ponašanja	Da	16%	Da	33%	Da	16%
Etička pitanja i pitanja bezbednosti	Da	100%	Da	17%	Da	17%
Računari u društvu	Da	86%	Da	43%	Da	57%

Tabela 2. Kurikulumi u Nemačkoj

Detaljnou analizom svih sedam nemačkih kurikuluma može se zapaziti da oni ne predviđaju izučavanje “jednostavnijih” bazičnih tema informatičkog kurikuluma. Nijedan od njih ne obuhvata teme koje se odnose na standardan softver za rad sa tekstom, slikom i prezentacijama. Sadržaji koji se odnose na temu “Obrazovanje i trening” (tutorski softverski sistemi za e-učenje, standardi i sl.) se eksplicitno navode samo u GER3. Principi Interneta takođe skoro nigde nisu zastupljeni kao ni tema „Kriterijumi za kupovinu računara”. Međutim, može se pretpostaviti da tvorci nemačkih kurikuluma očekuju od studenata da su te teme već izučili na prethodnom nivou školovanja. Sa druge strane, detaljnom analizom svih nemačkih kurikuluma može se zaključiti da oni podrazumevaju izučavanje naprednih informatičkih koncepata. Tako, uvidom u sadržaj kurseva, na primer, GER3, može se uočiti da se već na bečelor nivou dublje izučavaju C++ i Java programski jezici, modelovanje primenom UML-a, relacione baze podataka, SCORM i LOM standardi, “ugrađeni” sistemi itd. Na master nivou se izučavaju objektna i XML baze podataka, distribuirano programiranje itd. Takođe je zanimljivo da svi kurikulumi u nekom obliku (poseban kurs ili u okviru drugog kursa) predviđaju izučavanja principa softverskog inženjerstva. Ipak, ozbiljnija primedba bi se mogla odnositi na malu zastupljenost oblasti kao što su:

- veštačka inteligencija (obično su to samo teme vezane za Semantički Web i sisteme zasnovane na znanju),
- multimedije i kompjuterske grafike (samo u GER1 i GER6 su predviđene ove teme i to u vidu izbornih kurseva).

Takođe, ne izučavaju se sadržaji vezani za mobilne aplikacije. U kurikulumima se neretko može zapaziti odsustvo cele oblasti “Dizajn web stranica i razvoj”. Iz tabele 2 je uočljivo i da nedostaju mnoge teme oblasti „Etička pitanja i pitanja bezbednosti“ predviđene ACM K12 standardom. Ipak, niži procenat usaglašenosti nekih oblasti sa ACM K12 temama, poput „računarskih mreža“ u GER3 (tabela 2), ne znači nužno da mnogi važni principi te oblasti nisu kvalitetno

obrađeni. Tako kursevi računarskih mreža na ovom fakultetu detaljno izučavaju protokole vezane za sve OSI nivoe, kao i mrežno programiranje.

U Austriji su mogući različiti modaliteti obrazovanja budućih nastavnika informatike. Poput nemačkih kurikuluma, na Tehničkom Univerzitetu u Gracu (Technische Universität Graz, 2013) postoje studije nastave informatike i informatičkog menadžmenta organizovane kao bečelor i master studije u trajanju od 10 semestara. U okviru ovih studija se već na bečelor nivou izučavaju i domenski i pedagoški principi. Studije su namenjene pre svega nastavnicima u srednjim i višim školama. Na Univerzitetu u Salzburg-u (Universität Salzburg, 2013) postoje bečelor studije nastave informatike i informatičkog menadžmenta u trajanju od 8 semestara.

Na Tehničkom Univerzitetu u Beču (Technische Universität Wien, 2012) postoje master studije didaktike informatike u trajanju od 4 semestra, namenjene studentima koji žele da rade kao nastavnici u srednjim opštim i srednjim stručnim školama, kao informatički instruktori u ekonomiji ili kao instruktori za profesionalno usavršavanje i doživotno učenje u informatičkim profesijama. Uslov za upis studija su završene tehničke studije bečelor nivoa (pre svega, poslovna informatika ili računarstvo).

Na Univerzitetu u Talinu, Estonija (Tallinn University, 2013) takođe postoje master studije za nastavnike računarstva i IKT menadžere u trajanju od 4 semestra. Uslov za upis je bečelor nivo iz oblasti računarskih nauka.

Izraelski nastavnici moraju ispuniti rigorozan set kriterijuma koji uključuje osnovne CS studije (ili položen kvalifikacioni ispit iz odgovarajućih CS oblasti) i diplomske studije za pripremu nastavnika. U (Gal-Ezer & Zur, 2007) je predstavljen program udaljenog učenja na Open University Izrael (ISR) za pripremu CS nastavnika koji obezbeđuje sertifikaciju.

Kao ilustracija master programa u trajanju od 4 semestra, u tabeli 3 prikazane su zastupljenosti domenskih znanja MSC programa AUS3 i EST1.

Uočljivo je da kurikulumi strukturirani kao master studije, poput AUS3 i EST1, obuhvataju nedovoljno važnih domenskih znanja predviđenih ACM K12 modelom kurikuluma i to ih uglavnom nude kroz izborne kurseve.

Slična je situacija i na master studijama - ISR1 koji nije prikazan u tabeli. Ove studije predviđaju samo dva domenska kursa:

- Algoritmi: Osnova računarskih nauka,
- Teme u računarskom obrazovanju.

	Univerzitet/nivo studija			
	AUS3/ MSC (4 semestra)		EST1/ MSC (4 semestra)	
Oblast	Oblast	Teme (%)	Oblast	Teme (%)
Principi organizacije računara	Ne	0%	Ne	0%
Multimedija	Da	100%	Da	20%
Algoritmi i rešavanje problema	Da	71%	Ne	0%

Oblast	Univerzitet/nivo studija			
	AUS3/ MSC (4 semestra)		EST1/ MSC (4 semestra)	
	Oblast	Teme (%)	Oblast	Teme (%)
Programiranje i principi softverskog inženjerstva	Da	75%	Ne	0%
Operativni sistemi	Ne	0%	Da	17%
Aplikacije	Da	17%	Da	33%
Računarske mreže	Da	11%	Ne	0%
Veze između matematike i računarskih nauka	Da	22%	Ne	0%
Internet koncepti	Da	29%	Da	29%
Dizajn web stranica i razvoj	Ne	0%	Da	17%
Primene veštačke inteligencije i modeli inteligentnog ponašanja	Da	11%	Ne	0%
Etička pitanja i pitanja bezbednosti	Ne	0%	Da	17%
Računari u društvu	Da	14%	Da	14%

Tabela 3. MSC kurikulumi za nastavnike informatike u Austriji i Estoniji

Na analiziranim master studijama se naglašava značaj multimedije i e-učenja i studije su uglavnom koncipirane tako da su informatički sadržaji, već usvojeni na prethodnom nivou studija, prilagođeni profesiji nastavnika informatike. To posebno važi za EST1 čija se specifičnost u odnosu na većinu analiziranih kurikuluma u ovom radu ogleda u izučavanim temama informatičkih kurseva. U njemu se ne izučavaju baze podataka, programerske paradigme i slično, već teme obuhvaćene kursevima poput: Instrukcioni dizajn, Principi e-učenja, Dizajn obrazovnog softvera zasnovanog na multimediji, web bazirani alati za učenje, „Open source“ softverski razvojni modeli itd.

S obzirom na opisanu strukturu analiziranih master studija, iz tabele 3 nije moguće zaključiti da li će studenti po okončanju ovog nivoa studija imati dovoljna domenska znanja koja bi im omogućila da predaju u školama koje su koncipirale kurikulume u skladu sa ACM-ovim predlogom. Domenske kompetencije svršenih studenata gotovo isključivo zavise od znanja stečenih na osnovnim studijama.

Za razliku od mnogih američkih država (Wilson et al., 2010), u američkoj državi Teksas postoje jasni uslovi za nastavnike informatike: potrebno je da nastavnici imaju CS 8-12 K12 sertifikat. Jedan od koledža u toj američkoj državi sa studijskim programom za CS nastavnike koji ispunjava propisane uslove je College of Science and Technology (College of Science and Technology, 2013). Nakon završetka ovih studija stiže se bečelor zvanje i sertifikat za predavanje računarskih nauka u srednjoj školi (nivo 8-12).

U Škotskoj se kombinovane studije za obrazovanje srednjoškolskih nastavnika računarstva sastoje od: profesionalnih (obrazovnih) studija, studija računarskih nauka i nastavne prakse. Takođe, predavanje računarstva u srednjim školama je omogućeno i kandidatima koji poseduju diplomu iz računarstva plus odgovarajuću obrazovnu diplomu (Professional Graduate Diploma in Education –

PGDE koju obezbeđuju Univerziteti u Glazgovu i Stirlingu). Ovde je neophodan uslov da polovina od potrebnih kreditnih poena bude najmanje iz dve od četiri oblasti: 1) kompjuterski sistemi, 2) razvoj softvera, 3) baze podataka ili 4) informacioni sistemi. Drugu polovinu kredita je neophodno ostvariti iz bilo koje oblasti računarstva relevantne za aktuelni računarski kurikulum u škotskim školama. U ovom radu su analizirane kombinovane studije na Univerzitetu u Stirlingu, koji je po „*The Times Good University Guide*“ i „*The Guardian University Guide*“ „prvi u obrazovanju u Škotskoj“ (University of Stirling, 2013). Ove studije traju 4 godine i obezbeđuju „honours“ stepen u izabranoj temi (računarske nauke) i profesionalnoj edukaciji.

I u Turskoj, na Departmanu za računarsko obrazovanje i instrukcionu tehnologiju (CEIT) „Meadle East Technical“ Univerziteta (Middle East Technical University, 2013), studije traju 8 semestara i student po okončanju studija može da predaje u osnovnoj i srednjoj školi.

U tabelama 4 i 5 prikazana je zastupljenost domenskih znanja kurikuluma koncipiranih kao bečelor studije (USA1, SCO1, TUR1, AUS1). U UK su u (Teacher Agency, 2012) od strane nadležne nastavničke agencije definisana domenska znanja koja budući nastavnici računarstva treba da poseduju pre nego što počnu da pohađaju „treening za nastavnike računarskih nauka“. Stoga analiza domenskih znanja SCO1 u tabeli 4 zapravo pokazuje u kojoj meri su potrebna domenska znanja propisana od nadležne institucije u UK (Teacher Agency, 2012) usaglašena sa oblastima i temama ACM K12 kurikuluma.

Oblast	Univerzitet/nivo studija			
	USA1/ BSC		SCO1/ „Hon“ (8 semestara)	
	Oblast	Teme (%)	Oblast	Teme (%)
Principi organizacije računara	Da	25%	Da	75%
Multimedija	Ne	0%	Da	40%
Algoritmi i rešavanje problema	Da	43%	Da	71%
Programiranje i principi softverskog inženjerstva	Da	71%	Da	86%
Operativni sistemi	Da	100%	Da	67%
Aplikacije	Da	33%	Da	50%
Računarske mreže	Da	11%	Da	56%
Veze između matematike i računarskih nauka	Da	14%	Da	57%
Internet koncepti	Ne	0%	Da	43%
Dizajn web stranica i razvoj	Da	83%	Da	33%
Primene veštačke inteligencije i modeli inteligentnog ponašanja	Ne	0%	Ne	0%
Etička pitanja i pitanja bezbednosti	Da	17%	Da	40%
Računari u društvu	Da	29%	Da	50%

Tabela 4. BSC i dipl. kurikulumi u SAD i Škotskoj

Oblast	Univerzitet/nivo studija			
	TUR1 / Dipl (8 semestara)		AUS1/ BSC (8 semestara)	
	Oblast	Teme (%)	Oblast	Teme (%)
Principi organizacije računara	Da	75%	Da	50%
Multimedija	Da	60%	Da	60%
Algoritmi i rešavanje problema	Da	29%	Da	29%
Programiranje i principi softverskog inženjerstva	Da	57%	Da	71%
Operativni sistemi	Da	100%	Da	67%
Aplikacije	Da	67%	Da	67%
Računarske mreže	Da	44%	Da	11%
Veze između matematike i računarskih nauka	Da	14%	Da	29%
Internet koncepti	Da	57%	Da	14%
Dizajn web stranica i razvoj	Da	17%	Da	17%
Primene veštačke inteligencije i modeli inteligentnog ponašanja	Ne	0%	Ne	0%
Etička pitanja i pitanja bezbednosti	Da	17%	Da	17%
Računari u društvu	Da	43%	Da	14%

Tabela 5. BSC i dipl. kurikulumu u Turskoj i Austriji

Svim kurikulumima prikazanim u tabelama 4 i 5, u odnosu na ACM K12 preporuke, nedostaju principi veštačke inteligencije. Takođe, može se zapaziti da kurikulum u USA1 ne sadrži i neke druge važne domenske oblasti (Multimedija, Internet koncepti), a neke koje postoje (poput računarskih mreža) nisu detaljnije opisane.

Studijski program na TUR1 je strukturiran tako da se u domenskim znanjima mogu razlikovati dve generalne oblasti: jezgro računarskih nauka (programski jezici, računarski hardver, operativni sistemi, baze podataka) i specifični obrazovni CS kursevi poput grafike i animacije u obrazovanju, obrazovni servisi, udaljeno učenje. Dakle, domenska znanja, definisana ovim kurikulumom, predstavljaju kombinaciju pristupa sa EST1 i drugih kurikuluma koji naglašavaju samo „čiste“ računarske kurseve. Međutim, detaljnom analizom kurikuluma TUR1, opisanom na zvaničnom sajtu ustanove, uočljivo je da i ovde nedostaju mnoge teme predviđene ACM K12 modelom kurikuluma. To su teme poput mobilnih aplikacija, važne teme iz oblasti algoritama, tema koje se odnose na dizajn web stranica, itd.

Preporuke ekspertske grupe u UK o potrebnim domenskim znanjima za studente koji upisuju „trening za nastavnike računarskih nauka“ su u velikoj meri usaglašene sa ACM K12 modelom kurikuluma. Neretko se mogu pronaći i ekvivalentni opisi očekivanih ishoda. Ipak, ovim preporukama, u odnosu na ACM sadržaje, nedostaju, uglavnom, iste one teme koje se ne pojavljuju ni u drugim analiziranim kurikulumima poput:

- kriterijuma za kupovinu računara,

- aplikacija za rad sa tekstom, tabelama, prezentacijama,
- mobilnih aplikacija,
- tema iz multimedije, računara u društvu, itd.

I na AUS1 nedostaju slične teme kao i nemačkim kurikulumima. „Jednostavniji“ koncepti aplikacija za rad sa tekstom, tabelama i prezentacijama, kao i kriterijumi za naručivanje računara, Internet koncepti i sl. nisu obuhvaćeni kurikulumom. Iz tabele 5 se može videti da su neke oblasti na AUS1 samo navedene bez eksplicitnog davanja detaljnih tema koje izučavaju. Tako, u ovom kurikulumu eksplicitno nisu navedene većine tema oblasti „Računarske mreže“, „Algoritmi i rešavanje problema“, „Dizajn web strana i razvoj“, „Etička pitanja i pitanja bezbednosti“ i „Računari u društvu“.

2.7.1.2. Analiza kurikuluma iz Republike Srbije

U Republici Srbiji više fakulteta/univerziteta (Prirodno-matematički fakulteti u Novom Sadu (Prirodno-matematički fakultet Novi Sad, 2012), Nišu (Prirodno matematički fakultet Niš, 2007), Kragujevcu (Prirodno-matematički fakultet Kragujevac, 2012); Matematički fakultet u Beogradu (Matematički fakultet, 2013); Fakultet tehničkih nauka u Čačku (Fakultet tehničkih nauka, 2013); Tehnički fakultet „Mihajlo Pupin“ u Zrenjaninu (Tehnički fakultet „Mihajlo Pupin“, 2009) i Državni Univerzitet u Novom Pazaru (Državni Univerzitet Novi Pazar, 2013)) obrazuje srednjoškolske nastavnike informatike, sa različitim modalitetima studiranja:

- samo bečelor studije (6 semestara),
- osnovne akademske studije u trajanju od 8 semestara,
- master studije (2 semestra ili 4 semestra) i
- integrisane (master) studije (10 semestara).

U tabeli 6 prikazani su studijski programi tri fakulteta u Republici Srbiji koji školuju srednjoškolske nastavnike informatike:

- 1) studijski program „Informacione tehnologije – Informacione tehnologije u obrazovanju“ Tehničkog fakulteta „Mihajlo Pupin“ u Zrenjaninu koji daje zvanje inženjer informacionih tehnologija – profesor informatike i tehnike,
- 2) integrisane akademske studije „Tehnika i informatika“ Fakulteta tehničkih nauka u Čačku koje daju zvanje diplomirani profesor tehnike i informatike-master, i
- 3) dvopredmetne integrisane master studije Prirodno-matematičkog fakultetu u Novom Sadu (modul matematika i informatika) koji daje zvanje master profesor dvopredmetnih studija.

Pored toga, budući nastavnici se mogu edukovati i u okviru jednogodišnjih (na primer, studijski program „*Profesor matematike i računarstva*“ Matematičkog fakulteta u Beogradu) ili dvogodišnjih master studija „*Nastava informatike*” PMF-a u Novom Sadu, pri čemu

je osnovni uslov za upis na master studije završetak osnovnih akademskih studija iz oblasti računarskih nauka/informatike.

Oblast	Univerzitet/nivo studija					
	SRB1/Dipl 8 semestara		SRB2/Integrisane studije 10 semestara		SRB3/Integrisane studije 10 semestara	
	Oblast	Teme (%)	Oblast	Teme (%)	Oblast	Teme (%)
Principi organizacije računara	Da	50%	Da	25%	Da	25%
Multimedija	Da	100%	Da	40%	Da	100%
Algoritmi i rešavanje problema	Da	43%	Da	14%	Da	86%
Programiranje i principi softverskog inženjerstva	Da	57%	Da	71%	Da	100%
Operativni sistemi	Da	67%	Da	100%	Da	100%
Aplikacije	Da	83%	Da	83%	Da	67%
Računarske mreže	Da	67%	Da	100%	Da	89%
Veze između matematike i računarskih nauka	Da	100%	Da	43%	Da	100%
Internet koncepti	Da	86%	Da	86%	Da	57%
Dizajn web stranica i razvoj	Da	67%	Da	67%	Da	50%
Primene veštačke inteligencije i modeli inteligentnog ponašanja	Da	67%	Ne	0%	Da	67%
Etička pitanja i pitanja bezbednosti	Da	17%	Da	17%	Da	83%
Računari u društvu	Da	71%	Da	57%	Da	43%

Tabela 6. Izabrani kurikulumi za nastavnike informatike u Republici Srbiji

Iz tabele 6 je uočljivo da studijski program na SRB2 ne obuhvata teme iz oblasti veštačke inteligencije i, posebno, nisu eksplicitno navedene važne teme oblasti algoritmi i rešavanje problema.

Ni u SRB1 neke važne teme iz algoritamske oblasti nisu nabrojane, kao ni veliki deo tema oblasti etička pitanja i pitanja bezbednosti.

Integrisane studije SRB3 predviđaju izučavanje svih oblasti i značajnog broja tema ACM-ovog predloga kurikuluma, ali se veliki broj domenskih tema i oblasti (multimedija, računarske mreže, Internet i dizajn strana, veštačka inteligencija, principi softverskog inženjerstva) realizuje kroz izborne kurseve. Na ovaj način se može desiti da student ne prođe neke važne domenske oblasti/teme. Budući da je program osmišljen u vidu dvopredmetnih studija, domenski kursevi su ravnomerno raspoređeni između informatičkih sadržaja i sadržaja prvog predmeta izučavanja, što takođe (posebno u slučajevima geografija – informatika i fizika – informatika) može da dovede u pitanje informatičke kompetencije svršenog studenta.

2.7.2. Usaglašenost kurikuluma sa nedomenskim znanjima

U tabeli 7 su navedeni kursevi koji ne pripadaju domenu informatike, a postoje u kurikulumima prikazanim u tabelama 2, 3, 4, 5 i 6. Kursevi su klasifikovani u jednu od 4 neinformatičke kategorije predložene u sekciji 2.6 ovog rada: opšta obrazovna i pedagoška znanja, domenska metodičko-didaktička znanja, znanja nastavne prakse i opšta znanja.

Kurikulum	Oblast kojoj kurs pripada			
	Opšta obrazovna i pedagoška znanja	Domenska metodičko-didaktička znanja	Znanja nastavne prakse	Opšta znanja
GER1		Didaktika informatike; Metodika nastave informatike	Semestralna praksa	
GER2		Didaktika informatike; Planiranje kurikuluma	Praktikum iz oblasti buduće karijere; Semestralna praksa	
GER3		Didaktika informatike	Nastavna praksa	
AUS3	Pedagogija i didaktika; Komunikacije, prezentacija i moderacija; IKT u obrazovanju; Medijska didaktika	Osnove didaktike informatike; Dizajn nastave i učenja u informatičkom obrazovanju	Nastavna praksa	
AUS1		Didaktika i metodika informatike	Školska praksa	
EST1	Nastavnik u školi i društvu; Razvoj i učenje; Dizajn i razvoj učeničkog okruženja; Učenici sa posebnim potrebama; Nastavnik kao savetnik/vaspitač itd.	Didaktika informatike; Aktivno učenje u informatičkoj nastavi	Nastavna praksa	
USA1	Opšta psihologija; Škola i društvo; Upravljanje u učionici; Podučavanje studenata		Studentska praksa	Čitanje iz domenske oblasti za srednjoškolske studente; Diskretna matematika; Račun

Kurikulum	Oblast kojoj kurs pripada			
	Opšta obrazovna i pedagoška znanja	Domenska metodičko-didaktička znanja	Znanja nastavne prakse	Opšta znanja
TUR1	Uvod u prosvetu; Obrazovna psihologija; Materijali i dizajn i primena u obrazovanju; Instrukcioni dizajn; Upravljanje u učionici	Metodika nastave računarskog obrazovanja; Metodologija u oblasti specijalizacije itd.	Nastavna praksa	Jezici; Matematika; Fizika
SCO1	Obrazovanje: svrha, principi, praksa; Učenje i nastava; Pedagogija; Razlike i identiteti	Pedagogija vezana za kurikulum	Školsko iskustvo (praksa)	
SRB1	Pedagoška psihologija; Pedagogija sa didaktikom	Metodika nastave tehnike; Metodika nastave informatike	Metodička praksa	Numerička matematika; Engleski jezik
SRB2	Psihologija; Pedagogija; Obrazovna tehnologija; Dokimologija; Komunikologija	Metodika informatike i tehnike; Metodički praktikum iz osnova programiranja; Metodički praktikum iz robotike; itd.	Stručna školska praksa	Engleski jezik; Matematika; Fizika; Nacrtna geometrija; Verovatnoća i statistika
SRB3	Pedagogija; Obrazovni softver; Razvojna i pedagoška psihologija; Metodologija pedagoških istraživanja u nastavi	Metodika informatike; Metodika programiranja	Školska praksa	Analiza; Algebra; Osnovi geometrije; Metodika matematike, Numerička analiza itd.

Tabela 7. Kursevi nedomenskih oblasti

Iz tabele 7 je uočljivo da analizirani nemački kurikulumi za gimnazijskog nastavnika informatike ne sadrže posebne opšte obrazovne i pedagoške kurseve. Detaljnom analizom kurikuluma može se zaključiti da su teme iz te oblasti inkorporirane u domenske metodičke kurseve i nastavnu praksu. Slično se može zaključiti i za studijski program AUS1. Za razliku od GER1, GER2 i GER3, ostali

kurikulumi, prikazani u tabelama, predviđaju izučavanje pedagoških tema u vidu posebnih kurseva. To važi i za kurikulume koji nisu prikazani u tabelama, poput ISR1 koji sadrži opšte obrazovne i pedagoške kurseve kao što su: Dizajn kurikuluma, razvoj i implementacija, Kritičko razmišljanje, Obrazovna psihologija i Individualizovana instrukcija. U tabeli 7 čak nisu ni nabrojani svi opšti obrazovni i pedagoški kursevi studijskog programa na EST1, poput izbornih kurseva: Obrazovna Sociologija i politika, Multikulturalno učenje, itd.

Zasebni kursevi koji podrazumevaju izučavanje domenskih metodičko-didaktičkih principa postoje u svim kurikulumima analiziranim u ovom radu, izuzev kurikuluma USA1. Na osnovu dostupnog kurikuluma na zvaničnom sajtu, vidi se da teme metodičkih znanja vezane za specifičnost računarskih nauka nisu uključene ni u opšta pedagoška znanja. Interesantan predlog kurikuluma SRB2 i SRB3 je postojanje posebnog kursa metodike programiranja, što je u skladu sa sadržajem kurseva metodike nastave informatike u ISR1 i CSTA predlogu, koji posebno naglašavaju važnost izučavanja ove „podoblasti“ metodike nastave informatike.

Takođe, iz table 7 se vidi da su znanja nastavne prakse prisutna u svim kurikulumima.

Kursevi koji se mogu kategorizovati u oblast opštih znanja ne postoje u svim kurikulumima, a, ukoliko postoje, najčešće se odnose na izučavanje matematičkih principa i jezika. Nemački kurikulumi prikazani u tabelama 2 i 7 (GER1, GER2 I GER3) ne sadrže kurseve ove oblasti. Međutim, odvojeni matematički kursevi se nalaze u gotovo svim ostalim analiziranim studijskim programima u ovoj zemlji; tako GER5, GER6, GER7 predviđaju posebne matematičke kurseve. Takođe i integrisani studijski program AUS2, koji nije prikazan u tabelama, sadrži matematičke kurseve poput diskretne matematike i analize.

Kursevi koji se tiču primene IKT-a u obrazovanju su u ovom radu svrstani ili u domenska znanja i potom upoređeni sa ACM K12 sadržajima (poput informatičkih kurseva specifičnih za nastavnike informatike u TUR1) ili u opšta obrazovna i pedagoška znanja poput kurseva: IKT u obrazovanju (AUS1), Obrazovna tehnologija (SRB2), Obrazovni softver (SRB3) (tabela 7). Klasifikacija kurseva u jednu od ove dve oblasti znanja je izvršena na osnovu uvida u sadržaj kurseva. Ukoliko kurs ne predviđa dublje izučavanje principa i razvoja IKT-a, već samo podrazumeva primenu IKT-a u realizaciji nastave, tada je taj kurs klasifikovan kao deo opšte obrazovnih i pedagoških znanja. To znači da taj kurs nije specifičan za nastavnike informatike, već ga mogu sadržati i kurikulumi za nastavnike drugih domena.

2.7.3. Zaključna razmatranja

Pored preporuka (CSTA, NCATE) i opisa kurikuluma u naučnim radovima (američki, holandski, finski, izraelski kurikulumi), u disertaciji je detaljno analizirano i upoređeno 22 kurikuluma za nastavnike informatike iz 8 zemalja. 12 kurikuluma je detaljno

prikazano u tabelama. Analiza je pokazala da velika većina razmatranih kurikuluma predviđa izučavanje oblasti opštih obrazovnih i pedagoških znanja, domensko metodičko-didaktičkih znanja i znanja nastavne prakse i to

- najčešće u obliku posebnih kurseva koji se precizno mogu klasifikovati u odgovarajuću oblast ili
- kroz kurseve koji pripadaju bliskoj oblasti.

Tako:

- Samo jedan od svih analiziranih kurikuluma ne uključuje eksplicitno kurs iz oblasti domenskih metodičkih znanja;
- Oblast nastavna praksa, koja najčešće podrazumeva i istraživanje, sastavni je deo svih analiziranih kurikuluma;
- Opšta znanja, ukoliko postoje, obuhvataju matematičke kurseve i ponegde strane jezike.

S druge strane, studijski programi u GER1, GER2, GER3 i AUS1 ne predviđaju izučavanje posebnih kurseva iz opšte pedagoške oblasti ali izučavaju te teme, posredno, kroz domenske metodičke kurseve i nastavnu praksu.

Takođe, veoma je bilo važno odgovoriti na pitanje: koje informatičko (domensko) znanje kurikulum treba da sadrži. Jedan od načina utvrđivanja odgovora na to pitanje, realizovan u ovoj disertaciji, je upoređivanje domenskih (informatičkih) sadržaja odabranih kurikuluma sa ACM K12 CS modelom sa ciljem da se oceni da li student stiče informatičke kompetencije za srednjoškolske kurikulume organizovane u skladu sa ACM K12 modelom kurikuluma. Analiza je pokazala da većina analiziranih programa sadrži kurseve u kojima se detaljno izučavaju:

- algoritmi i rešavanja problema pomoću računara,
- strukture podataka i programske paradigme (proceduralna i objektna, pre svih),
- baze podataka,
- modelovanje i simulacija,
- principi softverskog inženjerstva,
- arhitektura računara,
- operativni sistemi,
- protokoli računarskih mreža.

U slučaju kurikuluma koncipiranih samo kao master studije, pretpostavlja se da su važna domenska znanja stečena na osnovnim studijama.

Ipak, ako se detaljno uporede sadržaji kurikuluma sa temama i oblastima ACM K12 modela može se uočiti da su mnoge teme i oblasti retko ili nikako prisutne. Na primer, kreiranje mobilnih računarskih aplikacija, bitne teme iz oblasti računarskih mreža, oblast multimedije (i kompjuterske grafike), dizajn web strana i razvoj, mogućnosti primene veštačke inteligencije i modeli inteligentnog ponašanja, aplikacije vezane za upravljanje e-učenjem, teme oblasti

računari u društvu i etička pitanja i pitanja bezbednosti, karijere u računarstvu. Stoga je mnoge od kurikuluma potrebno unaprediti da bi bili usklađeni sa ACM-ovim modelom.

Među kurikulumima postoje i razlike u strukturi domenskih znanja koja se izučavaju. Iako preovlađuje pristup po kome nastavnici informatike treba da znaju sve navedene informatičke teme, kurikulum na EST1, na primer, predviđa izučavanje samo izbornih informatičkih kurseva usko vezanih za primenu u obrazovanju (uslov za upis je bečelor nivo iz računarstva). S druge strane, na TUR1 se izučavaju paralelno CS kursevi i CS kursevi specifični za obrazovnu primenu. Po mišljenju autora ovog rada, **nastavnički kurikulumi treba da budu bazirani na opšte prihvaćenom standardizovanom domenskom modelu srednjoškolskog kurikuluma (u ovom trenutku je to ACM K12 model), uz paralelno izučavanje CS kurseva i CS kurseva specifičnih za obrazovnu primenu radi formiranja strukture domenskih znanja koja najbolje odgovaraju zahtevima nastavničke profesije.**

Analizirani kurikulumi se razlikuju i po modalitetima studiranja. Tako postoje *dva osnovna modaliteta* organizovanja studija:

- (1) studije u trajanju od 8 ili 10 semestara kojima odgovara većina kurikuluma prikazanih u ovom radu.
- (2) posebne master studije (EST1, ISR1, AUS3 i pojedini master studijski programi na SRB1, SRB3 i SRB4) koje najčešće mogu da pohađaju studenti koji poseduju bečelor diplomu iz domenske oblasti.

Prvi modalitet studiranja zagovara i Donaldsonov izveštaj (Donaldson, 2011) koji predlaže da se na nacionalnom nivou primeni model obrazovanja nastavnika Univerziteta u Stirlingu na kom se realizuju kombinovane studije u trajanju od 8 semestara. U (Grgurina, 2008) ističe se da *drugi modalitet* nosi i značajan rizik za stvaranje novog nastavničkog kadra zbog toga što studenti koji su završili studijske programe informatičkog usmerenja nisu zainteresovani da se dalje školuju za nastavnika informatike.

Stručno zvanje koje nastavnik informatike može steći po završetku studija je najmanje “diplomirani”/“honours“ (SCO1, TUR1, SRB1) a na ostalim, osim programa USA1 i AUS1 gde se stiče zvanje bečelor, stiče se master zvanje. Vredno je pomenuti i postojanje niza studijskih programa na nemačkim univerzitetima prilagođenih tipu škole u kojima će budući nastavnici predavati (gimnazije i opšte škole; osnovne, opšte i sveobuhvatne škole; stručne škole). Stav autora ovoga rada je **da prvi modalitet studiranja, uz obavezno permanentno usavršavanje i periodičnu resertifikaciju (posebno u informatičkom domenu), predstavlja najprikladniji modalitet obrazovanje novih nastavnika informatike. Drugi modalitet, uz pažljivo prilagođavanje kurikuluma prethodnim znanjima polaznika, može da bude alternativa za usavršavanje postojećeg nastavničkog kadra i doškolovanje IKT profesionalaca za nastavničke poslove. Model nemačkih univerziteta, koji uvažava tip srednje škole u kojoj će nastavnik predavati, otvara**

moćnosti kvalitetne refleksije specifičnosti informatičkih programa u različitim tipovima srednje škole na nastavničke kurikulume.

Poglavlje III

3. Istraživanje stavova nastavnika informatike o kurikulumu

U prethodnom poglavlju disertacije stanje u oblasti obrazovanja srednjoškolskih nastavnika informatike je identifikovano analizom relevantne literature, preporuka međunarodnih organizacija i reprezentativnih kurikuluma. S druge strane, aktuelno stanje u oblasti srednjoškolske nastave informatike je u velikoj meri određeno kompetencijama kojima raspolažu sadašnji (zaposleni) nastavnici informatike u srednjim školama. Jedan od načina da se utvrde kompetencije koje poseduju nastavnici je da se identifikuje njihovo formalno i dodatno obrazovanje kao i sklonost ka permanentnom obrazovanju.

Takođe, da bi se dublje analizirali kurikulumi za nastavnike informatike i srednjoškolski informatički kurikulumi u Republici Srbiji, važno je istražiti i stavove nastavnika informatike o aktuelnim kurikulumima. Preciznije, potrebno je istražiti stavove nastavnika o tome koje su teme *nastavničkih kurikuluma* značajne za realizovanje nastave u srednjoj školi, kao i u kojoj meri su aktuelne informatičke teme zastupljene u *srednjoškolskim informatičkim kurikulumima* po kojima oni drže nastavu.

Stoga je u disertaciji izvršeno istraživanje nad aktuelnim nastavnicima informatike u srednjim školama.

3.1. Cilj istraživanja

Konkretan cilj istraživanja je bio da se, kroz istraživanje objektivnih pokazatelja vezanih za akademsko i permanentno obrazovanje, subjektivnih stavova nastavnika o značaju domenskog i pedagoško-metodičkog korpusa i njihovoj zastupljenosti u kurikulumu obrazovanja nastavnika informatike, te ocene o sopstvenoj osposobljenosti, formira profil nastavnika koji bi poslužio kao jedan izvor informacija za planiranje modaliteta obrazovanja postojećeg i budućeg nastavničkog kadra za oblast informatike.

Profil nastavnika opisan je sledećim atributima:

- Akademsko (formalno) obrazovanje,
- Nastavničko iskustvo,
- Ocena osposobljenosti u domenskoj (informatičkoj) oblasti i pedagoško-metodičkom korpusu,
- Permanentno obrazovanje,
- Stav o značaju domenskog i pedagoško-metodičkog korpusa u kurikulumu obrazovanja nastavnika informatike,
- Stav o zastupljenosti domenskih znanja u kurikulumu srednjoškolskog obrazovanja,
- Stav o modalitetima obrazovanja nastavnika informatike.

3.2. Uzorak

Istraživanje je sprovedeno u maju i junu 2012. godine na reprezentativnom uzorku od 49 nastavnika informatike zaposlenih u 23 srednje škole u Vojvodini. Srednje škole su u ovom radu, uzimajući u obzir predviđen obim, nivo izučavanja i važnost informatičkih kurseva u njihovim kurikulumima, podeljene na tri tipa: gimnazije, tehničke stručne škole i ostale stručne škole. Tipu “ostalih stručnih škola” pripadaju sve one škole koje nisu ni tehničke ni gimnazije (ekonomske, medicinske, poljoprivredne škole i sl.).

Od 49 srednjoškolskih nastavnika informatike koji su obuhvaćeni istraživanjem, 17 izvode nastavu u gimnaziji, 10 je iz tehničkih stručnih škola, dok 22 rade u nekoj od ostalih stručnih škola.

Istraživanjem je obuhvaćeno 6 gimnazija, 6 tehničkih stručnih škola, i 11 ostalih stručnih škola tako da zastupljenost svakog tipa škole u uzorku odgovara zastupljenosti tog tipa škole na teritoriji Vojvodine. Tako su u ovom istraživanju gimnazije zastupljene sa 26.09%, tehničke škole takođe sa 26.09%, dok su ostale stručne škole u uzorku prisutne sa 47.8%. Takođe, broj ispitanih nastavnika informatike po tipu škole u uzorku je proporcionalan njihovoj ukupnoj raspoređenosti po tipu škole u Vojvodini. Tako je u istraživanju učestvovalo 17 nastavnika informatike iz gimnazija, jer gimnazije, od sva tri tipa škola, u proseku imaju najviše zaposlenih nastavnika informatike. Sa druge strane, broj nastavnika informatike koji su učestvovali u istraživanju iz ostalih stručnih škola je nešto veći (22) iako one pojedinačno imaju najmanje zaposlenih nastavnika informatike, ali je ukupan broj tih škola najveći na teritoriji Vojvodine. Takođe, vodilo se računa i o geografskoj rasprostranjenosti škola, kao i o procentu ispitanika u odnosu na ukupan broj srednjoškolskih nastavnika informatike u AP Vojvodina. Sve Školske Uprave na teritoriji Vojvodine (Sombor, Zrenjanin i Novi Sad) su obuhvaćene ovim istraživanjem tako da su prikupljeni upitnici iz sledećih opština: Bačka Palanka, Titel, Bečej, Ruma, Novi Sad, Bač, Sremski Karlovci, Sombor, Subotica, Kula, Apatin, Bačka Topola, Kovačica i Vršac.

3.3. Istraživački instrumenti

Instrument istraživanja je bio upitnik, distribuiran elektronskom poštom. Nastavnici su mogli da odaberu način na koji će popuniti upitnik: elektronski ili ispunjavanjem odštampane forme. Korišćen upitnik prikazan je u prilogu 2 ove disertacije.

Za obradu podataka primenjene su sledeće statističke metode: mere deskriptivne statistike (mere centralne tendencije, mere varijabilnosti, parametri distribucije), metode statistike zaključivanja (hi-kvadrat test, Fisher-ov egzaktni test, jednofaktorska univarijantna analiza varijanse – ANOVA).

3.3.1. Specifikacija upitnika

Upitnik je, u skladu sa atributima koji opisuju profil nastavnika (3.1.), podeljen u sekcije, tako da je svaki atribut opisan sa jednom ili više sekcija (prilog 2). Tako je atribut *akademsko obrazovanje* definisan na osnovu dve sekcije upitnika: „Formalno obrazovanje“ i „Struktura formalnog obrazovanja“. U ovim sekcijama nastavnici su unosili detaljne podatke o završenim studijama (nivo studija, naziv studijskog programa i fakulteta, trajanje studija, prosečna ocena) kao i predmetima koje su položili. *Nastavničko iskustvo* određuju sekcije „Nastavničko iskustvo u godinama“ (godine staža u radu u nastavi) i aktuelni „Rad u nastavi“ (obrazovni profili na kojima ispitani nastavnici drže nastavu i predmete koje predaju). Jedna od sekcija pomoću koje je definisan atribut *Permanentno obrazovanje* je „Sticanje znanja nakon formalnog obrazovanja“. U ovoj sekciji su nastavnici unosili podatke o tome da li su i na koji način sticali dodatna znanja za svaku od generalnih oblasti kurikuluma za nastavnike informatike kao i podatke o kursevima koje su pohađali. Takođe, atribut *Permanentno obrazovanje* je definisan i na osnovu istoimene sekciju u upitniku, kao i na osnovu stavova nastavnika o tome za koje bi softverske tehnologije dodatni kursevi bili najkorisniji u realizovanju nastave. *Ocena osposobljenosti u domenskoj (informatičkoj) oblasti i pedagoško-metodičkom korpusu* je određena na osnovu odgovora na pitanja o poznavanju tematskih oblasti navedenih u sekciji upitnika „Samoevaluacija“. Sekcija „Stavovi nastavnika o kurikulumima“ sadrži pitanja koja definišu dva atributa profila nastavnika: *Stav o značaju domenskog i pedagoško-metodičkog korpusa u kurikulumu obrazovanja nastavnika informatike* i *Stav o zastupljenosti domenskih znanja u kurikulumu srednjoškolskog obrazovanja*

Pored pitanja koja se odnose na definisanje profila nastavnika istraženi su stavovi ispitanih nastavnika informatike o standardizaciji kurikuluma, poznavanja kurikuluma (u okviru sekcije samoevaluacija), dostupnosti kurikuluma, prilagođenosti i usaglašenosti kurikuluma, kao i korisnosti softverskih alata u nastavi.

Tematske oblasti kurikuluma, čije su poznavanje, značaj i zastupljenost nastavnici ocenjivali, su izabrane na osnovu principa iz literature opisanih u drugom poglavlju disertacije, s tim što je pri definisanju domenskih znanja uzet u obzir i nastavni program za računarstvo i informatiku u gimnazijama iz 2012. godine. Program je preuzet sa zvaničnog sajta Zavoda za unapređivanje obrazovanja i vaspitanja na adresi <http://www.zuov.gov.rs/novisajt2012/dokumenta/CRPU/Programi%20za%20gimnaziju%20PDF/16%20racunarstvo%20i%20informatika.pdf>. Tematske oblasti informatičkog (domenskog) aspekta u upitniku obuhvataju sve oblasti sadržane u ovom programu plus neke od oblasti koje nisu predviđene ovim programom a postoje u ACM K12 standardu poput tematskih oblasti: Matematičke osnova računara, Hardverski dizajn, Osnove veštačke inteligencije, kao i pojedinačne teme poput: karijere u računarstvu, mobilne tehnologije, tipovi operativnih sistema. Upitnik podrazumeva i teme iz UML-a i

standarda za elektronski podržano učenje jer su to oblasti koje postoje u većini svetskih kurikuluma analiziranih u drugom poglavlju disertacije.

3.4. Rezultati istraživanja stavova nastavnika

Rezultati su grupisani u skladu sa opisanim profilom, a za ona pitanja za koja postoji statistički značajna razlika (određena Fisher-ovim ili ANOVA testom) rezultati su označeni zvezdicom.

3.4.1. Akademsko obrazovanje

Prosečna ocena ispitanih nastavnika na svim nivoima završenih studija je 7.99, bez statistički značajnih razlika među tipovima škole. Većina (75.5%) nastavnika je završila nivo osnovnih akademskih studija (4 godine), master nivo završilo je 16.3%, 8.2% ima zvanje magistra nauka. Pri tome nisu utvrđene statistički značajne razlike nivoa završenih studija u odnosu na tip škole u kojoj nastavnici predaju.

Nastavnici su u upitnik unosili naziv fakulteta i naziv studijskog programa koji su završili, kao i spisak položenih kurseva, podeljenih po oblastima (informatička, opšta pedagoška, domenska metodička oblast i oblast opštih znanja i veština). Detaljnom analizom tih podataka moglo se uočiti da postoji tri generalna tipa studijskih programa koji su nastavnici završili. To su *Profesor informatike*, (poslovni) *Informacioni sistemi* i *Tehničke nauke*. Grupi *Profesor informatike*, sem studijskog programa pod tim imenom, pripadaju i studijski programi: „Diplomirani informatičar” na Prirodno-matematičkom fakultetu (ovaj studijski program je, kada su ga ispitani nastavnici pohađali, sadržao i kurseve iz oblasti metodike nastave informatike i opštih pedagoških znanja), „Metodika nastave informatike” i „Informatika u obrazovanju”. Tipu završenih studija *Informacioni sistemi* pripadaju studijski programi pod istim nazivom na Ekonomskom fakultetu i Fakultetu organizacionih nauka. Tip završenih studija *Tehničke nauke* čine studijski programi poput: elektronika i telekomunikacije, automatika i računarska tehnika, mikroračunarska elektronika i sl.

Približno polovina (49%) nastavnika je završila neki od studijskih programa za *profesora informatike*, dok druga polovina ima približno podjednako zastupljeno akademsko obrazovanje iz oblasti *informacionih sistema* (26.5%), i *tehničkih nauka* (24.5%) (tabela 8).

		Tip škole						Total	
		Gimnazije		Tehničke stručne škole		Ostale stručne škole			
		N	%	N	%	N	%	N	%
Studijski program	Profesor informatike	13	76.5%	2	20.0%	9	40.9%	24	49.0%
	(poslovni) Informacioni sistemi	2	11.8%	1	10.0%	10	45.5%	13	26.5%
	Tehničke nauke	2	11.8%	7	70.0%	3	13.6%	12	24.5%

Tabela 8. Struktura nastavnika u odnosu na akademsko obrazovanje i tip škole u kojoj izvode nastavu

* Fisher's exact test 16.905, Exact Sig. (2-sided): 0.001

Iz tabele 8 se vidi da je 76.5% nastavnika u gimnaziji završilo neki od smerova za *profesora informatike*; najviše nastavnika informatike u „ostalim stručnim školama” (45.5%) je završilo *Informacione sisteme*, dok je 70% nastavnika u tehničkim stručnim školama završilo studije iz oblasti *tehničkih nauka*. Fisher-ov test ukazuje na to da se ne može isključiti postojanje statistički značajne razlike tipa završenih studijskih programa u odnosu na tip škole u kojoj nastavnici predaju. Ovi rezultati su očekivani i, u velikoj meri, tip završenih studija se poklapa sa obrazovnim profilima kojima nastavnici predaju. U okviru grupe “ostale stručne škole” značajan deo nastavnika radi u ekonomskim školama, a njihovo formalno obrazovanje je, najčešće, iz oblasti (poslovnih) informacionih sistema.

		Tip škole						Total	
		Gimnazije		Tehničke stručne škole		Ostale stručne škole			
		N	%	N	%	N	%	N	%
Tokom studija sticana znanja iz oblasti organizacije rada škole	Da	4	23.5%	3	30.0%	5	22.7%	12	24.5%
	Ne	13	76.5%	7	70.0%	17	77.3%	37	75.5%
Tokom studija sticanja znanja iz oblasti upravljanja školom	Da	1	5.9%	2	20.0%	1	4.5%	4	8.2%
	Ne	16	94.1%	8	80.0%	21	95.5%	45	91.8%
Tokom studija sticana znanja iz oblasti nastavne prakse: delatnost odeljenskog starešine	Da	5	29.4%	2	20.0%	8	36.4%	15	30.6%
	Ne	12	70.6%	8	80.0%	14	63.6%	34	69.4%
Tokom studija sticana znanja iz oblasti metodičke prakse u realizaciji informatičkih nastavnih sadržaja (hospitovanje)*	Da	13	76.5%	2	20.0%	10	45.5%	25	51.0%
	Ne	4	23.5%	8	80.0%	12	54.5%	24	49.0%

Tabela 9. Znanja i veštine iz oblasti nastavne prakse stečena tokom studija

* Fisher's exact test 8.395, Exact Sig. (2-sided):0.016

Detaljnijom analizom strukture i sadržaja studijskih programa iz grupa *informatičkih sistema i tehničkih nauka* može se uočiti da u njima ne postoje sadržaji oblasti opštih pedagoških znanja i metodike nastave informatike. Na ovo ukazuju i rezultati prikazani u tabeli 9 u kojoj su prikazana znanja i veštine iz oblasti nastavne prakse koje su nastavnici imali prilike da steknu tokom studija (čak 75.5% nastavnika nije sticalo znanja iz oblasti organizacije rada škole, 91.8% nije sticalo znanja i veštine iz oblasti upravljanja školom, dok 69.4% nastavnika nisu tokom studija upoznati sa delatnošću odeljenskog starešine). Ovde nije utvrđena statistički značajna razlika u odnosu na tip škole. Nešto je povoljnija situacija povezana sa sticanjem znanja iz oblasti metodičke prakse u realizovanju informatičkih sadržaja (hospitovanje) (51%). Međutim, ovde je utvrđena statistički značajna razlika među tipovima škola u kojima nastavnici izvode nastavu (Fisher-ov egzaktni test 8.935 sa nivoom značajnosti od 0.016), što je i očekivano, uzevši u obzir pomenuti podatak da u gimnazijama radi najviše profesora informatike (76.5%) koji su tokom studija imali priliku da stiču znanja iz ove oblasti. Istraživanje ukazuje da čak 80% nastavnika u tehničkoj školi i 54.5% nastavnika koji rade u ostalim stručnim školama nisu sticali znanja ni iz ove oblasti, što je, takođe, zabrinjavajući podatak.

3.4.2. Nastavničko iskustvo

Prosečno iskustvo nastavnika u radu u školi je 12.88 godina, pri čemu ne postoji statistički značajna razlika u odnosu na tip škole u kojoj nastavnici predaju. Ovde je sabrano i srednjoškolsko i osnovnoškolsko iskustvo, ukoliko ga nastavnik ima.

3.4.3. Ocena osposobljenosti nastavnika u domenskoj (informatičkoj) oblasti i pedagoško-metodičkom korpusu

U upitniku (prilog 2) su postojale 64 tematske oblasti podeljene, u skladu sa analizom prikazanom u drugom poglavlju ovog rada, u 5 aspekata: informatički (domenski) aspekt, opšti pedagoški aspekt, aspekt metodike nastave informatike, aspekt nastavne prakse i opšta znanja i veštine. Nastavnici su ocenama 1-5 ocenjivali svoja znanja iz tih oblasti, tako da je značenje ocena bilo sledeće:

- 1 - „Ne znam da oblast postoji“,
- 2 - „Znam da oblast postoji ali ne znam ništa o oblasti“,
- 3 - „Znam koji su osnovni koncepti oblasti ali ne razumem njihovu suštinu“,
- 4 - „Poznajem i razumem sve osnovne koncepte“,
- 5 - „Posedujem praktična iskustva/primenjujem znanja iz oblasti u nastavi“.

Na ovu grupu pitanja je odgovorilo 47 nastavnika. U radu (Mandić et al., 2014), koji sadrži opis istraživanja iz ovog poglavlja disertacije, tematske oblasti upitnika svih 5 aspekata grupisane su u 22 srodne grupe, a ocena grupe dobijena je izračunavanjem srednje ocene tematskih oblasti koje joj pripadaju. Takva klasifikacija tematskih

oblasti u grupe i princip utvrđivanja ocene grupe primenjeni su i u disertaciji (tabele 10, 12 i 13).

Samoevaluacija nastavnika u odnosu na izabrane 22 oblasti prikazana je u tabeli 10. Tabela sadrži srednje vrednosti i standardne devijacije izračunatih ocena u odnosu na tip završenog studijskog programa.

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Osnove informatike	4.42	.565	4.32	.751	4.29	.582	4.36	.605
Osnove operativnih sistema, programske podrške računara, hardvera i Interneta	4.78	.323	4.64	.408	4.68	.439	4.72	.372
Softverske aplikacije za rad sa tekstom, tabelama i prezentacijama	4.88	.338	4.91	.302	4.86	.332	4.88	.322
Matematičke osnove računara i hardverski dizajn	4.25	.643	3.86	1.398	4.17	.615	4.14	.864
Računarske mreže*	4.63*	.448	3.82*	1.168	4.00*	.640	4.28*	.793
Kompjuterska grafika*	4.63*	.711	3.82*	1.250	4.58*	.515	4.43*	.878
Multimedija*	4.71*	.690	3.82*	1.250	4.42*	.669	4.43*	.903
E-učenje (LMS, standardi za e-učenje, obrazovne računarske igre)*	3.67*	.811	2.82*	.923*	2.53	.627	3.18*	.935
Tipovi i strukture podataka i algoritmi	4.63	.532	4.21	1.138	4.14	.658	4.40	.761
Proceduralno programiranje	4.60	.707	4.09	1.158	4.08	.875	4.35	.890
Objektno-orijentisano programiranje	3.94	.866	3.12	1.393	3.42	1.016	3.62	1.081
Modelovanje i simulacije i osnove veštačke inteligencije*	3.06*	1.035	2.50*	.806	2.17*	.749	2.70*	.982
Baze podataka*	4.58*	.602	4.41*	.801	3.54*	1.054	4.28*	.883
Napredni Internet i programiranje statičkih i dinamičkih web strana*	3.79*	1.062	2.82*	.923	3.00*	.667	3.36*	1.028
Obrazovna psihologija*	4.33*	.816	3.09*	1.336	3.75*	.754	3.89*	1.005
Didaktika*	4.42*	.654	3.36*	1.286	3.67*	.651	3.98*	.944
Pedagogija, personalizovano učenje i sociologija obrazovanja	3.90	.975	3.00	1.256	3.42	.740	3.57	1.045
Strani jezici	4.21	.721	3.91	.831	3.83	.937	4.04	.806
Matematika*	4.63*	.495	4.00*	.894	4.67*	.492	4.49*	.655
Primene novih tehnologija u realizovanju nastave (obrazovna	4.10*	.705	3.33*	1.247	3.22*	1.122	3.70*	1.031

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
tehnologija, LMS/CMS, društveni softver) *								
Metodika nastave informatike	4.15	.896	3.61	1.340	3.44	.956	3.84	1.056
Angažovanje u svakodnevnoj nastavnoj praksi	4.28	.612	4.43	.686	4.08	.581	4.27	.621

Tabela 10. Srednje vrednosti samoevaluacije podeljene po oblastima i tipu završenih studija

*označava oblasti za koje postoji statistički značajna razlika između grupa izračunata ANOVA testom

Uočava da su nešto nižim ocenama nastavnici ocenili svoja znanja iz oblasti savremenijih informatičkih oblasti poput: e-učenja (analizom upitnika se zapaža da se to posebno odnosi na standarde za e-učenje, administriranje LMS/CMS) – 3.18, Objektno-orijentisano programiranje – 3.62, Modelovanje i simulacije i osnove veštačke inteligencije – 2.70; Napredni Internet i programiranje statičkih i dinamičkih web strana – 3.36. Visoke ocene nisu date ni znanjima iz oblasti obrazovne psihologije - 3.89, pedagogije, personalizovanog učenja i sociologije obrazovanja – 3.57, primeni novih tehnologija u realizovanju nastave – 3.70, i metodike nastave informatike – 3.84. Ova pojava je još izraženija ako se uporede stečena znanja u odnosu na tip završenog studijskog programa, i za tu namenu je primenjena jednofaktorska univarijantna ANOVA. Tako, statistički značajne razlike postoje za oblast e-učenja za koju su nastavnici koji su završili oblast informacionih sistema ocenili svoje znanje sa 2.82, dok nastavnici koji su završili tehničke nauke svoje su znanje u proseku ocenili sa 2.53. Višu ocenu iz ove oblasti su sebi dali nastavnici informatike koji su neke od oblasti koje obuhvata aspekt e-učenja izučavali i tokom studija. Statistički značajne razlike postoje i u oblasti multimedije, kompjuterske grafike, računarskih mreža koje su manje izučavane na studijskim programima informacionih sistema. I u oblasti modelovanja i simulacije i veštačke inteligencije, baza podataka i naprednog Interneta i programiranja statičkih i dinamičkih web strana postoji statistički značajna razlika među ocenama. I za ove oblasti najviše ocene za svoja znanja dali su profesori informatike. ANOVA test je pokazao i statistički značajne razlike u znanjima ove tri grupe nastavnika u odnosu na veoma važne oblasti obrazovne psihologije i didaktike. Iako su prosečne ocene svih nastavnika za ove dve oblasti na zadovoljavajućem nivou (3.89 i 3.98), kada se pogleda raspodela ocena po tipu završenog studijskog programa uočavaju se značajno više ocene kod profesora informatike u odnosu na druge dve grupe nastavnika. Razlog za to može da bude struktura studijskih programa informacionih sistema i različitih smerova tehničkih nauka u kojima opšte pedagoške oblasti nisu zastupljene. Slični zaključci se mogu izvesti i za oblast primene novih tehnologija u realizovanju nastave i oblast pedagogije, personalizovanog učenja i sociologije

obrazovanja. Ovde, takođe, postoji statistički značajna razlika među grupama nastavnika, i profesori informatike najviše ocenjuju svoja znanja. Statistički značajna razlika postoji i iz oblasti matematike, s tim što su za ovu oblast nešto nižim ocenama sebe ocenili nastavnici koji su završili studijske programe iz oblasti informacionih sistema.

3.4.4. Permanentno obrazovanje

Tabele 11 i 12 pokazuju stavove nastavnika o tome da li je i koliko često potrebno sticanje dodatnih znanja za svaki od aspekata kurikuluma.

Koliko je potrebno permanentno obrazovanje za:	Nije potrebno		Potrebno je		Veoma je potrebno	
	N	%	N	%	N	%
Informatički (domenski) aspekt*	0	0.0%	7	15.1%	37	84.9%
Opšti pedagoški aspekt	3	6.8%	27	61.4%	14	31.8%
Aspekt metodike nastave informatike	2	4.5%	24	54.5%	18	40.9%
Aspekt nastavne prakse	3	6.8%	18	40.9%	23	52.3%

Tabela 11. Stavovi nastavnika o potrebi za permanentnim obrazovanjem

*Fisher's exact test 6.572, Exact Sig. (2-sided): 0.03

Koliko je često potrebno unapređivati znanja za:	Bar jednom u 10 godina		Bar jednom u 5 godina		Bar jednom u 2 godine		Svake godine	
	N	%	N	%	N	%	N	%
Informatički (domenski) aspekt	0	0.0%	3	6.7%	11	24.4%	31	68.9%
Opšti pedagoški aspekt	4	8.9%	18	40.0%	17	37.8%	6	13.3%
Aspekt metodike nastave informatike	3	6.7%	12	26.7%	22	48.9%	8	17.8%
Aspekt nastavne prakse	3	6.7%	15	33.3%	18	40.0%	9	20.0%

Tabela 12. Stavovi nastavnika o učestalosti potrebe za permanentnim obrazovanjem

Tabele 11 i 12 pokazuju da nastavnici smatraju da je potrebno često unapređivati znanja i to iz svih aspekata kurikuluma. Očekivano, većina nastavnika smatra da je za informatički (domenski) aspekt najpotrebnije permanentno obrazovanje (84.9% nastavnika smatra da je „veoma potrebno“, tabela 11) kao i da je ta znanja potrebno najčešće unapređivati (tabela 12). Statistički značajna razlika u odnosu na tip škole postoji samo kod pitanja u kom su nastavnici ocenjivali koliko je potrebno permanentno obrazovanja za informatički aspekt (tabela 11). Od 7 nastavnika koji su dali srednju ocenu („potrebno je“), 6 nastavnika je iz „ostalih stručnih škola“ i samo jedan nastavnik iz gimnazije, dok su svi ostali nastavnici iz gimnazija i svi nastavnici iz tehničkih škola, koji su odgovarali na ovo pitanje, ocenili najvišim rangom („veoma potrebno“) potrebu permanentnog obrazovanja za informatički aspekt.

U tabeli 13 prikazani su rezultati koji pokazuju u kojoj meri su nastavnici sticali dodatna znanja za svaki od aspekata.

Način sticanja dodatnih znanja	Oblast							
	Informatička (domenska) oblast		Opšta pedagoška oblast		Oblast metodike nastave informatike		Oblast nastavne prakse	
	N	%	N	%	N	%	N	%
Nisu sticana dodatna znanja	1	2.1%	3	6.5%	9	19.6%	12	26.1%
Dodatno znanje sticano učešćem na radionicama/seminarima	32	68.1%	31	67.4%	19	41.3%	21	45.7%
Dodatno znanje sticano učešćem na naučnim i stručnim konferencijama	14	29.8%	2	4,3%	2	4.3%	4	8.7%
Dodatno znanje sticano proučavanjem literature	45	95.7%	35	76,1%	32	69.6%	26	56.5%
Dodatno znanje sticano pohađanjem posebnih kurseva	28	59.6%	21	45.7%	11	23.9%	4	8.7%

Tabela 13. Način sticanja znanja nakon formalnog obrazovanja

Većina nastavnika (97.9%) je ostvarivala neki vid sticanja dodatnih znanja iz informatičke oblasti nakon završenih studija. Ovde je bilo moguće označiti više odgovora i većina nastavnika (95.7%) se dodatno obrazovala bar proučavanjem literature, dok je 68.1% učestvovalo i na radionicama i seminarima, 29.8% i na naučnim i stručnim konferencijama.

59.6% nastavnika je pohađalo posebne kurseve. Uvidom u detaljnije podatke o odslušanim dodatnim kursevima koje su nastavnici unosili u upitnik, može se zapaziti da su nastavnici najčešće pohađali neki od osnovnih (ECDL i “Arhimedes”) kurseva koji predviđaju izučavanje: računarskog hardvera, softverskih aplikacija za rad sa tekstom, prezentacijama i tabelama, osnova Interneta, operativnih sistema, kao i kurseve o multimediji, računarskim mrežama, osnovama baza podataka. Ova znanja upravo pripadaju onim oblastima za koje su nastavnici sebi dali najvišu ocenu u samoevaluaciji (tabela 10). Nastavnici su ređe pohađali kurseve iz savremenijih oblasti informatičkog domena za koje su i niže ocenili svoja znanja u tabeli 10. Nastavnici koji su pohađali dodatne kurseve iz objektno-orijentisanog (OO) programiranja, Moodle LMS, SCORM standarda ili „računarstva u oblaku“ su u samoevaluaciji višim ocenama ocenili svoja znanja iz tih oblasti. Ovi rezultati ukazuju na potrebu organizovanja dodatnih kurseva iz niže ocenjenih oblasti, posebno onih za koje su se nastavnici najređe dodatno edukovali, kao i stimulisanja nastavnika da ih pohađaju.

U poređenju sa domenom informatike, gde samo 2.1% nastavnika nije sticalo dodatna znanja, nešto je veći procenat nastavnika koji nisu uopšte sticali dodatna znanja iz opšte pedagoške oblasti (6.5 %), dok

rezultati za oblasti metodike nastave informatike (19.6 %) i oblasti nastavne prakse (26.1%) ukazuju na značajno manju zastupljenost permanentnog obrazovanja. I za ova tri aspekta, najveći deo nastavnika je sticao dodatna znanja proučavanjem literature, a u najmanjoj meri učešćem na naučnim i stručnim konferencijama. S obzirom na značajan udeo nastavnika koji su završili studije koje ne sadrže aspekte pedagoške oblasti, metodike nastave informatike, i nastavne prakse, rezultati koji ukazuju da su nastavnici ipak sticali dodatna znanja iz ovih oblasti su ohrabrujući, a značajan deo nastavnika je dodatno znanje sticao i pohađanjem posebnih kurseva, posebno iz opšte pedagoške oblasti (45.7%). Fisher-ovi testovi nisu ukazali na statističku razliku u sticanju dodatnih znanja iz informatičke oblasti, opšte pedagoške oblasti, oblasti metodike nastave informatike i nastavne prakse u odnosu na tip škole u kojoj nastavnici rade.

3.4.5. Stavovi nastavnika o značaju domenskog i pedagoško-metodičkog korpusa u kurikulumu obrazovanja nastavnika informatike

U tabeli 14 prikazani su stavovi nastavnika o sadržaju visokoškolskih kurikuluma za nastavnike informatike, odnosno koliko je značajno da se svaka od predloženih oblasti nalazi u visokoškolskom kurikulumu da bi se uspešno realizovala nastava informatike na srednjoškolskom nivou. Nastavnici su ocenjivali značaj u odnosu na obrazovne profile na kojima oni drže nastavu. U upitniku su bile 64 tematske oblasti čiji je značaj ocenjivan na sledeći način: 1 – „nema nikakav značaj“, 2 – „vrlo mali značaj“, 3 – „srednji značaj“, 4 – „veliki značaj“, 5 – „presudan značaj“. Na ovu grupu pitanja je odgovorilo 46 nastavnika. Nastavnici su imali mogućnost da dodaju tematsku oblast koja nije ponuđena u upitniku, a koju smatraju značajnom za realizovanje nastave. Međutim, nastavnici nisu smatrali da postoje takve oblasti. Tematske oblasti su i ovde grupisane na isti način kao pri samoevaluaciji nastavnika a rezultati ocene značaja prikazani su u tabeli 14.

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Osnove informatike	4.44	.583	4.15	.784	4.08	.629	4.23	.656
Softverske aplikacije za rad sa tekstom, tabelama i prezentacijama	4.90	.283	4.53	.670	4.89	.315	4.82	.426
Osnove operativnih sistema, programske podrške računara, hardvera i Interneta	4.62	.323	4.62	.494	4.32	.608	4.50	.506
Matematičke osnove računara i hardverski dizajn*	3.88*	.574	4.05*	.832	3.18*	1.169	3.63	.974
Računarske mreže	4.21	.639	4.30	.537	3.84	1.106	4.08	.856

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Kompjuterska grafika	4.41	.618	4.10	.738	4.11	1.049	4.22	.841
Multimedija	4.53	.514	4.30	.483	4.21	.713	4.35	.604
E-učenje (LMS, standardi za e-učenje, obrazovne računarske igre)	3.92	.722	3.83	.707	3.42	.777	3.70	.763
Tipovi i strukture podataka i algoritmi*	4.63*	.576	4.20*	1.021	3.61*	1.167	4.12	1.038
Proceduralno programiranje*	4.68*	.585	4.25*	1.034	3.08*	1.216	3.92	1.211
Objektno-orijentisano programiranje*	4.55*	.634	3.97*	.909	2.98*	1.288	3.78	1.213
Modelovanje i simulacije i osnove veštačke inteligencije*	3.76*	.710	3.65*	.914	2.74*	1.019	3.32	1.002
Baze podataka	4.56	.659	4.25	1.034	3.82	1.397	4.18	1.122
Napredni Internet i programiranje statičkih i dinamičkih web strana*	4.24*	.695	3.93*	.750	3.35*	1.136	3.80	.980
Obrazovna psihologija	4.24	.752	4.40	.699	4.26	.806	4.28	.750
Didaktika	4.29	.772	4.20	.789	4.37	.761	4.30	.756
Pedagogija, personalizovano učenje i sociologija obrazovanja	4.29	.576	4.27	.750	4.21	.931	4.25	.761
Strani jezici	4.76	.437	4.70	.483	4.74	.562	4.74	.491
Matematika*	4.82*	.393	4.30*	.823	4.21*	.855	4.46	.751
Primene novih tehnologija u realizovanju nastave (obrazovna tehnologija, LMS/CMS, društveni softver)	4.22	.824	4.07	.927	3.81	.723	4.01	.810
Metodika nastave informatike*	4.66*	.324	4.43*	.598	4.13*	.665	4.39	.585
Nastavna praksa	4.56	.768	4.05	.762	4.16	.883	4.28	.836

Tabela 14. Značaj tematskih oblasti u kurikulumu obrazovanja nastavnika informatike

*označava oblasti za koje postoji statistički značajna razlika između grupa izračunata ANOVA testom

Uočljivo je da su nastavnici dali visoke ocene značaja za gotovo svaku od oblasti (minimalna ocena je 3.32). Za većinu onih oblasti koje imaju nešto nižu ocenu značaja u nastavi, jednofaktorski univarijantni ANOVA test je ukazao na statistički značajne razlike u odnosu na tip škole u kojima nastavnici predaju. Tako, iako je ukupna ocena značaja oblasti matematičkih osnova računara i hardverskog dizajna nešto niža (3.63), za ovu oblast ocene se razlikuju po tipu škole (u tehničkoj školi 4.05, u gimnaziji 3.88, u ostalim stručnim školama samo 3.18), a rezultat ANOVA testa ukazuje na statistički značajnu razliku u odnosu

na tip škole. Slično, značaj proceduralnog programiranja u nastavi je u gimnaziji 4.68, u tehničkim stručnim školama 4.25, dok je u ostalim stručnim školama 3.08; značaj objektno-orijentisanog programiranja u gimnaziji je ocenjen prosečno sa 4.55, u tehničkim stručnim školama 3.97, dok je u ostalim stručnim školama 2.98. Takođe, značaj aspekta modelovanje i simulacije i osnove veštačke inteligencije je prosečno ocenjen sa „samo“ 3.32, međutim značaj ove oblasti u gimnazijama je 3.76, u tehničkim stručnim školama 3.65, dok je u ostalim stručnim školama 2.98. Statistički značajna razlika u odnosu na tri tipa škole postoji i za oblasti: tipovi i strukture podataka i algoritmi, napredni Internet i programiranje statičkih i dinamičkih web strana. Uočljivo je da se razlike značaja pojedinih tema u odnosu na tip škole odnose na savremenije i/ili kompleksnije koncepte iz informatičke oblasti. Nastavnici u gimnazijama i tehničkim stručnim školama ih smatraju značajnijim za realizovanje nastave nego nastavnici u ostalim stručnim školama. Naravno, i sadržaji i broj informatičkih predmeta koji obrađuju ove oblasti, kao i nivo izučavanja je drugačiji ako se uporede aktuelni kurikulumi tri tipa škola. Iz tabele 14 primetno je i da ne postoji statistički značajna razlika između tipova škole u odnosu na važnost oblasti baze podataka, kao što bi se možda očekivalo. Detaljna analiza uzorka obuhvaćenih škola navodi na mogući zaključak da je značaj ove oblasti u ostalim stručnim školama na višem nivou možda i zbog značajnog udela ekonomskih škola u ostalim stručnim školama, za koje oblast baza podataka predstavlja važan deo informatičkog kurikuluma. Statistički značajne razlike, dobijene ANOVA testom postoje i za oblasti matematike i metodike nastave informatike čiji je značaj, takođe, najviše ocenjen u gimnazijama, a najniže u ostalim stručnim školama.

3.4.6. Stavovi nastavnika o zastupljenosti domenskih znanja u kurikulumu srednjoškolskog obrazovanja

Tabela 15 pokazuje stavove nastavnika o zastupljenosti informatičkih tematskih oblasti u kurikulumu po kome oni drže nastavu. Dakle, ovde su ocenjivani srednjoškolski informatički kurikulumi u odnosu na predložene tematske oblasti. Na ovu grupu pitanja je odgovorilo 46 nastavnika. I ovde su tematske oblasti grupisane u srodne grupe izračunavanjem srednje ocene, a zastupljenost je ocenjivana ocenama 1-5 (1 – „ne postoji“, 2 – „nedovoljno“, 3 – „prihvatljivo“, 4 – „dobro je zastupljeno“, 5 – „adekvatno je zastupljeno“). Ni ovde nastavnici nisu smatrali da postoje još neke tematske oblasti koje bi trebale biti zastupljene u kurikulumu a nisu navedene u upitniku.

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Osnove informatike	3.79	.730	3.70	.422	3.68	.869	3.73	.728
Osnove operativnih sistema, programske podrške računara,	4.42	.429	4.38	.629	4.06	.743	4.26	.629

Tematska oblast	Tip završenih studija						Total	
	Profesor informatike		Informacioni sistemi		Tehničke nauke			
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
hardvera i Interneta								
Softverske aplikacije za rad sa tekstom, tabelama i prezentacijama	4.94	.131	4.50	.707	4.53	.731	4.67	.602
Matematičke osnove računara i hardverski dizajn	3.24	1.002	3.60	.937	2.84	1.143	3.15	1.069
Računarske mreže	3.71	.792	3.65	1.001	3.29	.933	3.52	.900
Kompjuterska grafika*	4.71*	.588	3.80*	1.229	3.11*	1.524	3.85	1.366
Multimedija*	4.53*	.717	4.20*	.789	3.42*	1.170	4.00	1.054
E-učenje (LMS, standardi za e-učenje, obrazovne računarske igre)	2.06	.775	2.20	.613	2.32	1.003	2.20	.839
Tipovi i strukture podataka i algoritmi*	4.22*	.881	3.43*	1.458	2.58*	1.105	3.37	1.313
Proceduralno programiranje*	4.44*	.917	3.25*	1.477	2.11*	.937	3.22	1.474
Objektno-orijentisano programiranje*	3.82*	.826	2.97*	1.222	2.09*	.942	2.92	1.224
Modelovanje i simulacije i osnove veštačke inteligencije	1.79	.936	2.10	.775	1.92	1.071	1.91	.950
Baze podataka	3.94	.933	3.55	1.066	3.08	1.465	3.50	1.243
Napredni Internet i programiranje statičkih i dinamičkih web strana	3.16	1.119	2.47	.757	2.28	1.172	2.64	1.127

Tabela 15. Zastupljenost domenskih znanja u kurikulumu srednjoškolskog obrazovanja

*označava oblasti za koje postoji statistički značajna razlika između tipa škole izračunata ANOVA testom

I ovde, za neke od niže ocenjenih oblasti, primena jednofaktorskog univarijantnog ANOVA testa ukazuje na postojanje statistički značajne razlike u odnosu na tip škole. To se posebno odnosi na tri oblasti: tipovi i strukture podataka i algoritmi, proceduralno programiranje i objektno-orijentisano programiranje. Tako, iako je prosečna ocena zastupljenosti oblasti tipovi i strukture podataka i algoritmi 3.37, zastupljenost ovog aspekta u gimnazijama je 4.22, a u ostalim stručnim školama 2.58. Sličan odnos prosečnih ocena zastupljenosti je i kod proceduralnog programiranja i objektno-orijentisanog programiranja. Ocena zastupljenosti oblasti proceduralno programiranje u gimnazijama je 4.44 (ukupna prosečna zastupljenost je 3.22), dok je zastupljenost oblasti objektno-orijentisano programiranje u gimnazijama ocenjena sa 3.82 (ukupna prosečna zastupljenost za sve škole je samo 2.92). S obzirom na razlike u kurikulumima tri tipa škola, zastupljenost aspekata iz oblasti programiranja je očekivano veća u gimnaziji u odnosu na, posebno,

ostale stručne škole. Međutim, ako je očekivano niska zastupljenost ovih oblasti u ostalim stručnim školama, njihova zastupljenost u tehničkim stručnim školama nije odgovarajuća; zastupljenost objektno-orijentisanog programiranja je, na primer, ocenjena u tehničkim školama sa samo 2.97. Statistički značajne razlike u odnosu na tip škole postoje i za oblasti računarske grafike i multimedije; njihova zastupljenost je najviše ocenjena u gimnazijama, dok je ocena njihove zastupljenost najniža u ostalim stručnim školama.

Zadovoljavajuće prosečne ocene (jednake ili veće od 3.5) o zastupljenosti u srednjoškolskim kurikulumima su nastavnici dali za sedam (od 14) aspekata: osnove informatike, softverske aplikacije za rad sa tekstom, tabelama i prezentacijama, osnove operativnih sistema, programske podrške računara, hardvera i Interneta, kompjuterska grafika, multimedija, računarske mreže i baze podataka. Ipak, od nabrojanih 7 aspekata samo 3 imaju prosečnu ocenu jednaku ili veću od 4.00. Takođe, detaljnom analizom upitnika može se zapaziti da pojedine nabrojane, zadovoljavajuće ocenjene oblasti sadrže tematske oblasti čija je zastupljenost ocenjena veoma nisko. Primer za to je oblast osnove informatike čija je zastupljenost ocenjena sa 3.73. Ipak, zastupljenost tematske oblasti „kompjuterske etika, bezbednost i zaštita podataka“ koja spada u ovu oblast je ocenjena veoma nisko, pri čemu je značaj te tematske oblasti u nastavi ocenjen kao visok.

Posebno je niska ocena data u svim tipovima škola za zastupljenost sledeće četiri oblasti: e-učenje, modelovanje i simulacije i osnove veštačke inteligencije, napredni Internet i programiranje statičkih i dinamičkih web strana, matematičke osnove računara i hardverski dizajn. Iako su te četiri oblasti najniže ocenjene i kada je kriterijum ocenjivanja bio značaj u nastavi, kao i to da je zastupljenost oblasti programiranja (očekivano, s obzirom na različite srednjoškolske kurikulume) značajno različita u zavisnosti od tipa škole (i najviše je ocenjena njena zastupljenost u gimnazijama), rezultati ukazuju na moguću neadekvatnu zastupljenost značajnog broja oblasti. To se posebno odnosi na kompleksnije i savremenije koncepte (posebno OO programiranje, napredne Internet tehnologije, standarde i tehnologije e-učenja i sl.)

3.4.7. Stavovi o modalitetima obrazovanja nastavnika informatike

	Master akademske studije usmerene na pedagoške aspekte i aspekte metodike informatike		Master akademske studije usmerene na informatičke (domenske) aspekte		Integrisane osnovne i master akademske studije	
	N	%	N	%	N	%
Modalitet obrazovanja nastavnika	8	17.78%	8	17.78%	29	64.44%

Tabela 16. Najpodesniji koncept obrazovanja nastavnika

Nastavnici informatike obuhvaćeni istraživanjem većinom (64.44%, tabela 16) smatraju da je za obrazovanje budućih nastavnika informatike najpodesniji modalitet integrisanih osnovnih i master

akademskih studija gde bi se istovremeno izučavali informatički i pedagoški aspekti. Nastavnici koji smatraju da je za obrazovanje budućih nastavnika informatike pogodniji modalitet gde su osnovne akademske studije razdvojene od master studija su jednako distribuirani (po 17.78%) po opcijama gde bi se najpre izučavali samo informatički aspekti, a zatim na master studijama pedagoški aspekti, ili, suprotno, u kome bi se nakon završenih nekih od pedagoških studija na master studijama izučavali samo kursevi informatičkog domena.

3.4.8. Stavovi nastavnika o standardizaciji, dostupnosti i usaglašenosti kurikuluma

Ispitani nastavnici informatike su u upitniku odgovarali i na pitanja koja se ne odnose direktno na formiranje profila nastavnika kako je definisan u sekciji 3.1. Tako su istraženi stavovi nastavnika i o potrebi za standardizacijom, o poznavanju savremenih standarda u oblasti, o dostupnosti i usaglašenosti aktuelnih kurikuluma svih nivoa obrazovanja. U ovoj sekciji je prikazana analiza rezultata koji se odnose na te aspekte.

3.4.8.1. Standardizacija kurikuluma

	Uopšte mi nije poznat proces		Poznajem minimalno proces		Znam osnovnu strukturu procesa i mogu nabrojati većinu učesnika		Poznajem ceo proces	
	N	%	N	%	N	%	N	%
Da li su Vam poznate faze i učesnici procesa standardizacije kurikuluma?	16	34.0%	25	53.2%	4	8.5%	2	4.3%

Tabela 17. Poznavanje procesa standardizacije kurikuluma

	Nisam upoznat/a		Znam ukupan planirani cilj		Poznajem strukturu i ciljeve po nivoima		Detaljno poznajem standard	
	N	%	N	%	N	%	N	%
Da li Vam je poznat ACM K12 CS standard?	34	72.3%	9	19.1%	3	6.4%	1	2.1%

Tabela 18. Poznavanje ACM K12 CS standarda

	Nema potrebe		Potrebno je minimalno usaglašavanje		Potrebno je srednje usaglašavanje		Standard treba usvojiti	
	N	%	N	%	N	%	N	%
Ocenite potrebu usaglašavanja informatičkog obrazovanja u R. Srbiji sa ACM K12 CS standardom (ukoliko ga poznajete)	0	0.0%	0	0.0%	3	27.3%	8	72.7%

Tabela 19. Potreba usaglašavanja informatičkog obrazovanja sa ACM K12 CS standardom

	Nisam upoznat/a		Znam ukupan planirani cilj		Poznajem strukturu i nazive svih modula		Poznajem sadržaje svih modula	
	N	%	N	%	N	%	N	%
Da li Vam je poznat ECDL plan i program?	7	14.9%	7	14.9%	10	21.3%	23	48.9%

Tabela 20. Poznavanje ECDL programa

	Nema potrebe		Potrebno je minimalno usaglašavanje		Potrebno je srednje usaglašavanje		Principe ECDL-a treba usvojiti u našim srednjim školama	
	N	%	N	%	N	%	N	%
Ocenite potrebu usaglašavanja informatičkog obrazovanja na srednjoškolskom nivou sa ECDL sertifikacijom (ukoliko je poznajete)	0	0.0%	3	7.5%	14	35.0%	23	57.5%

Tabela 21. Potreba usaglašavanja srednjoškolskog informatičkog obrazovanja sa ECDL sertifikacijom

* Fisher's exact test 10.445, Exact Sig. (2-sided): 0.013

Većina ispitanih nastavnika minimalno ili uopšte ne poznaje proces standardizacije kurikuluma. Takođe, nastavnici uglavnom nisu upoznati sa ACM K12 predlogom kurikuluma. Ipak, nastavnici koji poznaju ACM-ov standard (Tabela 19) su saglasni sa stanovištem da je potrebno razmotriti ACM K12 model pri definisanju srednjoškolskih informatičkih kurikuluma. S druge strane, nastavnici su tokom nastavničke karijere imali više prilike da se neposredno upoznaju sa ECDL programima; neki od ispitanih nastavnika su u upitniku upravo naveli da su pohađali ECDL kurseve. Na to ukazuju i rezultati prikazani u tabeli 20.

Statistički značajna razlika u odnosu na tip škole postoji samo kod poslednjeg pitanja prikazanog u tabeli 21. Većina nastavnika iz ostalih stručnih škola (15 nastavnika od 19 koliko ih je odgovaralo na ovo pitanje) smatra da "Principe ECDL-a treba usvojiti u našim srednjim školama". S druge strane, kod nastavnika iz gimnazija i tehničkih

škola su odgovori ravnomernije raspoređeni između poslednja dva ponuđena odgovora, s tim što većina nastavnika iz ova dva tipa škole smatra da je “potrebno srednje usaglašavanje” sa ECDL sertifikacijom na srednjoškolskom nivou.

3.4.8.2. Poznavanje kurikuluma

	Ne znam koji su to predmeti		Znam samo koji su to predmeti		Znam planirane ciljeve predmeta		Znam planirane ishode predmeta		Znam nastavne sadržaje predmeta	
	N	%	N	%	N	%	N	%	N	%
Ocenite ocenom 1 – 5 Vaše poznavanje informatičkih predmeta u nastavnom planu osnovne škole	7	14.9%	7	14.9%	5	10.6%	8	17.0%	20	42.6%
Ocenite ocenom 1 – 5 Vaše poznavanje informatičkih predmeta koji postoje u nastavnim planovima srednjoškolskog nivoa obrazovanja (profili za koje Vi ne držite nastavu)	1	2.1%	7	14.9%	10	21.3%	11	23.4%	18	38.3%
Ocenite ocenom 1 – 5 Vaše poznavanje informatičkih predmeta u informatički usmerenim studijskim programima visokoškolskog nivoa obrazovanja	11	23.9%	10	21.7%	17	37.0%	4	8.7%	4	8.7%
Ocenite ocenom 1 – 5 Vaše poznavanje informatičkih predmeta u studijskim programima visokoškolskog nivoa obrazovanja koji nisu informatički usmereni	18	39.1%	9	19.6%	9	19.6%	4	8.7%	6	13.0%

Tabela 22. Poznavanje kurikuluma svih nivoa obrazovanja

Da bi nastavnici adekvatno realizovali informatičku nastavu u srednjoj školi potrebno je da znaju koje su informatičke sadržaje učenici izučavali u osnovnoj školi, kao i informatičke sadržaje koje je predviđeno da učenici izučavaju u srednjoj školi u okviru drugih kurseva i obrazovnih profila. Rezultati prikazani u tabeli 22 pokazuju stavove nastavnika da zadovoljavajuće poznaju predviđene kurikulume nivoa na kome drže nastavu, kao i informatički kurikulum osnovnoškolskog nivoa obrazovanja. Takođe, budući da srednjoškolsko obrazovanje predstavlja i osnovu za dalje visokoškolsko obrazovanje za učenike koji se za to opredele, važno je da nastavnici poznaju koji se informatički sadržaji izučavaju na visokoškolskim studijskim programima. Međutim, iz tabele 22 se vidi da njihovo poznavanje fakultetskih informatičkih kurseva nije odgovarajuće. Za stavove nastavnika prikazane u tabeli 22 Fisher-ov test nije ukazao na postojanje statistički značajne razlike u odnosu na tip škole u kojoj nastavnici predaju.

3.4.8.3. Stavovi nastavnika o dostupnosti kurikuluma

	Nikako mi nije dostupan		Dostupan mi je u štampanom obliku		Dostupan mi je putem Interneta	
	N	%	N	%	N	%
Ocenite ocenom 1–3 da li Vam je/su dostupan/ni informatički nastavni plan/ovi i program/i koji se primenjuje/ju u obrazovnim institucijama Republike Srbije za nivo obrazovanja na kome Vi držite nastavu?	0	0.0%	21	46.7%	24	53.3%

Tabela 23. Dostupnost kurikuluma

	Nije potrebno		Potrebno je		Neophodno je	
	N	%	N	%	N	%
Ocenite ocenom 1-3 potrebu dostupnosti nastavnog plana i programa putem Interneta	3	6.5%	11	23.9%	32	69.6%
Ocenite ocenom 1–3 potrebu Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja na kome Vi držite nastavu	5	10.9%	24	52.2%	17	37.0%
Ocenite ocenom 1–3 da li smatrate da je potreban uvid u informatičke nastavne planove i programe prethodnog nivoa obrazovanja za uspešno sprovođenje nastave informatike koju Vi držite	5	10.9%	22	47.8%	19	41.3%
Ocenite ocenom 1–3 potrebu Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja koji prethodi onom nivou na kome	11	23.9%	22	47.8%	13	28.3%

držite nastavu						
Ocenite ocenom 1 – 3 da li smatrate da je potreban uvid u informatičke nastavne planove i programe sledećeg nivoa obrazovanja za uspešno sprovođenje nastave informatike koju Vi držite	5	10.9%	23	50%	18	39.1%

Tabela 24. Potreba dostupnosti kurikuluma

	Nema mogućnosti		Postoji ograničena-srednja mogućnost uticaja		Potpuno mi je omogućeno učešće u razvoju nastavnog plana i programa	
	N	%	N	%	N	%
Ocenite ocenom 1 – 3 mogućnost Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja na kome držite nastavu	23	51.1%	18	40%	4	8.9%
Ocenite ocenom 1–3 mogućnost Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja koji prethodi onom nivou na kome držite nastavu	31	67.4%	14	30.4%	1	2.2%

Tabela 25. Mogućnost uticaja na kurikulum

	Nije koristan		Srednje koristan		Veoma koristan	
	N	%	N	%	N	%
Ocenite ocenom 1–3 koliko bi Vam bio koristan softverski alat koji bi Vam omogućio pretraživanje i prikazivanje nastavnog plana i programa/kurikuluma po: ishodima; nastavnim sadržajima; znanjima i veštinama koje obezbeđuju; vezama između sadržaja i ishoda (koji sadržaji obezbeđuju ishode); nastavnim metodama; primerima organizacije nastave; nastavnim materijalima i sl.	1	2.2%	17	37.0%	28	60.9%
Ocenite ocenom 1 – 3 koliko bi Vam bio koristan softverski alat koji bi Vam omogućio elektronsko prilaganje predloga za razvoj novih i modifikaciju postojećih nastavnih planova i programa	1	2.2%	16	34.8%	29	63.0%

Tabela 26. Korisnost softvera za rad sa kurikulumima

Iz tabele 23 je uočljivo da nastavnici smatraju da su im dostupni nastavni planovi i programi po kojima realizuju nastavu. Većina nastavnika je iskazala stavove (tabela 24) da je za uspešno realizovanje nastave potreban uvid u kurikulumе svih nivoa obrazovanja kao i mogućnost uticaja na kreiranje srednjoškolskog i osnovnoškolskog informatičkog kurikuluma. Međutim, na osnovu rezultata prikazanih u tabeli 25 ispitanim nastavnicima nije omogućeno učešće u razvoju kurikuluma. Takođe, nastavnici su većinom visoko ocenili funkcionalnost softvera koji bi im obezbeđivao izmene postojećih i kreiranje novih kurikuluma, kao i pretraživanje kurikuluma po različitim kriterijumima (tabela 26). Ne postoji statistički značajna razlika u odnosu na tip škole kod pitanja prikazanih u tabelama 23-26.

3.4.8.4. Stavovi nastavnika o usaglašenosti kurikuluma

	Stepen strukturiranosti i sadržajnosti nastavnog plana i programa po kome Vi držite nastavu	
	N	%
Nastavni plan i program je nepregledan i na neadekvatan način prikazuje sadržaje	2	4.3%
Nastavni plan i program je dobro strukturiran ali nedovoljno (ili previše) detaljno opisuje sadržaje	13	28.3%
Nastavni plan i program nije dobro strukturiran iako na dobrom nivou detalja prikazuje sve sadržaje	11	23.9%
Nastavni plan i program je adekvatno strukturiran i sadržaji tematskih oblasti su dovoljno detaljno prikazani	20	43.5%

Tabela 27. Stepen strukturiranosti i sadržajnosti kurikuluma

	Uopšte ne odgovara		Minimalno odgovara		Srednje odgovara		Odgovara stanju u oblasti		Odlučno je prilagođen	
	N	%	N	%	N	%	N	%	N	%
Ocenite ocenom 1-5 koliko Vaš trenutni nastavni plan i program odgovara aktuelnom stanju u oblasti računarstva	0	0.0%	2	4.3%	25	54.3%	16	34.8	3	6.5%

Tabela 28. Aktuelnost kurikuluma

	Nikako ne prati dinamiku promena u oblasti		Prati dinamiku promena u oblasti sa neprihvatljivim kašnjenjem		Prati dinamiku promena u oblasti sa velikim kašnjenjem		Prati dinamiku promena u oblasti sa prihvatljivim kašnjenjem		Prati dinamiku promena u oblasti bez kašnjenja	
	N	%	N	%	N	%	N	%	N	%
Ocenite ocenom 1 –	4	8.7%	6	13.0%	17	37.0%	18	39.1%	1	2.2%

5 dinamiku promena nastavnog plana i programa u odnosu na dinamiku promena u oblasti računarstva										
--	--	--	--	--	--	--	--	--	--	--

Tabela 29. Prilagođenost kurikuluma dinamici promena u oblasti

	Nije nikako prilagođen		Minimalno je prilagođen		Zadovoljavajuće je prilagođen		Dobro je prilagođen		Odlično je prilagođen	
	N	%	N	%	N	%	N	%	N	%
Ocenite ocenom 1 – 5 koliko je aktuelni nastavni plan i program prilagođen potrebama primene tehnološki podržanog učenja	3	6.7%	8	17.8%	25	55.6%	8	17.8%	1	2.2%
Ocenite ocenom 1 – 5 koliko je, po Vama, aktuelni nastavni plan i program prilagođen upotrebi novih tehnologija i elektronskih servisa	5	10.9%	16	34.8%	11	23.9%	10	21.7%	4	8.7%

Tabela 30. Prilagođenost kurikuluma primeni novih tehnologija

Ocenite ocenom 1 – 5 koliki je <i>stepen usaglašenosti</i> nastavnih planova i programa	Nije nikako usaglašeno		Minimalno je usaglašeno		Zadovoljavajuće je usaglašeno		Dobro je usaglašeno		Odlično je usaglašeno	
	N	%	N	%	N	%	N	%	N	%
Osnovna – srednja škola*	4	8.7%	7	15.2%	18	39.1%	7	15.2%	10	21.7%
Fakultet – srednja škola	0	0.0%	5	11.1%	17	37.8%	8	17.8%	15	33.3%

Tabela 31. Usaglašenost kurikuluma različitih nivoa obrazovanja

* Fisher's exact test 18.120, Exact Sig. (2-sided): 0.006

	Nikako		Srednje		Veoma	
	N	%	N	%	N	%
Ocenite ocenom 1 – 3 koliko bi Vam bio koristan softverski alat koji bi Vam omogućio utvrđivanje usaglašenosti nastavnih planova i programa različitih nivoa obrazovanja	2	4.3%	18	39.1%	26	56.5%

Tabela 32. Korisnost softvera za proveru usaglašenosti kurikuluma različitih nivoa obrazovanja

	Ne proveravam		Upoređivanjem sa pisanim dokumentom		Primenom računarskih alata		Neki drugi način	
	N	%	N	%	N	%	N	%
Na koji način proveravate usaglašenost realizovanog nastavnog sadržaja u odnosu na propisani nastavni plan i program	5	10.9%	30	65.2%	10	21.7%	1	2.2%

Tabela 33. Način provere realizovanosti propisanog kurikuluma

	Nije koristan		Srednje		Veoma	
	N	%	N	%	N	%
Ocenite ocenom 1 – 3 koliko bi Vam bio koristan softverski alat koji bi Vam omogućio utvrđivanje usaglašenosti realizovanog i propisanog nastavnog plana i programa/kurikuluma	2	4.3%	20	43.5%	24	52.2%

Tabela 34. Korisnost softvera za proveru usaglašenosti realizovanog i propisanog kurikuluma

U ovoj sekciji disertacije prikazani su rezultati istraživanja koji se odnose na stavove nastavnika o usaglašenosti kurikuluma. Rezultati ukazuju na to da su kurikulumi po kojima nastavnici drže nastavu zadovoljavajuće opisani i struktuirani (tabela 27). Ipak, ispitanici nastavnici su iskazali stav da je srednjoškolske informatičke kurikulume neophodno unaprediti tako da budu aktuelniji, odnosno bolje usaglašeni sa promenama u oblasti (tabela 28 i 29). Takođe, na osnovu tabele 30, može se zaključiti da je sadašnje kurikulume potrebno bolje prilagoditi primeni novih tehnologija. Najveći procenat ispitanih nastavnika je odgovorio da je srednjoškolski informatički kurikulum (po kome oni drže nastavu) „zadovoljavajuće“ usaglašen sa osnovnoškolskim i visokoškolskim informatičkim kurikulumima (tabela 31). Statistički značajna razlika u odnosu na tip škole postoji kod pitanja u kome se ocenjivala usaglašenost osnovnoškolskih i srednjoškolskih informatičkih kurikuluma. Nastavnici zaposleni u srednjim stručnim školama su dali značajno više ocene za ovu kategoriju u odnosu na nastavnike iz gimnazije i tehničke škole što je

u skladu sa razlikama u informatičkim kurikulumima u ovim tipovima škola. Budući da informatički kurikulum u srednjim stručnim školama uglavnom ne predviđa izučavanje kompleksnih informatičkih sadržaja, usaglašenost između osnovnoškolskih informatičkih kurikuluma i informatičkih kurikuluma u srednjim stručnim školama je očekivano visoka. Nastavnici su visoko ocenili i funkcionalnost softvera koji bi im omogućavao proveru usaglašenosti kurikuluma različitih nivoa obrazovanja (tabela 32) kao i utvrđivanje da li je propisan nastavni plan i program realizovan (tabela 34). Na potrebe nastavnika za softverom koji bi obezbeđivao ovakvu funkcionalnost ukazuju i odgovori u tabeli 33 iz koje se vidi da najveći procenat ispitanih nastavnika koristi pisane dokumente za proveru usaglašenosti realizovanog i propisanog nastavnog plana i programa.

3.4.9. Korisnost softverskih alata za realizovanje nastave

U tabeli 35 nabrojano je 10 softverskih alata koje su nastavnici rangirali ocenama 1-10 u odnosu na njihovu korisnost za realizovanje nastave informatike.

Softverski alati za realizovanje nastave	Prosečni rang (1-10)	
	Mean	SD
Prevodioci za različite programske jezike	3.93	3.365
Alati za dizajn i specifikaciju softvera	4.24	2.564
Simulacioni alati i simulatori fenomena i uređaja	3.74	2.499
Digitalni rečnici, enciklopedije, atlasi, itd.	4.50	2.119
Obrazovne računarske igre	3.86	1.885
Alati za upravljanje nastavnim sadržajima (kreiranje i skladištenje digitalnih (multimedijalnih) nastavnih sadržaja; pretraživanje i preuzimanje digitalnih nastavnih sadržaja; kreiranje elektronskih kurseva)	8.23	1.927
Alati za kolaborativno učenje (blog, facebook, itd)	4.93	2.672
Alati za komunikaciju (nastavnik-učenik, učenik-učenik)	6.88	2.301
Alati za proveru postignuća učenika (testiranje)	7.53	1.919
Alati za podršku individualizovanom učenju	7.10	2.304

Tabela 35. Stavovi nastavnika o korisnosti softverskih alata za izvođenje nastave

Uočljivo je da su nastavnici najvišim ocenama ocenili korisnost softverskih alata koji olakšavaju realizovanje nastave uopšte, odnosno alate koji nisu specifični za nastavu informatike. Iz table 35 se vidi da su najviše ocenjeni alati za upravljanje nastavnim sadržajima. Više ocene (iznad 6.5) dobili su i alati za komunikaciju, alati za proveru postignuća učenika i alati za podršku individualizovanom učenju.

U poslednjem delu upitnika (prilog 2) nastavnici su trebali da navedu za koje softverske tehnologije smatraju da bi im dodatni kursevi bili najpotrebniji u realizovanju nastave. Oni su većinom iskazali potrebu upravo za onim tehnologijama čije su poznavanje najlošije ocenili u samoevaluaciji (tabela 10) i za koje nisu imali prilike da se dodatno obrazuju kroz posebne kurseve (sekcija 3.4.4).

Tako su najčešće navedene sledeće oblasti: objektno-orijentisano programiranje, programiranje statičkih web strana, UML, napredne Internet tehnologije, „računarstvo u oblaku“, Moodle i uopšte upravljanje okruženjima za e-učenje (LMS, CMS). Manji broj nastavnika je naveo da bi za realizovanje nastave bili korisni i kursevi iz računarskih mreža, multimedije, kompjuterske grafike i baza podataka.

3.5. Zaključci i implikacije istraživanja

Istraživanje je pokazalo da značajan broj ispitanih nastavnika informatike u srednjim školama u Vojvodini ne poseduje adekvatno formalno obrazovanje. Tokom studija, većina nastavnika nije imala prilike da stekne znanja i veštine iz oblasti nastavne prakse. Takođe, samoevaluacija je pokazala da značajnom broju nastavnika, koji nemaju adekvatno akademsko obrazovanje, nedostaju znanja iz opšte pedagoške oblasti (poput obrazovne psihologije i didaktike), kao i iz oblasti primene novih tehnologija u obrazovanju. Ukupna prosečna ocena kojima su nastavnici evaluirali sopstvena znanja iz oblasti metodike nastave informatike je takođe relativno niska.

Ispitani nastavnici su većinom skloni da se dodatno usavršavaju iz svih oblasti. Međutim, kada je u pitanju informatička oblast, oni to najmanje čine, iz sfere savremenijih i kompleksnijih aspekata informatike i računarstva poput objektno-orijentisanog programiranja, naprednih Internet koncepata ili standarda i sistema za upravljanje e-učenjem. Upravo iz ovih oblasti informatičkog domena nastavnici su ocenili svoja znanja najnižom ocenom i istakli najveću potrebu za dodatnim usavršavanjem. **Stoga je potrebno da nadležno ministarstvo i univerziteti obezbede kvalitetno permanentno obrazovanje u skladu sa dinamikom promena u oblasti i potrebama nastavnika.**

Sve oblasti kurikuluma nastavnika informatike ponuđene upitnikom, ispitanici su ocenili visokim ocenama u odnosu na značaj za nastavničku profesiju i nisu dodali ni jednu novu oblast. **Ovi rezultati ukazuju na potrebu kreiranja savremenih kurikuluma za nastavnike u kojima su sva navedena znanja inkorporirana.** Pri tome, rezultati ukazuju na postojanje statistički značajne razlike za neke oblasti u odnosu na tip škole (matematičke osnove računara i hardverski dizajn, proceduralno i OO programiranje, tipovi i strukture podataka i algoritmi, itd.). Ovi rezultati su očekivani jer značaj, na primer, OO paradigme u realizovanju nastave u gimnaziji i u nekoj od stručnih škola (na primer poljoprivrednoj školi) nije isti, budući da i informatički kurikulumi u ovim školama na različitom nivou i obimu predviđaju izučavanje ovih oblasti.

Rezultati zastupljenosti informatičkih oblasti u srednjoškolskim informatičkim kurikulumima su pokazali da je potrebno osavremenjivanje srednjoškolskih kurikuluma (iako je u gimnazijama nešto povoljnija zastupljenost najniže ocenjenih oblasti, mnoge značajne oblasti su ocenjene kao nedovoljno zastupljene).

Većina ispitanih nastavnika informatike smatra da su **posebne integrisane osnovne i master studije, u kojima bi se paralelno/istovremeno izučavali svi aspekti, najpodesniji modalitet obrazovanja budućih nastavnika informatike.**

Nastavnici, koji su upoznati sa srednjoškolskim informatičkim standardima navedenim u upitniku (ACM K12 i ECDL), smatraju da je **potrebno usaglašavanje srednjoškolskih informatičkih kurikuluma sa aktuelnim standardima.**

Stavovi nastavnika o potrebi uvida u kurikulume i uticaja na kurikulume, kao i iskazani stavovi o (potrebi za boljom) usklađenosti srednjoškolskih informatičkih kurikuluma sa aktuelnim stanjem u oblasti, upućuju na važnost **reprezentovanja kurikuluma tako da nastavnicima bude omogućen uvid u kurikulume, razvoj kurikuluma, menjanje kurikuluma (u skladu sa dinamikom promena u oblasti) i praćenje promena u kurikulumu.** Takođe, ispitani nastavnici informatike su i eksplicitno istakli potrebu za softverom koji bi im obezbedio utvrđivanje usaglašenosti kurikuluma različitih nivoa obrazovanja, izmene postojećih i kreiranje novih kurikuluma, kao i pretraživanje kurikuluma po različitim kriterijumima.

Poglavlje IV

4. Ontološki model kurikuluma informatike

4.1. Pojam ontologije

Ontologija se, po (Gruber, 1995), odnosi na “specifikaciju konceptualizacije”, odnosno predstavlja opis koncepata domena i relacija među njima, namenjen ljudima i programima (softverskim agentima) da dele znanje. U (Daconta et al., 2003) se pojam „konceptualizacija“ tumači kao apstraktni (mentalni) model „nekih fenomena u svetu“ i podrazumeva identifikovanje relevantnih koncepata domena, njihovih veza i svojstava. Takođe, konceptualizacija se u istom izvoru opisuje kao način razmišljanja o određenom delu realnosti. “Specifikacija” se odnosi na način da se detaljno i precizno opišu delovi realnosti, odnosno da se opiše “konceptualizacija”. Domen se, u kontekstu definisanja ontologije, odnosi na oblast predmeta interesovanja ili oblast znanja (Daconta et al., 2003).

U literaturi se često ističe razlika u tumačenju pojma ontologije kroz pisanje početnog „velikog slova O,“ i „malog o“. „Veliko O“ označava, najčešće, filozofsku disciplinu, dok „malo o“ upućuje da je ontologija inženjerska disciplina iz domena informacionih tehnologija koja se razvija “proteklih desetak godina” (Daconta et al., 2003). Kao inženjerski “produkt” ontologija se sastoji od specifičnog rečnika primenjenog da bi se opisao (deo) realnosti, kao i od “skupa eksplicitnih pretpostavki koje se odnose na namenjeno značenje tog rečnika” (Guarino, 1998). Ontologija je, dakle, slična rečniku, s tim što nudi više detalja i strukturu koja omogućava kompjutersko procesuiranje njenog sadržaja.

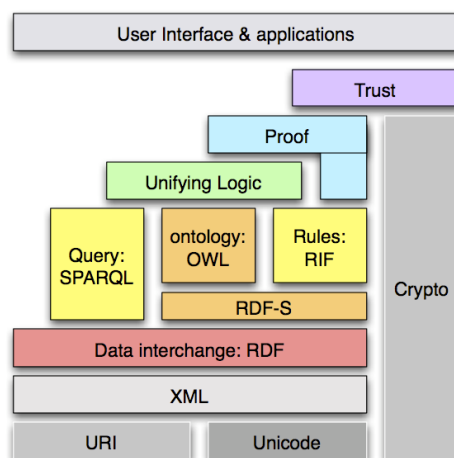
U literaturi se često navode i sledeće definicije ontologije: ontologija obezbeđuje deljeno i zajedničko razumevanje domena koje može biti razmenjeno među ljudima i aplikacionim sistemima (Davies et al., 2002); ontologije predstavljaju zajednički, deljiv i ponovo upotrebljiv pogled na domen, i daju značenje informacionim strukturama koje razmenjuju informacioni sistemi (Brewster et al., 2004).

Ontologije se reprezentuju pomoću sledećih komponenti:

1. Klase (generalni koncepti),
2. Instance,
3. Svojstva (i vrednosti svojstava) koncepata,
4. Veze među konceptima,
5. Funkcije i procesi koji uključuju koncepte,
6. Ograničenja i pravila (Daconta et al., 2003).

Za reprezentaciju ontologija koristi se OWL (Web Ontology Language). OWL predstavlja reviziju DAML-OIL jezika, odnosno semantički unapređenu verziju tog jezika (W3C, 2004a). OWL je razvijen od strane Web Ontology Working Group-e

(<http://www.w3.org/2001/sw/WebOnt/>). Svi jezici Semantičkog Weba imaju prednost u odnosu na jezike koji se nalaze „ispod njih“ u zamišljenoj piramidi ili „kolaču“ izgrađenom od slojeva ili stek dijagramu koje je prvi predstavio Tim Berners-Lee (slika 1).



Slika 1. Semantički Web stek (preuzeto iz (Berners-Lee et al., 2006))

Prvi sloj steka Semantičkog Weba čine *Unicode* i *URI*, koji se koriste za identifikovanje resursa pomoću jedinstvenih identifikatora. Naredni sloj čine *XML* i *XML Schema*. *XML* obezbeđuje “površinsku” sintaksu za strukturane dokumente pri čemu ne nameće semantička ograničenja za značenje tih dokumenata (W3C, 2004a). *XML Schema* ograničava „slobodnu“ strukturu *XML* dokumenata i, takođe, proširuje *XML* sa tipovima podataka (*datatypes*). *RDF* predstavlja model podataka za objekte (resurse) i relacije među njima, obezbeđujući jednostavnu semantiku za te modele podataka, koji se takođe mogu reprezentovati pomoću *XML*-a. Sledeći sloj, *RDF Schema*, je vokabular za opisivanje svojstava i klasa *RDF* resursa sa semantikom u formi hijerarhije - generalizacije. Iznad je *OWL* (zajedno sa *RIF-rule* jezikom) koji obogaćuje rečnik za opis svojstava i klasa obezbeđujući (između ostalog): relacije među klasama (na primer *disjoint* relacije), kardinalnost, jednakost, bogatiji opis svojstava, karakteristika svojstava i enumerisane klase. Generalno, interpreter jezika višeg nivoa može korektno interpretirati jezik svakog nivoa ispod svog nivoa, naravno, pored svog specifičnog jezika (Daconta et al., 2003). *SPARQL* predstavlja upitni jezik za *RDF*. Naredni nivo je *Unifying Logic* sloj koji se koristi za zaključivanje nad *RDF* iskazima. Za validaciju *RDF* modela koristi se *Proof* sloj, dok se *Trust* sloj koristi kao podrška bezbednosti Semantičkog Weba. Na vrhu steka nalaze se korisnički interfejs i aplikacije.

OWL obezbeđuje tri „podjezika“ različitog nivoa ekspresivnosti:

- OWL Lite je namenjen prvenstveno korisnicima kojima je potrebna hijerarhija u vidu klasifikacije i jednostavna ograničenja. Dozvoljene vrednosti kardinalnosti su samo 0 ili 1, ali za ovako definisan nivo jezika je jednostavno obezbediti softversku podršku. Takođe, obezbeđuje jednostavnu migraciju sa različitih taksonomija.

- OWL DL (*Description Logic*) je namenjen korisnicima koji žele maksimalnu ekspresivnost ali uz zadržavanje računarske kompatibilnosti i mogućnosti zaključivanja/interpretacije od strane računara (u konačnom vremenu). Uključuje sve OWL jezičke konstrukte, ali uz određena ograničenja kao na primer: iako klasa može biti potklasa drugih klasa, ona ne može biti instanca druge klase.
- OWL Full obezbeđuje maksimalnu izražajnost ali bez „računarskih garancija“. Tako, klasa može biti tretirana kao kolekcija entiteta ili kao instanca sama po sebi. Nije realno očekivati, po (W3C, 2004a), da će neki softver biti u mogućnosti da podrži kompletno zaključivanje za svako svojstvo ovog jezika.

Često je u primeni i Semantic Web Rule Language (SWRL) (W3C, 2004b) koji je nastao u cilju unapređenja mogućnosti OWL-a. SWRL pravila omogućavaju redukovanje ograničenja ontološkog zaključivanja, posebno u pogledu identifikovanja semantičkih veza među instancama. SWRL je zasnovan na kombinaciji OWL-a (OWL DL i OWL Lite) i Rule Markup Language-a (RML).

U ovom radu je korišćen OWL DL jezik za reprezentaciju ontologija. Razlog za to je visoka ekspresivnost ovog jezika i mogućnost računarske interpretacije.

4.2. Pregled literature iz oblasti primene ontologija u obrazovanju

Ontologije mogu opisivati obrazovni domen iz različitih perspektiva, obezbeđujući bogatiji opis i ponovno preuzimanje obrazovnog sadržaja (Fernández-Breis et al., 2012). Na osnovu raspoložive savremene literature, ontološka primena u obrazovanju se najčešće svodi na najmanje jednu od sledeće tri kategorije:

1. Upotreba ontologija za definisanje obrazovnih resursa (i metapodataka).
2. Primena ontologija za personalizovano učenje.
3. Korišćenje ontologija za reprezentovanje kurikuluma.

Jedan od najčešćih oblika primene ontologije u obrazovanju je definisanje koncepata i veza obrazovnih objekata u nekom repozitoriju e-učenja. Po (Wang, 2008) ontologija za LO (*Learning Object*) repozitorije je „konceptualna mreža svih međusobno povezanih objekata učenja“. Ona predstavlja semantičke veze među LO u „aplikacionom domenu“ (Wang, 2008).

LO se u (LTSC, 2002) definišu kao „bilo koji entitet, digitalan ili ne-digitalan, koji može biti korišćen, ponovo upotrebljen, referenciran tokom tehnološki podržanog učenja“. Metapodaci se definišu kao informacije o objektu; bilo da je fizički ili digitalan.

Po (Doan & Bourda, 2006), semantika šeme metapodataka nije dobro definisana. Primena ontologija predstavlja „ključni korak“ ka unapređenju opisa semantike obrazovnih resursa tako da softverskim

agentima bude olakšano zaključivanje, a alatima za pretraživanje preciznije pronalaženje informacija.

Budući da je semantika heterogena, opisi iste discipline od strane različitih nastavnika mogu biti različiti. Iako su, na neki način, ontologije skrivene iza metapodataka, takve skrivene ontologije je suviše komplikovano predstaviti, budući da sve moguće semantičke veze među svim obrazovnim resursima ne mogu biti lako standardizovane. U (Xin-juan et al., 2007) se navodi da:

- Različiti resursi za e-učenje mogu koristiti drugačije termine za opis istog domenskog znanja u istoj oblasti.
- Jedan termin može imati različito značenje u različitim resursima e-učenja.
- Postoje drugačiji izrazi za iste ili slične koncepte u okviru raznih sistema e-učenja.
- Iako postoji mnogo veza među različitim koncepcijama e-učenja, uzevši u obzir svojstva distribuiranosti i autonomnosti sistema za e-učenje, značenje tih odnosa nije moguće efikasno opisati.

Nabrojani faktori dovode do toga da obrazovni resursi ne mogu biti efikasno deljeni među različitim platformama i sistemima korišćenjem „samo“ metapodataka (Xin-juan et al., 2007). Takođe, u (Mustaro & Silveira, 2006) se naglašava da se u situacijama kada je repozitorij za čuvanje LO „glomazan“ i distribuiran na Internetu, korišćenje metapodataka i ključnih reči za pronalaženje željenog LO-a pokazalo kao neefikasno, jer mnoge „potencijalne asocijacije sa različitim aspektima učenja“ mogu biti preskočene. Ovi problemi mogu biti razrešeni importovanjem ontologija u obrazovne sisteme. Ontologije obezbeđuju neophodnu „armaturu“ oko koje treba graditi bazu znanja, i postavljaju osnovu za razvoj i deljenje web sadržaja, servisa i aplikacija koje je moguće ponovo koristiti. Važno je napomenuti da korišćenje ontologija ne isključuje upotrebu metapodataka, već omogućava dodatno definisanje eksplicitnih veza među objektima učenja i pomaže u „razvoju nestruktuiranog pedagoškog razvoja“ (Wang, 2008). Budući da konvencionalnim LOM (*Learning Object Metadata*) nedostaje računarska interpretativnost potrebna sistemima za reprezentaciju znanja pri pretraživanju relevantnih LO, u (Hsu, 2012) se predlaže integrisanje semantičkih web tehnologija u LOM. U (Knight et al., 2006) se zastupa primena ontologija za specifikaciju dizajna učenja, objekata učenja i relacija među njima što može rezultovati u njihovoj većoj (re)iskoristivosti upotrebom poluautomatskih alata i servisa. U (Romero et al., 2014) je izneta pretpostavka da odgovarajući opis LO namenjenih ocenjivanju zasnovan na ontologijama unapređuje njihovo pronalaženje i preuzimanje od strane nastavnika, učenika i softverskih sistema i favorizuje njihovo ponovno korišćenje i kolaborativan rad. U (Pathmeswaran & Ahmed, 2009) je ilustrovana primena semantičkih web tehnologija i ontologija za razvoj mobilnih repozitorija objekata učenja i omogućavanje mobilnog učenja (engl. *m-learning*).

U (Xin-juan et al., 2007) je prikazan jedan način analize repozitorija za e-učenje koji omogućava prilagodljivo i personalizovano učenje. Kursevi i treninzi su podeljeni u što manje delove informacija, tako da se prilagođavaju individualnim sposobnostima učenika. Autori ističu prednosti kreiranja „personalnih ontologija korisnika“ analiziranjem informacija dobijenih pri registraciji koje mogu uključiti prethodno znanje, kognitivni stil ili karakteristike učenja. U (Kizilkaya et al., 2007) se daje pregled primene ontologija u okruženjima za e-učenje i, takođe, naglašava važnost potrebe da se u personalizovanom okruženju instrukcioni dizajn osmisli tako da uzima u obzir karakteristike učenika – pre svega: metakognitivne sposobnosti. U (Sanchez-Alonso & Vovides, 2007) je korišćena specifična ontologija kao osnova za inkorporiranje informacija o metakognitivnim aspektima studenata tako da LMS može „sam“ selektovati i preporučiti zadatke prilagođene razvoju i poboljšanju studentskih metakognitivnih sposobnosti u kontekstu e-učenja. U istom izvoru se zaključuje da ontologije obezbeđuju kreiranje nove paradigme za objekte učenja po kojoj objekti učenja mogu biti definisani tako da pomažu studentima da „poboljšaju svoje performanse“. U radu (Yaghmaie & Bahreininejad, 2011) je pokazano da primena ontologija može da unapredi adaptivne sisteme u kojima „standardi za obrazovni sadržaj pokazuju slabosti pri personalizaciji“. U (Paquette, 2007) je opisana ontologija za dizajniranje učenja zasnovanog na kompetencijama. Na osnovu predstavljene ontologije definisani su alati i servisi softverske platforme za e-učenje. U (Vesin et al., 2012) prikazana je primena Semantičkog Weba i ontologija za personalizovano učenje kursa (Java programskog jezika) korišćenjem principa identifikacije stila učenja i preporuke odgovarajućih sadržaja. U (Harchay et al., 2012) se predlaže primena ontologija za personalizovano ocenjivanje učenika u mobilnom okruženju.

4.2.1. Primena ontologija za reprezentaciju kurikuluma

Zbog mogućnosti reprezentovanja kurikuluma na mašinski razumljiv način, kao i osobina ponovne iskoristivosti i deljivosti, ontološki pristup se sve češće primenjuje za prezentovanje nekog od oblika kurikuluma (Chi., 2009; Fernández-Breis et al, 2012; Demartini et al., 2013; Elsayed, 2009). U (Dexter & Davies, 2009) se opisuje motivacija za primenu ontologija pri razvoju kurikuluma koji, zasnovan na predviđenim ishodima učenja, kompetencijama studenata i standardima, „formira kompleksnu multidimenzionalnu matricu informacionih elemenata povezanih u obliku više ka više veza“. Korišćenje ontologija može olakšati promenu bilo kog aspekta kurikuluma u skladu sa novim preporukama pristiglih od akreditacionih tela, agencija za standardizaciju, nadležnih organa i sl. Ontološka primena može omogućiti, pri novonastalim promenama u kurikulumu, identifikovanje i praćenje svih aspekata na koje su te promene imale uticaja. Upotreba ontologija podržava najvažnije principe koji moraju biti zadovoljeni pri razvoju kurikuluma:

- dinamičnost sistema,

- usklađenost sa promenama,
- lakoću administriranja i
- nisku cenu održavanja.

U (Dexter & Davies, 2009) se sumira da ontološki-bazirani kurikulumi mogu obezbediti: definisanje najpre samo najvažnijih delova (jezgra) kurikuluma tako da elementarni „materijal“ bude odvojen od kasnijih, naprednijih koncepata; mapiranje eksternih kriterijuma i očekivanih ishoda tako da ih razvijeni kurikulum zadovoljava; dizajn sadržaja kurikuluma na sistematičniji, rigorozniji i transparentniji način tako da veća grupa nastavnika može da učestvuje u njegovom kreiranju; efikasno praćenje i upravljanje promenama u bilo kom delu kurikuluma i njihovog uticaja na celokupni kurikulum; personalizovano učenje studenata – studenti mogu da upravljaju sopstvenim procesom učenja; „preciznije i prilagođenije“ planiranje i organizovanje nastavne aktivnosti; pretraživanje kurikuluma u skladu sa rutom prilagođenom administratoru, tutoru ili studentu. Po (Memon & Khoja, 2009) primena ontologija i tehnologija Semantičkog Weba može obezbediti lakše akademsko upravljanje i semantičko povezivanje različitih funkcionalnosti i podсистема Univerziteta u integrisanu platformu. Upotreba ontologija može omogućiti dodeljivanje kursa odgovarajućem predavaču u skladu sa njegovim sposobnostima, ponudu izbornih kurseva u skladu sa osobnostima studenata određene grupe, poređenje uspeha studenata koji pripadaju različitim departmanima i usklađivanje kurikuluma u skladu sa tim, adaptiranje kursa u zavisnosti od raznih oblika evaluacija različitih predmeta i studijskih programa i slično. U istom izvoru se navodi da bi predloženi princip primene ontologija umanjio redundantni i „umarajući“ posao prilikom procesa akreditacije i obezbedio podršku inteligentnih upita od različitih zainteresovanih organizacija.

U radu (Libbrecht et al., 2008) je kreirana ontologija koja omogućava deljenje digitalnog sadržaja za držanje nastave matematike i predstavlja matematički domen tema i kompetencija. U (Cassel et al., 2008) je dat predlog ontologije kurikuluma iz oblasti računarstva i opisana je ideja primene ontologija po kojoj bi korisnik mogao da bira iz padajućeg menija željene ishode učenja i, u skladu sa izabranim ishodom, odgovarajući koncepti u razvijenoj ontologiji bi bili obeleženi. Takođe, predloženi sistem za reprezentaciju kurikuluma, zasnovan na ontologiji, omogućava određivanje razlike među različitim računarskim studijskim programima. Poređenjem označenih sekcija u različitim programima, mogu se utvrditi preklapanja i razlike među programima. U (Fernández-Breis et al, 2012) su primenjene ontologije kao osnova softvera za razvoj i održavanje obrazovnog kurikuluma koji je obezbeđivao informacije o dužini instrukcionih jedinica, trajanju nastave, instrumentima ocenjivanja, prikaz neispredavanih lekcija i sl. U radu (Chi, 2009) je korišćena ontologija za reprezentovanje redosleda sadržaja u kurikulumu („putanje učenja“), kao i materijala kurseva. U (Demartini et al., 2013) je prikazana ontologija koja reprezentuje akademsko okruženje predloženo Bolonjskom reformom. Fokus je na postupku

kreiranja ontologije tranzicijom iz postojećih relacionih sistema poluautomatizovanim postupkom transformacije leksikona koncepata u ontologiju, pri čemu predložena ontologija ne sadrži eksplicitnu reprezentaciju kurikuluma niti predviđa usaglašavanje kurikuluma.

4.3. Ontološki model kurikuluma

Za razvoj ontologija, u ovom radu, primenjen je programski paket Protégé (Noy et al., 2000). Semantička verifikacija kreiranog modela realizovana je primenom *Pellet reasoner*-a (Sirin et al., 2007).

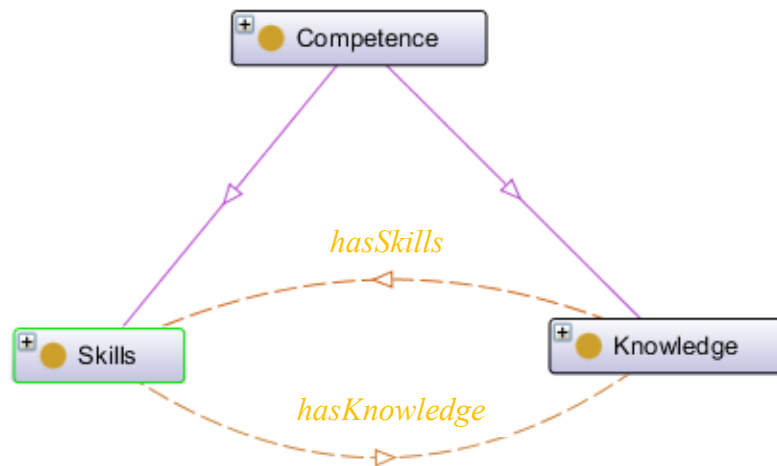
4.3.1. Kompetencija kao osnovna klasa ontološkog modela

Cilj ovog rada je predlaganje softverskog alata koji bi omogućio utvrđivanje da li nastavnički kurikulum obezbeđuje adekvatne kompetencije potrebne za izvođenje nastave u srednjoj školi, realizovane u skladu sa aktuelnim srednjoškolskim informatičkim kurikulumom. U srednjoškolskom informatičkom kurikulumu su, s druge strane, definisane informatičke kompetencije koje je potrebno da steknu učenici. Stoga su modeli nastavničkog i srednjoškolskog informatičkog kurikuluma zasnovani na kompetencijama, te je osnovna klasa ontološkog modela nastavničkog i srednjoškolskog kurikuluma kompetencija.

U literaturi se mogu pronaći brojne definicije kompetencije. U (Commission of the European Communities, 2005) kompetencije su definisane kao kombinacija znanja, sposobnosti i stava prilagođenim određenoj situaciji. Po (LTSC, 2008) reč kompetencija se koristi u „veoma širokom značenju koja uključuje sposobnosti, znanja, zadatke i učeničke ishode“ i može biti primenjena za opis informacija bilo kog od nabrojanih pojmova u „kontekstu obrazovanja, učenja ili treninga“. (Fleishman et al., 1995) određuju kompetenciju kao „mešavinu“ znanja, veštine, mogućnosti, motivacije, uverenja, vrednosti i interesovanja. Slične pojmove koje označavaju reč kompetencija navodi i Mirabile (1997); to su: znanja, veštine, mogućnosti ili karakteristike povezane sa visokim performansama na poslu. U (Shippmann, 2000) se navode još neka tumačenja reči kompetencija od strane eksperata, poput: „konstrukt koji pomaže pri definisanju nivoa znanja i sposobnosti“; „znanja, sposobnosti i osobine koje razlikuju visoko uspešne izvršioce od prosečnih“ itd. U istom izvoru se sumira da, bez obzira na kontekst (obrazovni, psihološki, pravni ili stručni), pojam kompetencija označava uspešno obavljanje određenog zadatka ili aktivnosti, odnosno – adekvatno poznavanje određenog domena **znanja ili veštine**.

Prema tome, kompetencija podrazumeva znanja i sposobnosti (veštine). Stoga su u ovom radu, znanja i veštine mapirane u posebne klase ontološkog modela kurikuluma (*Knowledge* i *Skills*), strukturirane kao potklase klase *Competence* (slika 2). U kontekstu ontološkog modela kurikuluma, tematske oblasti kurikuluma su mapirane u potklase klase *Knowledge*, dok su predviđene stečene veštine nakon izučavanja određene tematske oblasti mapirane u

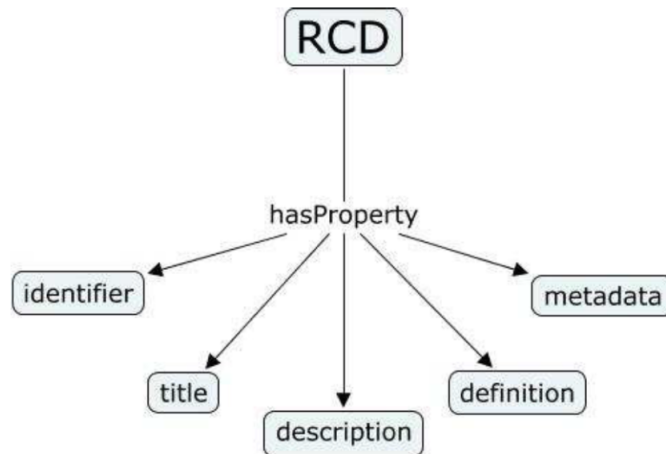
odgovarajuće potklase klase *Skills*. Potklase klase *Skills* i *Knowledge* su povezane preko objektnog svojstva *hasKnowledge*, odnosno njemu inverznog svojstva *hasSkills*. Objektno svojstvo *hasKnowledge* može povezivati jednu potklasu domena *Skills* sa jednom ili više različitih *Knowledge* potklasa. Analogno, jedna *Knowledge* potklasa može biti povezana preko objektnog svojstva *hasSkills* sa jednom ili više potklasa klase *Skills*.



Slika 2. Struktura klase *Competence*

4.3.2. IEEE RCD standard

Da bi se obezbedila interoperabilnost sa sistemima za upravljanje učenjem, više (engl. *upper*) klase informatičkih kurikuluma su modelovane u skladu sa IEEE RCD standardom. IEEE RCD (IEEE Standard for Learning Technology—Data Model for Reusable Competency Definitions) standard (LTSC, 2008) definiše model podataka za opis, referenciranje i deljenje definicija kompetencije, prvenstveno za potrebe online i distribuiranog učenja. Standard obezbeđuje metod za formalno reprezentovanje ključnih karakteristika kompetencija, nezavisno od određenog konteksta primene. Takođe, obezbeđuje interoperabilnost između sistema učenja koji sadrže informacije o kompetenciji. IEEE RCD je široko rasprostranjen internacionalan standard (videti, na primer, <http://wiki.teria.no/display/inloc/LOD> gde je za eContentPlus Icoper definiciju ishoda učenja usvojen aplikacioni profil ovog standarda) i predstavlja sintaksni standard, čime je omogućena interoperabilnost između različitih sistema podataka, tako da svaki sistem može da definiše koji deo primljenih podataka predstavlja kompetenciju i koji deo kompetencija označava naslov, opis i sl. (Lundqvist et al., 2011). Potiče od IMS Reusable Definitions for Competencies and Educational Objectives (RDCEO) (IMS Global Learning Consortium, 2002). Slika 3 ilustruje data model IEEE RCD standarda sa njegovim komponentama.



Slika 3. IEEE RCD Data Model (preuzeto iz (Mandić et al., 2013))

RCD se, dakle, sastoji od:

- jedinstvenog identifikatora (engl. *identifier*) i
- naslova (engl. *title*).

Opciono može da sadrži, takođe, i:

- opis (engl. *description*),
- definiciju (engl. *definition*) (referenca ka drugoj definiciji ili repozitorijumu),
- *metadata* element, koji predstavlja dalje informacije o određenoj kompetenciji (može biti različitog formata, na primer IEEE LOM).

Listing 1 prikazuje RCD sinopsis uz korišćenje notacije kao u (LTSC, 2008).

```

reusable_competency_definition :
record
(
identifier :
long_identifier_type,
// Mandatory
// Occurs 1 time

title :
bag of langstring_type(1000),
// Mandatory
// Occurs 1 time
// SPM: 20 instances of langstring_type in the bag
// The parameter value is the SPM for the number
// of characters in the string element of the
// langstring_type

description :
bag of langstring_type(4000),
// Optional
// Occurs 0 or 1 times
// SPM: 20 instances of langstring_type in the bag
// The parameter value is the SPM for the number // of characters in the
string element of the
// langstring_type

definition :
definition_type,
// Optional
// Occurs 0 or more times
// SPM: 10 instances of definition in a
// reusable_competency_definition record

metadata :
metadata_type,
// Optional (implied default values - see 6.2.5)
// Occurs 0 or 1 times
)
  
```

Listing 1. RCD sinopsis

Iako su osnovni delovi ovog standarda (poput identifikatora, naslova i opisa) definisani u „ljudski čitljivoj formi“ i predstavljaju nestruktuirane podatke, moguće je obezbediti semantičko značenje standarda tako da postane razumljiv računarima. To je moguće postići dodavanjem posebnih „znanja“ u *metadata* element i vezama sa drugim RCD-ovima pomoću tih podataka u *metadata* delu standarda. S obzirom da *metadata* može biti definisan pomoću IEEE LOM standarda, za dodavanje semantike RCD-ovima je moguće iskoristiti sledeće IEEE LOM kategorije: *Relation i Classification* (posebno *taxonpath* polje). Takođe, semantika može biti obezbeđena i eksternim povezivanjem RCD-ova u ontologije, što je u ovom radu primenjeno.

RCD standard ne predviđa posebna polja o principima ocenjivanja postignuća učenika, primenjenim instrukcionim metodama, sertifikovanju kompetencija, i sl. što se može posmatrati kao mana primene ovog standarda za reprezentovanje kompetencija. Ipak, ukoliko za tim bude potrebe, model je moguće proširiti tako da i ovi aspekti budu zadovoljeni, jer RCD, donekle, omogućava definisanje navedenih polja u okviru struktuiranog opcionog elementa *definition* koji sadrži „nula ili više“ *model source* elemenata i jedan ili više *statement* elemenata tako da ih korisnici mogu „sami“ definisati kao, na primer: kriterijum, nivo osposobljenosti i sl. Međutim, cilj kreiranja predloženih ontoloških modela u ovom radu je predstavljanje kompetencija svršenih studenata kurikuluma za nastavnike informatike odnosno informatičkih kompetencija učenika po završetku srednje škole, te za takve modele navedena nedostajuća polja nisu značajna.

Zbog navedenog cilja, za reprezentovanja kompetencija u ovoj disertaciji nije korišćen IEEE draft standard Simple Reusable Competency Mappings (SRCM) (LTSC, 2006). Ovaj standard, u odnosu na IEEE RCD standard, ima dodatno direktni aciklični graf čvorova koji sadrže kompetencije, pri čemu svaki čvor ima nekoliko povezanih svojstava poput:

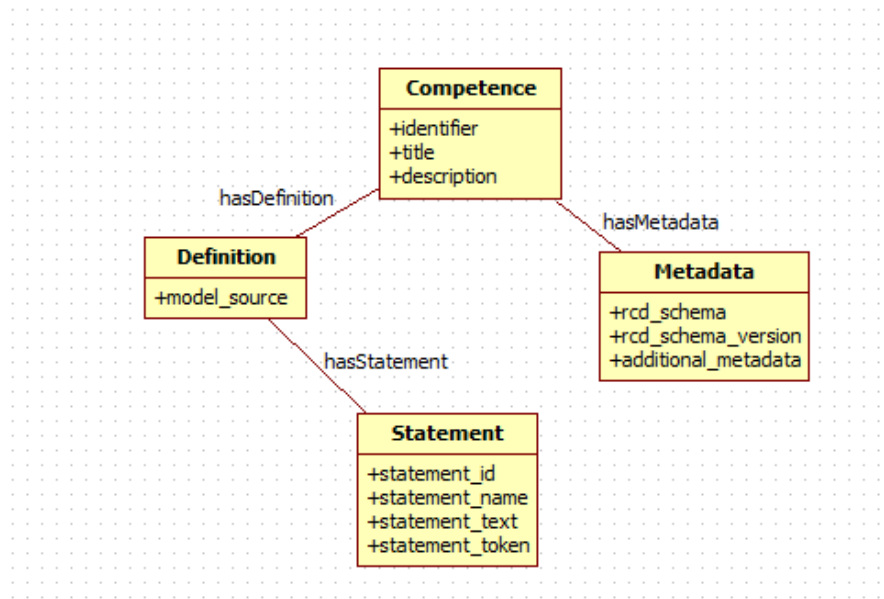
- kompetencija (RCD ili drugi SRCM),
- rezultat sposobnosti (potreban ili željeni),
- veze ka drugim čvorovima unutar grafa (roditelji ili deca) (Lundqvist et al., 2011).

Mana IEEE SRCM standarda je i to što trenutno predstavlja standard u fazi „nacrt“. Logičke veze unutar SRCM-a nisu bazirane na nekoj od formalnih logika čime nije omogućeno automatizovano razumevanje logike na kojoj je zasnovan. Takođe, neki autori (De Coi et al., 2007) smatraju da je semantika definisana u ovom modelu „zbunjujuća“.

4.3.3. Ontološka reprezentacija kompetencija zasnovana na IEEE RCD standardu

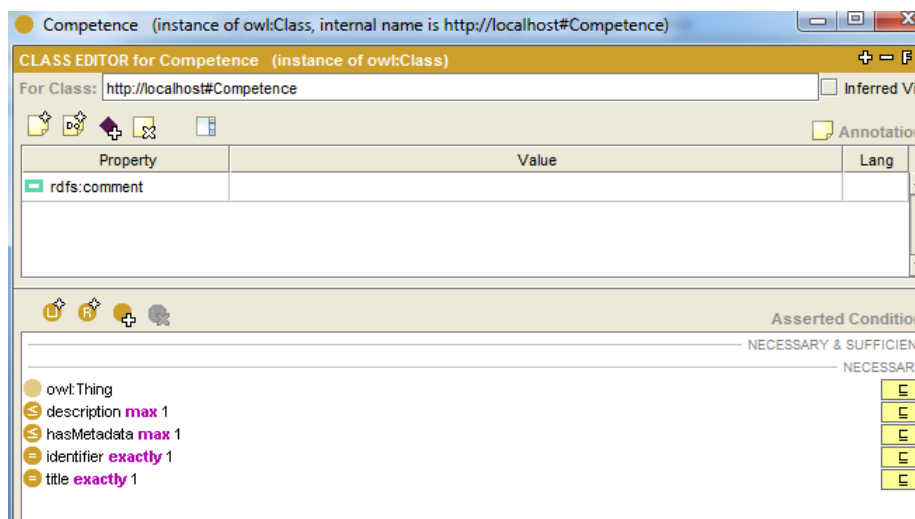
Način ontološkog reprezentovanja kompetencija zasnovanih na IEEE RCD standardu opisan je u (Mandić et al., 2013) i prikazan na slici 4. Sam model je sličan delu modela opisanom u (De Coi et al., 2007)

koji se odnosi na RCD kompetencije. Pravougaonici označavaju klase u ontologiji, pune linije objektna svojstva, dok se *datatype* svojstva nalaze unutar pravougaonika



Slika 4. Ontološka reprezentacija IEEE RCD standarda (preuzeto iz (Mandić et al, 2013))

Osnovnu klasu čini *Competence* sa svojstvima *identifier* (identifikator), *title* (naslov) i *description* (opis). S obzirom da su standardom ovi elementi definisani kao nestruktuirani, oni klasu opisuju u ontološkom modelu kao *datatype* svojstva. Posebne klase su *Statement*, *Metadata* i *Definition*. Opcioni element *definition* (slika 3, listing 1) predstavlja struktuiran deo kompetencije i može sadržati, ukoliko postoji, jedan ili više *statement* elemenata, poput ocene, kriterijuma, ishoda i sl., pa ga je potrebno predstaviti posebnom klasom *Definition* povezanom sa klasom *Statement* preko objektnog svojstva *hasStatement*. Slično, *metadata* element je opisan izdvojenom klasom *Metadata*. Kardinalnost svojstava u ontološkom modelu, kreiranom pomoću Protégé alata, odgovara opisanom standardu - pošto u IEEE RCD standardu kompetencije imaju tačno jedan identifikator i naslov a „0 ili jedan“ *metadata* element, objektna i *datatype* svojstva imaju kardinalnost kao na slici 5.



Slika 5. Kardinalnost svojstava klase *Competence*

S obzirom da ontološki model kurikuluma, predstavljen u ovom radu, reprezentuje kompetencije iz perspektive stečenih znanja i vještina, klase *Metadata*, *Definition* i *Statement* nisu dalje strukturirane u sadašnjoj verziji nastavnčkog i srednjoškolskog kurikuluma.

Dakle, više (“upper“) klase ontologije su:

- *Competence*,
- *Statement*,
- *Metadata*,
- *Definition*.

Analizom sadržaja i forme nastavnčkih kurikuluma dostupnih na web sajtovima institucija u više zemalja (Nemačka, Austrija, Turska i Republika Srbija) zapaženo je da su kompetencije vezane za svaki predmet (kurs) određene, pre svega, sa dva polja: „sadržaj premeta“ i „ishod predmeta“ (i/ili obrazovni ciljevi). U našem modelu kurikuluma „sadržaj premeta“ odgovara klasi *Knowledge*, a „ishod predmeta“ klasi *Skills*. Pojam „cilj predmeta“, koji postoji, na primer, u formularima za akreditaciju visokoškolskih studijskih programa u Republici Srbiji, nije mapiran u model, budući da analizirani svetski nastavnčki kurikulumi podrazumevaju definisanje samo sadržaja i očekivanih vještina u okviru svakog kursa. Druga polja predviđena kurikulumima (metode izvođenja nastave, ocene znanja) nisu trenutno obuhvaćena ontološkim modelom, budući da je cilj kreiranja modela definisanje kompetencija obezbeđenih kurikulumom. Ipak, definisani model omogućava mapiranje ovih polja korišćenjem potklasa klase *Definition* i *Statement*. Takođe, objekti učenja odgovarajući svakoj tematskoj oblasti ili pojedinačnoj temi bi mogli biti jednostavno uključeni u model putem potklasa klase *Metadata* i povezivanjem preko objektnog svojstva sa odgovarajućom tematskom oblašću.

Listing 2 prikazuje deo OWL koda predloženog ontološkog modela.

```

<owl:Class rdf:about="#Competence">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:FunctionalProperty rdf:ID="title"/>
      </owl:onProperty>
      <owl:cardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"
        >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:FunctionalProperty rdf:ID="identifier"/>
      </owl:onProperty>
      <owl:cardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"
        >1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:maxCardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"
        >1</owl:maxCardinality>
      <owl:onProperty>
        <owl:FunctionalProperty rdf:ID="description"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:ID="hasMetadata"/>
      </owl:onProperty>
      <owl:maxCardinality
rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger"
        >1</owl:maxCardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf
rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
</owl:Class>
<owl:Class rdf:ID="Skills">
  <rdfs:subClassOf rdf:resource="#Competence"/>
</owl:Class>

```

Listing 2. Deo OWL koda ontološkog modela kurikuluma

Taksonomija učenja predstavlja način za opisivanje različitih ponašanja u procesu učenja i karakteristika koje učenici u tom procesu treba da razviju (O'Neill & Murphy, 2010). Ona obezbeđuje strukturu za klasifikovanje obrazovnih ciljeva i ishoda pa je u ovom radu korišćena za reprezentovanje veština. Odabrana je Revidirana Blumova taksonomija (engl. *Revised Bloom's Taxonomy*) (Anderson et al., 2001) zbog toga što predstavlja dominantnu taksonomiju CS oblasti ali i uopšteno (Fuller et al., 2007).

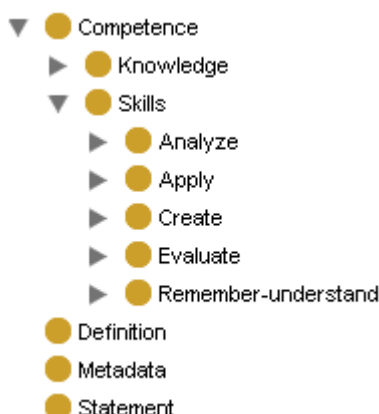
U tabeli 36 su prikazane kategorije dimenzije kognitivnih procesa revidirane Blumove taksonomije i njima odgovarajući kognitivni procesi (Anderson et al., 2001; Fuller et al., 2007).

Kategorija	Kognitivni procesi
1. Remember (zapamtiti)	upamtiti, prepoznavati, setiti se
2. Understand (razumeti)	shvatiti, interpretirati, ilustrovati, klasifikovati, sumirati, zaključiti, porediti, obrazložiti
3. Apply (primeniti)	koristiti, izvršiti, implementirati
4. Analyze (analizirati)	razlikovati, organizovati, “povezati sa uzrokom”
5. Evaluate (proceniti)	oceniti, proveriti, kritikovati
6. Create (kreirati)	razviti, generisati, planirati, proizvesti

Tabela 36. Revidirana Blumova taksonomija (Fuller et al., 2007)

Kategorije su u (Anderson et al., 2001) hijerarhijski poredane u odnosu na njihovu kompleksnost, pa se podrazumeva da su kognitivni procesi koji odgovaraju 'Remember' kategoriji manje kompleksni od 'Understand' kognitivnih procesa, 'Understand' od 'Apply' itd. Izabrana taksonomija, po (Fuller et al., 2007), ne definiše sekvencu instrukcija, što za reprezentovanje „statičkih“ kurikuluma poput predstavljenog modela u ovom radu nije ni značajno, već nivoe izvođenja za svaki predstavljeni sadržaj.

Potklase klase *Skills* su u ovom radu predstavljane klasama koje odgovaraju kategorijama dimenzije kognitivnih procesa revidirane Blumove taksonomije, s tim što su kategorije 'Remember' i 'Understand' predstavljene jednom klasom *Remember-understand* zbog prirode domenske CS oblasti u kojoj je ishod učenja koji podrazumeva samo prepoznavanje/pamćenje bez razumevanja malo verovatan. Tako su potklase klase *Skills* sledeće klase: *Remember-understand*, *Apply*, *Analyze*, *Evaluate*, *Create* (slika 6).



Slika 6. Potklase klase *Skills*

Autori revidirane Blumove taksonomije ukazuju da je moguće preklapanje između pojmova koji odgovaraju kognitivnoj kompleksnosti, posebno između viših nivoa kognitivne hijerarhije što

se u ontološkom modelu može reprezentovati izostavljanjem deklaracije „disjoint“ (disjunktnosti) za klase.

4.3.4. WordNet leksička baza podataka za engleski jezik

Budući da je krajnji cilj disertacije utvrđivanje usaglašenosti kurikuluma primenom poluautomatizovane softverske platforme, ontološke modele je neophodno kreirati tako da, u što je moguće većoj meri, sadrže pojmove koji su “razumljivi računarima”. Dakle, pored računarske interpretativnosti strukture i relacija modela kurikuluma koja je obezbeđena primenom principa Semantičkog Weba (ontologija), važno je obezbediti i računarsku interpretativnost značenja pojedinačnih pojmova sadržanih u modelima kurikuluma (pojmovi sadržanih u nazivima klasa, labela, nazivima svojstava itd.). Time bi se omogućilo da se u softverskoj platformi za poređenje reči (pored primene algoritama kojima se reči analiziraju samo kao niz karaktera) koriste i algoritmi kojima se utvrđuje sličnost *značenja* reči. U disertaciji je računarska interpretativnost značenja reči obezbeđena primenom WordNet rečnika.

WordNet predstavlja obimnu elektronsku leksičku bazu podataka strukturiranu u vidu semantičke mreže koja povezuje reči i grupu reči (Fellbaum, 2006). Struktura WordNet-a čini WordNet korisnim alatom za računarsku lingvistiku i obradu prirodnog jezika. Po (Krstev et al., 2008) WordNet se danas može smatrati *de facto* standardom u ovoj oblasti.

U WordNet-u fokus je više na značenju reči nego na samoj formi, iako je morfološki aspekt uključen (Lin & Sandkuhl, 2008). Imenice, glagoli, pridevi i prilozi su grupisani u skupove kognitivnih sinonima (Synset-i) koji predstavljaju različite koncepte (Princeton University, 2010). Synset-i su međusobno povezani konceptualno-semantičkim i leksičkim vezama. Osnovna veza među rečima u WordNet-u je, dakle, relacija sinonima koja označava da su dve reči “zamenljive”, odnosno da imaju slično ili identično značenje. Između synseta su moguće relacije hipernima/hiponima (“is-a” hijerarhija/taksonomija kojom su povezani superkoncepti/potkoncepti), meronima (“deo-celina” relacija). Pridevi su strukuirani pomoću antonima (reči sa suprotnim značenjem). Glagoli su organizovani u hijerarhije tako da glagoli na dnu hijerarhije (troponimi) odgovaraju sve specifičnijim radnjama karakterističnim za neki događaj. WordNet, takođe, obezbeđuje tekstualni opis konceptata (engl. *gloss*) i primere korišćenja.

Prva verzija WordNet-a predstavljala je elektronsku leksičku bazu podataka za engleski jezik i razvijena je 1986. godine na Princeton Univerzitetu, gde je i unapređivana tokom narednih godina (Fellbaum, 2006). Engleska verzija WordNet baze predstavljala je osnovu za razvoj leksičkih baza podataka na drugim jezicima. WordNet baza je javna i slobodna za preuzimanje i, postoje (posebno za engleski jezik) odgovarajuće biblioteke koje je moguće iskoristi pri razvoju softverskih aplikacija. Budući da su dostupni standardi i preporuke međunarodnih organizacija većinom napisani na engleskom jeziku, svi pojmovi sadržani u ontološkim modelima kurikuluma, predstavljeni u

narednim sekcijama ove disertacije, napisani su na engleskom jeziku i pri poređenju pojmova (poglavlje 5) korišćena je engleska verzija WordNet baze podataka.

4.3.5. Ontološki model srednjoškolskog informatičkog kurikulumu

U poglavlju 2 ove disertacije dat je detaljan pregled stanja u oblasti informatičkog obrazovanja iz kog se vidi da, za razliku od visokog obrazovanja kojim se školuju nastavnici informatike, u domenu srednjoškolskog informatičkog obrazovanja postoji jasno prepoznata potreba i rezultati standardizacije kurikulumu. Na osnovu detaljnog opisa aktuelnih školskih informatičkih standarda (sekcije 2.1. i 2.1.1.) može se zaključiti da ACM K12 predlog standarda (CSTA Standards Task Force, 2011) predstavlja najkompletniji i najsavremeniji predlog modela informatičkog obrazovanja za osnovnoškolski i srednjoškolski nivo. Stoga je ontološki model srednjoškolskog informatičkog kurikulumu u ovom radu kreiran na osnovu ACM K12 CS predloga kurikulumu, s tim što su u model mapirana samo ona znanja koja su predviđena za nivo obrazovanja koji odgovara srednjoškolskom (K8 ili viši nivoi korišćenog standarda). Tri generalna nivoa (L1, L2, L3) ACM K12 CS standarda su odvojeno detaljno opisana u (Frost et al., 2009), (Verno et al., 2006) i (Madden et al., 2007). Ovi dokumenti su strukturirani tako da su podeljeni u 12, 10 i 14 tema, respektivno. Svaka tema (oblast) sadrži uopšten opis, kratak navod opreme za podršku nastave, predviđeno vreme za realizovanje oblasti, detaljno opisane ciljeve učenja, preporučen način ocenjivanja, fokuse svake oblasti i predlog za realizaciju svakog fokusa.

Na slici 7 je prikazan deo teme 'Programming Languages' L2 nivoa ACM-ovog predloga (strane 25 i 26 (Verno et al., 2006)).

Topic 11: Programming Languages

Topic Description:

Programming Languages will introduce the student to some basic issues associated with program design and development. The focus of this unit is to establish an appreciation of the work being done by software.

Textbooks and Supplies:

A programming language; interactive development environment recommended.

Time to Complete: 2-4 weeks

Student Learning Objectives	Assessment Measures
<i>The student will be able to:</i>	
1. Code, test, and execute a program that corresponds to a set of specifications.	Lab activity
2. Convert a word problem into code using top-down design.	Written activity Lab activity
3. Select appropriate data types.	Written activity Lab activity
4. Write structured program code.	Lab activity
5. Draw a series of diagrams showing the scope and values of variables during execution of a simple program.	Written activity

Assessment Recommendations:

An average of 60% from combined assessment measures is required to demonstrate proficiency in course material.

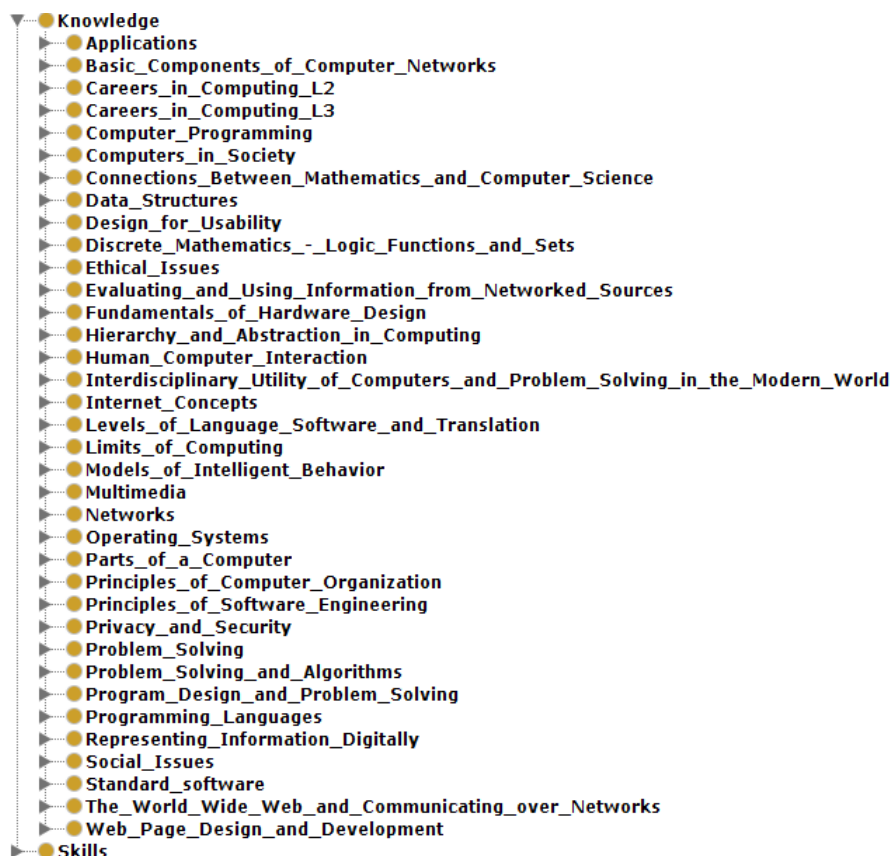
Lab activities	50%
Written activities, including tests, quizzes, and written assignments	50%

Detailed Outline	
Focus	Sample Lab / Hands-on Activity
1. Terminology	Identify and define key terms associated with programming.
2. Representation of text inside the computer	Each student writes a sentence in binary and exchanges it with a neighbor. The neighbor translates the sentence into text. Students stand or sit to mime a secret word in binary. Flashlights can also be used to represent binary code.
3. Representation of numbers inside the computer, including the largest and smallest values	Numbers are placed into imaginary bytes in a grid, each imaginary byte having a unique address. (A spreadsheet can be used for this purpose.) Instructions are provided to add and subtract values by

Slika 7. Tema 'Programming languages' (Verno et al., 2006)

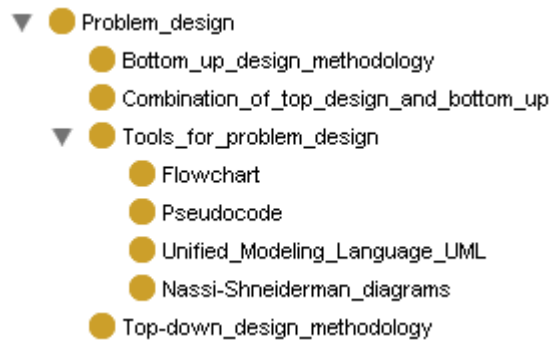
Ontološki model srednjoškolskog informatičkog kurikuluma kreiran je u dve faze (Mandić et al., 2015).

U **prvoj fazi** generisanja ontološkog modela kurikuluma svaka od 36 tema sva tri nivoa je modelovana kao potklasa klase *Knowledge* (slika 8). Fokus koji pripadaju temama su modelovani kao potklase klase u koje su mapirane teme, odnosno teme su modelovane kao superklase klase koje odgovaraju fokusima koji im pripadaju.



Slika 8. Potklase klase *Knowledge* nakon prve faze kreiranja srednjoškolskog ontološkog modela

Deo sadržaja koji pripadaju koloni „Sample Lab/Hands-on Activity“ (slika 7), u kojoj su opisani načini realizacije određenog fokusa, mapirani su, takođe, u model. Tako, ukoliko u ovom polju postoje pojmovi (sadržaji) koji dodatno opisuju fokus, oni su mapirani na labelu klase koja reprezentuje fokus; ako su ti sadržaji predstavljeni u formi nabiranja, oni su mapirani na potklase klase koja odgovara fokusu. Karakterističan primer je fokus ‘Problem design’ teme ‘Problem solving’ (strana 9 (Verno et al., 2006)). Realizacija ovog fokusa predviđa da učenici, pri kreiranju algoritama, koriste dizajn metodologije (“od vrha ka dnu” pristup, “od dna ka vrhu” pristup, kombinaciju pristupa), dok se za predstavljanje algoritama predviđa korišćenje sledećih “standardnih dizajn alata”: dijagram toka, pseudokod, objedinjeni jezik modeliranja (UML) i Nassi-Shneiderman dijagrami. Način mapiranja fokusa ‘Problem design’ i navedenih sadržaja prikazan je na slici 9.



Slika 9. Struktura klase *Problem_design*

Ciljevi učenja definisani u (Frost et al., 2009; Madden et al., 2007; Verno et al., 2006) su mapirani na odgovarajuće potklase klase *Skills* u skladu sa dimenzijom kognitivnih procesa revidirane Blumove taksonomije. Pri utvrđivanju kojoj generalnoj potklasi klase *Skills* određeni ishod pripada analizirana su značenja kategorija revidirane Blumove taksonomije prikazana u tabeli 36, kao i detaljniji opisi revidirane Blumove taksonomije dati u (Heer, 2012; Krathwohl, 2002) i digitalna taksonomija prikazana u (Churches, 2007). Navedeni izvori sadrže veliki broj glagola/sinonima za kategorije revidirane Blumove taksonomije koji su se pokazali dovoljnim za jednoznačno mapiranje ciljeva učenja na odgovarajuću potklasu klase *Remember-understand, Apply, Analyze, Evaluate* i *Create*.

Ciljevi učenja su povezani objektnim svojstvom *hasKnowledge* sa odgovarajućom temom kojoj, po ACM referencama/dokumentima, pripadaju.

Pri mapiranju ciljeva uzeta su u obzir moguća preklapanje između pojmova koji opisuju različite nivoe revidirane Blumove taksonomije. Ilustrovaćemo to na primeru mapiranja cilja 'Select appropriate data types' (selektovati odgovarajuće tipove podataka) teme 'Programming Languages' (slika 7) u odgovarajuću potklasu klase *Analyze*. Razmatrajući značenje navedene veštine, sinonim za "selektovati odgovarajuće" bi mogao biti glagol „pronaći“ (engl. „find“) koji pripada 'Analyze' kategoriji revidirane Blumove taksonomije. S druge strane, gledajući samo glagole koji određuju kategorije digitalne Blumove taksonomije, navedene u (Churches, 2007), ova veština bi mogla biti i potklasa klase *Remember-understand* jer se reč „find“ može pronaći i u listi sinonima za kategoriju 'Remember'. Međutim, cilj 'Select appropriate data types' (Verno et al., 2006), u kontekstu programiranja, predstavlja suštinski viši nivo revidirane Blumove taksonomije od 'Remember' i 'Understand'. To se može zaključiti i na osnovu detaljnijeg opisa primera na stranama 51, 54 i 55 izvora (Churches, 2007). Takođe, po (Mayer, 2002) (strana 230) „*Differentiating (also called discriminating, selecting, distinguishing, or focusing)*“ pripada 'Analyze' kategoriji. U prilog mapiranju cilja 'Select appropriate data types' u 'Analyze' kategoriju ide i to što ishodi učenja opisani akcionim glagolima: „compare“ („porediti“), „contrast“ („razlikovati“), „select“ („selektovati“) pripadaju 'Analyze' kategoriji i po originalnoj („nerevidiranoj“) Blumovoj taksonomiji.

U tabeli 37 su prikazani još neki primeri mapiranja ciljeva učenja iz ACM K12 kurikuluma u odgovarajuće potklase klasa revidirane Blumove taksonomije.

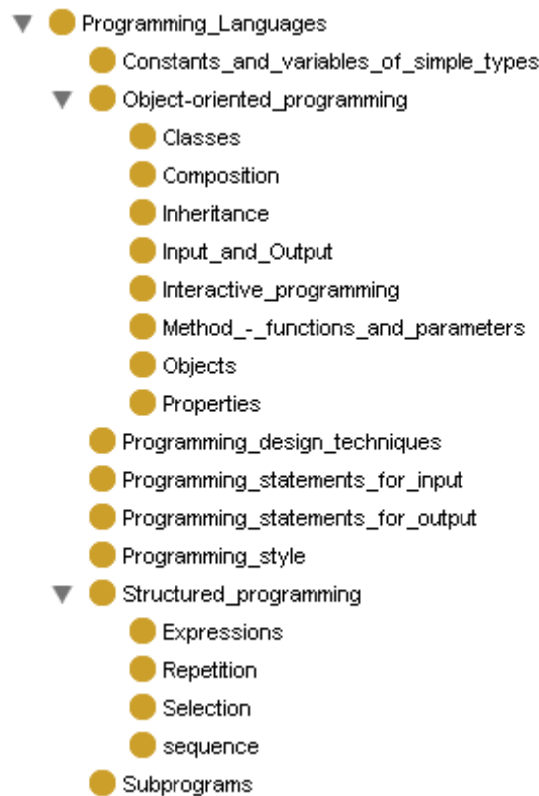
Klase revidirane Blumove taksonomije	Skills potklase
<i>Remember-understand</i>	<i>Describe_at_least_one_problem_computers_cannot_solve</i> (Opisati najmanje jedan problem koji računar ne može da reši)
<i>Apply</i>	<i>Connect_a_computer_to_a_network</i> (Povezati računar u mrežu)
<i>Analyze</i>	<i>Analyze_the_degree_to_which_a_computer_model_accrately_represents_the_real_world</i> (Analizirati stepen do kog računarski model tačno reprezentuje stvarni svet)
<i>Evaluate</i>	<i>Test_and_execute_a_program_that_corresponds_to_a_set_of_specifications</i> (Testirati i pokrenuti program koji odgovara skupu specifikacija)
<i>Create</i>	<i>Write_a_computer_program_that_implements_an_algorithm</i> (Napisati računarski program koji implementira algoritam)

Tabela 37. Primeri kategorisanja Skills potklasa srednjoškolskog modela kurikuluma

Jedan primer mapiranja teme u ontološki model srednjoškolskog informatičkog kurikuluma u prvoj fazi kreiranja modela je dat u nastavku. Tema 'Programming Languages' (slika 7) je mapirana u klasu *Programming_Languages* - potklasu klase *Knowledge*, dok su njeni fokusi mapirani na potklase klase *Programming_Languages*. Tako je fokus 'Representation of text inside the computer' mapiran na klasu *Representation_of_text_inside_the_computer*, potklasu klase *Programming_Languages*. Analogno, fokus 'Programming design techniques' je reprezentovan klasom *Programming_design_techniques*, koja je struktuirana kao potklasa klase *Programming_Languages*. Cilj 'Write structured program code' je, u skladu sa revidiranom Blumovom taksonomijom, modelovan kao klasa *Write_structured_program_code* - potklasa klase *Create*. Klasa *Write_structured_program_code* je, dalje, povezana preko *hasKnowledge* objektnog svojstva sa klasom *Programming_Languages*.

U **drugo** fazi kreiranja modela kurikuluma 36 tema sva tri nivoa su svrstana u 13 generalnih oblasti. Svrstavanje je izvršeno na sledeći način. Srodne teme (teme koje se preklapaju) su mapirane na jednu klasu i/ili su mapirane kao potklase zajedničke superklase. Analogno,

srodni fokusi različitih tema su mapirani na potklase jedne superklase. Karakterističan primer za to su sledeće teme: ‘Computer Programming’, ‘Programming Languages’, ‘Program Design and Problem Solving’, ‘Problem solving’ i ‘Problem Solving and Algorithms’. Tako su teme ‘Computer Programming’ i ‘Programming Languages’ mapirane na jednu roditeljsku klasu *Programming_Languages*. Na potklase klase *Programming_Languages* su mapirani i fokusi teme ‘Program Design and Problem Solving’ koji se odnose na oblast programiranja (poput fokusa: ‘Interactive programming’, ‘Method (functions) and parameters’, ‘Objects and classes’) (slika 10). Drugi deo fokusa teme ‘Program Design and Problem Solving’, koji se odnose na oblast rešavanja problema, su mapirani na potklase roditeljske klase *Problem_solving*. Klasa *Problem_solving*, dalje, predstavlja potklasu klase *Problem_Solving_and_Algorithms*.

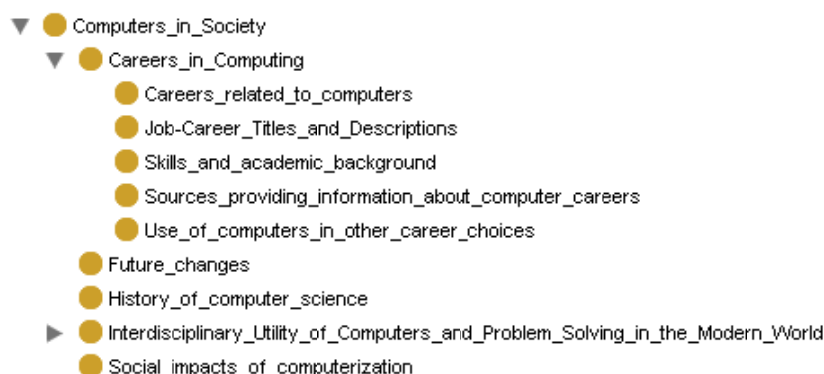


Slika 10. Hijerarhijska struktura klase *Programming_Languages*

Sa slike 10 se može zapaziti da su fokusi ‘Object-oriented programming’ i ‘Structured programming’ reprezentovani posebnim roditeljskim klasama. Tako su na potklase klase *Object-oriented_programming* mapirani fokusi koji se mogu grupisati u oblast objektno-orijentisanog programiranja.

Takođe, detaljnom analizom tema utvrđeno je da su pojedini fokusi, obično nižeg nivoa ACM K12 kurikuluma (L1 ili L2), predstavljeni posebnim srodnim temama u L2 ili L3. Tada su ti fokusi mapirani na superklasu na koju je mapirana i srodna tema. Primer je fokus ‘Computing careers’ L1 nivoa koji je mapiran na superklasu

Careers_in_Computing, budući da u L2 i L3 postoji posebna tema ‘Careers in Computing’ (slika 11).



Slika 11. Hijerarhijska struktura klase *Careers_in_Computing*

Takođe, ukoliko su se neki fokusi ponavljali (poput, na primer, nizova) u više tema, oni su mapirani na jednu klasu.

Konačno, kreirane su superklase (*Programming_and_software_engineering*, *Ethical_and_security_Issues*, *Algorithms_and_data_structures*) na čije su direktne potklase mapirane roditeljske klase koje reprezentuju srodne tematske oblasti. Tako su klase *Data_Structures* i *Problem_Solving_and_Algorithms* postale potklase superklase *Algorithms_and_data_structures*. Na taj način je dobijena konačna lista potklasa klase *Knowledge* prikazana na slici 12.



Slika 12. Potklase klase *Knowledge* srednjoškolskog modela

Važno je napomenuti da najnovija aktuelna integralna verzija ovog standarda (CSTA Standards Task Force, 2011) (postoji i noviji draft) nije neposredno mapirana u računarski model, budući da ne sadrži dovoljno eksplicitnih informacija o potrebnim oblastima i znanjima kurikuluma koje bi omogućile konzistentno mapiranje u *Knowledge* potklase ontološkog modela. Predlog kurikuluma, prikazan u (CSTA Standards Task Force, 2011), definiše prvenstveno ciljeve učenja učenika po završetku specifičnih nivoa K12 kurikuluma. O predviđenim sadržajima, na osnovu tog dokumenta, je moguće samo posredno zaključiti. Tako su u ovom radu, pre svega, iskorišćena

dokumenta (Frost et al., 2009), (Verno et al., 2006) i (Madden et al., 2007). Ipak svi ciljevi definisani u (CSTA Standards Task Force, 2011) su analizirani i ako neki od njih nije obuhvaćen u (Frost et al., 2009; Verno et al., 2006; Madden et al., 2007) tada je taj cilj dodat modelu kao *Skills* potklasa i povezan sa adekvatnom tematskom oblašću (potklasom klase *Knowledge*). S obzirom na nešto raniji datum objavljivanja pojedinačnih dokumenata (Frost et al., 2009; Verno et al., 2006; Madden et al., 2007) to su obično oni ciljevi koji se odnose na savremenije principe i tehnologije i alate poput cilja 'Use mobile devices/emulators to design, develop, and implement mobile computing applications' (Korišćenje mobilnih uređaja/emulatora za dizajn, razvoj i implementiranje mobilnih računarskih aplikacija). Rad sa mobilnim aplikacijama se ne pominje u ranijim izdvojenim dokumentima. Još neki primeri ciljeva učenja iz (CSTA Standards Task Force, 2011) dodatih u model su:

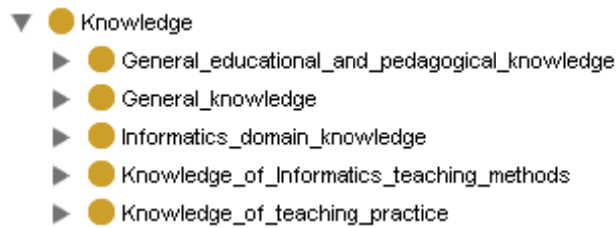
- 'Evaluate what kinds of problems can be solved using modeling and simulation' (Proceniti koje vrste problema mogu biti rešene primenom modelovanja i simulacija),
- 'Evaluate algorithms by their efficiency, correctness, and clarity' (Oceniti algoritme po njihovoj efikasnosti, tačnosti i jasnoći),
- 'Use Application Program Interfaces (APIs) and libraries to facilitate programming solutions' (Koristiti API-je i biblioteke da bi se olakšalo programiranje),
- 'Demonstrate concurrency by separating processes into threads and dividing data into parallel streams' (Demonstrirati konkurentnost razdvajanjem procesa u niti i deljenjem podataka u paralelne tokove).

Ontološki model nalazi se na adresi <http://www.pef.uns.ac.rs/SecondaryInformaticsCurriculum/index.html>

4.3.6. Ontološki model referentnog kurikulumu za nastavnike informatike

Budući da u domenu obrazovanja nastavnika informatike ne postoje predlozi standardizovanih modela kurikulumu, ontološki model referentnog nastavničkog kurikulumu je kreiran na osnovu istraživanja kurikulumu srednjoškolskih nastavnika informatike prikazanog u drugom poglavlju ove disertacije. Pri kreiranju modela referentnog nastavničkog kurikulumu, dakle, uzeti su u obzir sadržaji reprezentativnih nastavničkih kurikulumu iz osam zemalja (Nemačka, Austrija, Izrael, Estonija, Turska, Škotska, USA i Republika Srbija), kao i analizirana literatura i preporuke međunarodnih organizacija. Pri tome je iz analize izuzet izabrani kurikulum iz Republike Srbije, koji će biti reprezentovan posebnim ontološkim modelom (sekcija 4.3.7).

Pet generalnih oblasti koje svaki program za pripremu nastavnika informatike treba da sadrži, detaljno opisanih u sekciji 2.6., su modelovane potklasama klase *Knowledge* (slika 13).



Slika 13. Potklase klase *Knowledge* nastavničkog kurikuluma

Slika 13 ukazuje na različitu strukturu *Knowledge* potklasa modela referentnog nastavničkog kurikuluma i modela srednjoškolskog informatičkog kurikuluma (slika 12). U srednjoškolskom modelu (direktne) potklase *Knowledge* klase reprezentuju samo informatičke tematske oblasti, jer srednjoškolski informatički kurikulum ne podrazumeva izučavanje opštih obrazovnih i pedagoških znanja, znanja nastavne prakse, metodike nastave informatike i opštih znanja. Ta znanja su specifična za nastavničke kurikulume.

Pri definisanju potklasa klase *Knowledge* za kategorizaciju viših klasa informatičkih domena iskorišćene su klasifikacije prikazane u (ACM, 2008; 2012; Cassel et al., 2008).

Ontološki model sadrži sve oblasti informatičkih (domenskih) znanja za koje je analiza (data u sekciji 2.7.) pokazala da su sadržane u većini analiziranih kurikuluma. S druge strane, oblasti koje su sadržane u nekim kurikulumima a nisu uvrštene u model su one oblasti koje se pojavljuju u samo jednom ili najviše 2 kurikuluma i ne mogu se klasifikovati kao neophodna znanja za nastavnike informatike. Model, dakle, ne predstavlja sve oblasti sadržane u svim kurikulumima (uniju), ali nije u potpunosti nastao ni primenom preseka svih kurikuluma. Razlog što ni potonji pristup nije primenjen je to što se pokazalo da neki kurikulumi ne sadrže važne oblasti programa za pripremu nastavnika a koje postoje u većini drugih kurikuluma. Teme i potklase nižih nivoa ontološkog modela su odabrane na isti način (pojavljuju se u većini analiziranih kurikuluma).

U ontološkom modelu referentnog nastavničkog kurikuluma dodatni opis klasa, ukoliko postoji, mapiran je na labele.

Takođe, sadržaj referentnog kurikuluma je usaglašen sa istraživanjem sprovedenim nad nastavnicima informatike u srednjim školama (Mandić et al., 2014) o značaju i zastupljenosti svih predloženih tema nastavničkog informatičkog kurikuluma (poglavlje 3). Sve oblasti i teme za koje su ispitani nastavnici informatike iskazali da su im značajne za realizovanje nastave su obuhvaćene predloženim modelom.

Na slici 14 prikazane su više klase informatičkih (domenskih) znanja.



Slika 14. Deo ontološke reprezentacije referentnog nastavničkog kurikuluma

Potklase klase *Skills* su kreirane, pre svega, na osnovu ISTE/NCATE standarda navedenih u (East et al., 2011; ISTE, 2011) kao i detaljnog opisa ranije verzije ovog standarda za obrazovanje srednjoškolskih CS nastavnika prikazanog u (ISTE, 2002). Takođe, pri izboru nastavničkih veština koje je potrebno mapirati u referentni nastavnički model uzeti su u obzir i ishodi/ciljevi kurseva sadržani u analiziranim nastavničkim kurikulumima. Odabrane nastavničke veštine su na osnovu (Anderson et al., 2001; Churches, 2007; Heer, 2012; Krathwohl, 2002), klasifikovane u odgovarajuće potklase klase revidirane Blumove taksonomije i povezane sa znanjima na koja mogu da se odnose. Tako klasa *Teach* predstavlja potklasu klase *Apply* i sadrži potklase koje predstavljaju veštine realizovanja nastave prilagođene individualnim potrebama učenika, predavanja lekcija uz primenu kolaborativnog pristupa, držanje nastave uz korišćenja raznih oblika medija, učešće u nastavi na srednjoškolskom nivou i slično. Korišćenje različitih softverskih i razvojnih okruženja, korišćenje hardvera, operativnih sistema i sl. su, takođe, reprezentovani potklasama klase *Apply*. Klasi *Create* u najvećoj meri pripadaju potklase koje odgovaraju veštini dizajniranja, programiranja, sposobnosti razvoja plana nastave, plana samoevaluacije, profesionalnog razvoja i sl. Primer potklase klase *Analyze* je *Analyze_results_of_assessment_to_improve_instructional_practice* (Analizirati rezultate ocenjivanja da bi se unapredila instrukciona praksa). Potklase klase *Evaluate* odnose se na sposobnost ocenjivanja realizovane aktivnosti, testiranja softvera, procene ograničenja računara i mogućnosti primene veštačke inteligencije i sl. *Remember - understand* veštine obuhvataju širok dijapazon tema koje nastavnici treba da prepoznaju, nabroje i razumeju.

U tabeli 38 prikazani su primeri veština referentnog nastavničkog kurikuluma mapiranih u odgovarajuće potklase klasa revidirane Blumove taksonomije.

Klase revidirane Blumove taksonomije	<i>Skills</i> potklase
<i>Remember-understand</i>	<i>List_problem_solving_phases</i> (Nabrojati faze rešavanja problema)
<i>Apply</i>	<i>Use_learning_environments</i> (Koristiti obrazovna okruženja)
<i>Analyze</i>	<i>Analyze_algorithms_using_complexity_efficiency_aesthetics_and_correctness</i> (Analizirati algoritme primenom složenosti, efikasnosti, estetike i tačnosti)
<i>Evaluate</i>	<i>Estimate_impact_of_the_hardware_components_on_the_computer_performance</i> (Proceniti uticaj hardverskih komponenti na performanse računara)
<i>Create</i>	<i>Design_programs_in_languages_from_two_different_programming_paradigms_in_a_manner_appropriate_to_each_paradigm</i> (Dizajnirati programe u jezicima iz dve različite programerske paradigme na način prilagođen svakoj paradigmi)

Tabela 38. Primeri kategorisanja *Skills* potklasa referentnog nastavničkog kurikuluma

Jedna *Knowledge* potklasa odgovarajuća tematskoj oblasti ili kursu (direktna potklasa neke od klasa koje reprezentuju pet generalnih oblasti nastavničkog kurikuluma, prikazanih na slici 13) najčešće je u vezi sa više potklasa klase *Skills*, budući da većina kurseva u analiziranim nastavničkim kurikulumima predviđa više ishoda učenja.

Inverzno, jedna *Skills* potklasa je ređe u vezi preko objektnog svojstva *hasKnowledge* sa više *Knowledge* potklasa.

Ontološki model referentnog nastavničkog kurikuluma nalazi se na adresi www.pef.uns.ac.rs/InformaticsTeacherEducationCurriculum/index.html.

4.3.7. Ontološki model kurikuluma za nastavnike informatike iz Republike Srbije

U cilju poređenja sa opisanim ontološkim reprezentacijama srednjoškolskog informatičkog kurikuluma i referentnog nastavničkog informatičkog kurikuluma (sekcije 4.3.5. i 4.3.6.) kreirana je ontološka reprezentacija kurikuluma integrisanih akademskih studija „Tehnika i informatika“ Fakulteta tehničkih nauka u Čačku.

Integrisane studije daju zvanje diplomirani profesor tehnike i informatike-master. Ovaj studijski program je odabran iz više razloga: detaljne informacije o studijskom programu dostupne su na zvaničnom sajtu; integrisane studije traju 5 godina i daju zvanje mastera što je u skladu sa našim preporukama opisanim u poglavlju 2; studije su akreditovane u oba akreditaciona procesa realizovana u Republici Srbiji; “studijski program je utemeljen na tradiciji školovanja profesora tehničkih disciplina na Fakultetu tehničkih nauka u Čačku” i “nadovezuje se na studijske programe za obrazovanje profesora tehničkog obrazovanja (realizovan od 1975. godine), profesora tehnike i informatike (realizovan od 1993. godine), profesora informatike (realizovan od 2006. godine), a koji su 2009. godine integrisani u ovaj studijski program” (standard 1) (Fakultet tehničkih nauka, 2013).

Više klase ontološke reprezentacije izabranog kurikuluma iz Republike Srbije su, poput prethodnih ontoloških modela, modelovane u skladu sa ontološkom reprezentacijom kompetencija zasnovanom na IEEE RCD standardu (slike 4 i 6). Takođe, poput ontološkog modela nastavnčkog kurikuluma prikazanog u sekciji 4.3.6., klasa *Knowledge* ima 5 direktnih potklasa (slika 13).

U ontološki model su mapirani svi predmeti specificirani u standardu 5 akreditacionog materijala datog na zvaničnom sajtu institucije (Fakultet tehničkih nauka, 2013). Pri tome su u model mapirani i obavezni i izborni predmeti. Predmeti su kategorisani u jednu od 5 generalnih oblasti (*Opšta znanja, Opšta obrazovna i pedagoška znanja, Informatička domenska znanja, Znanja nastavne prakse i Znanja metodike nastave informatike*). Tako su u *Opšta znanja* klasifikovani:

- matematički kursevi (Matematika 1, Matematika 2, Diskretna matematika i sl.),
- fizika,
- engleski jezik i
- kursevi tehničkog domena koji ne sadrže informatičke sadržaje.

Informatičkim domenskim znanjima pripadaju:

- informatički predmeti i
- kursevi tehničkog domena relevantni za nastavu informatike u srednjoj školi.

Znanjima nastavne prakse pripada stručna školska praksa 1, 2 i 3.

Znanjima metodike nastave informatike pripadaju:

- kursevi metodike informatike i tehnike,
- metodički praktikum iz kompjuterske grafike,
- metodički praktikum iz robotike,
- metodički praktikum iz osnova programiranja,
- metodički praktikum iz arhitekture i organizacije računara.

Opštim obrazovnim i pedagoškim znanjima pripadaju kursevi:

- komunikologije,
- obrazovne tehnologije,
- pedagogije,

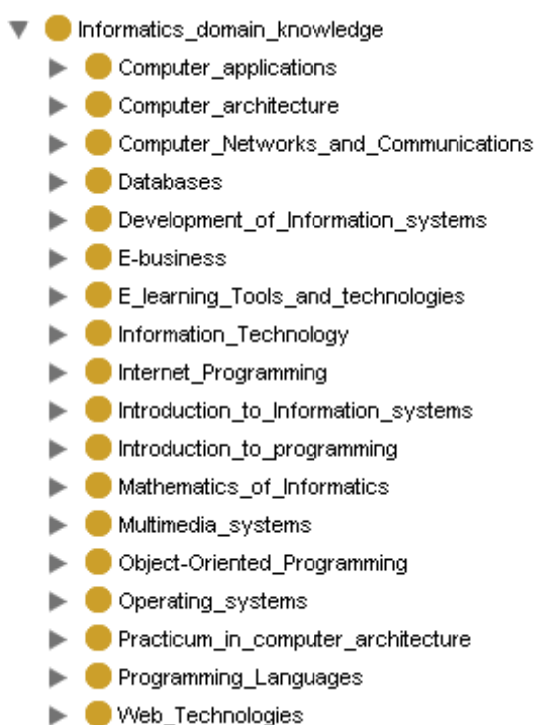
- psihologije,
- dokimologije.

Kategorizacija nedomenskih znanja može se videti i iz tabele 7 (sekcija 2.7.2).

Princip mapiranja kurseva kurikuluma u ontološki model je bio sledeći. Zbog potrebe (sekcija 4.3.4) da termini sadržani u kurikulumima budu, u što većoj meri, “računarski razumljivi” i uporedivi sa pojmovima sadržanim u drugim kreiranim kurikulumima, svi nazivi predmeta, kao i ishodi i sadržaji predmeta u kurikulumu za diplomiranog profesora tehnike i informatike-mastera, Fakulteta tehničkih nauka u Čačku, su prevedeni, najpre, na engleski jezik.

Nazivi kurseva, u zavisnosti od domena u koji su klasifikovani, su mapirani na potklase klase koje reprezentuju odgovarajuće generalne oblasti kurikuluma. Tako su, na primer, kreirane klase *Physics*, *Probability_and_Statistics* i mapirane kao potklase klase *General_knowledge*; kreirane su klase *Pedagogy*, *Psychology*, koje su postale potklase klasi *General_educational_and_pedagogical_knowledge* itd.

Potklase klase koja odgovara informatičkim domenskim znanjima prikazane su na slici 15.



Slika 15. Više klase informatičkih (domenskih) znanja nastavničkog kurikuluma iz Republike Srbije

U narednom koraku kreiranja ontološkog modela teme opisane u sadržaju predmeta su reprezentovane kao potklase klase koje odgovaraju predmetima (kursevima). Ukoliko neka tema sadrži više drugih tema (najčešće predstavljeno u formi nabiranja), tada su sve nabrojane teme mapirane kao njene potklase. Takođe, ako je neka

tema dodatno opisana, dodatno pojašnjenje je mapirano na njenu labelu.

U poslednjoj fazi kreiranja ontološkog modela ishodi predmeta su u skladu sa (Anderson et al., 2001; Churches, 2007; Heer, 2012; Krathwohl, 2002) i opisom datim u sekciji 4.3.3. mapirani u odgovarajuće potklase klasa revidirane Blumove taksonomije i povezani preko *hasKnowledge/hasSkills* objektnih svojstava sa klasom koja odgovara predmetu kome pripadaju.

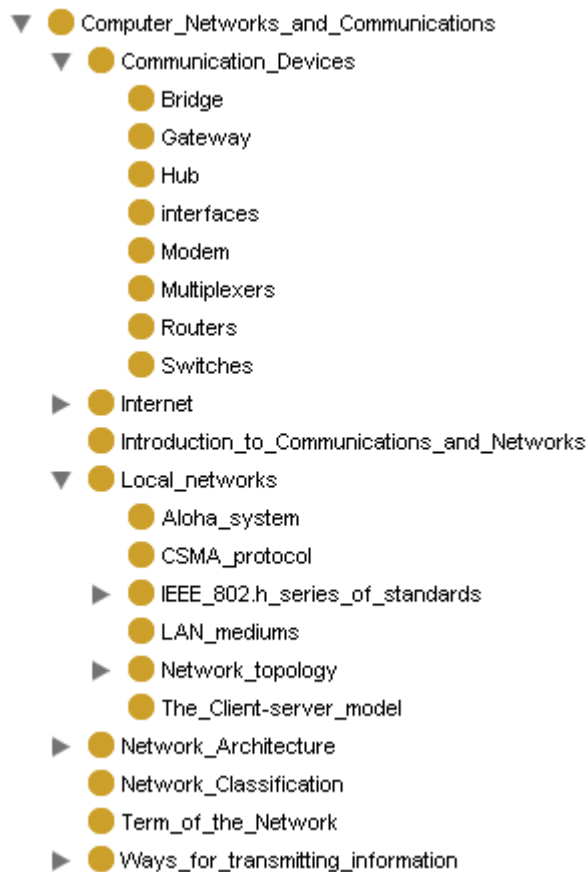
Primer mapiranja predmeta 'Računarske mreže i komunikacije' u ontološki model opisan je u nastavku. Tabela 39 prikazuje ishode i sadržaje predmeta.

<p>Ishod predmeta</p> <p>Ishodi omogućavaju da studenti:</p> <ul style="list-style-type: none">- poznaju principe organizacije i projektovanja računarske mreže, rad mrežnih resursa (hardvera i osnovnih mrežnih protokola),- umeju da podese osnovne mrežne postavke pod različitim operativnim sistemima, da instaliraju i pokrenu različite Internet servise, da analiziraju različite parametre mreže, da dijagnostifikuju i otklone kvarove u mreži.
<p>Sadržaj predmeta</p> <p><i>Teorijska nastava</i></p> <p>Uvod u komunikacije i mreže (značaj i primene računarskih komunikacija). Pojam mreže. Značaj komunikacija. Klasifikacija mreža. 2) Načini i sredstva za prenos informacija (vrste signala, sredstva za prenos signala, kodovi za prenos podataka, zaštita podataka u prenosu, otkrivanje i ispravljanje grešaka). 3) Komunikacioni uređaji (modemi, multiplekseri, koncentratori, mostovi, preklopnici, ruteri, vratnice, interfejsi). 4) Arhitektura mreža. Protokoli. Hijerarhija protokola. Troslojni model. OSI referentni model. Standardizacija u oblastima IT i telekomunikacija. Standardizacija mreža. Protokol HDLC. Mreže X.25. 5) Lokalne mreže. Medijumi koji se koriste u LAN-ovima. Topologije mreža (prsten, magistrala, stablo, zvezda). Aloha sistem. CSMA protokol. IEEE 802.h serija standarda (Ethernet, magistrala sa žetonom, prsten sa žetonom). Klijent-server model. 6) Internet. Internet usluge (E- pošta, FTP, diskusione grupe, WWW...). P2P mreže. Povezivanje na Internet. Mobilna telefonija. TCP/IP referentni model).</p> <p><i>Praktična nastava</i></p> <p>Vežbe, drugi oblici nastave, studijski istraživački rad. Auditorne (kompletan auditorijum) i vežbe u umreženoj računarskoj laboratoriji sa Internet vezom</p>

Tabela 39. Ishodi i sadržaji predmeta 'Računarske mreže i komunikacije' (Fakultet tehničkih nauka, 2013)

Predmet 'Računarske mreže i komunikacije' je u ontološkom modelu reprezentovan klasom *Computer_Networks_and_Communications* koja predstavlja potklasu klase *Informatics_domain_knowledge* (slika 15). Teme prikazane u sadržaju predmeta (tabela 39) su mapirane kao potklase klase *Computer_Networks_and_Communications* (slika 16).

Na primer, tema 'Uvod u komunikacije i mreže' je reprezentovana klasom *Introduction_to_Communications_and_Networks*, dok je sadržaj opisan u zagradi – ‘značaj i primene računarskih komunikacija’, preveden na engleski jezik i mapiran u labelu klase. Teme koje sadrže nabranjanja, poput ‘Komunikacioni uređaji’, ‘Topologije mreža’, ‘Internet’, su mapirane na odgovarajuće klase (*Communication_Devices*, *Network_topology*, *Internet*), dok su pripadajuće nabrojane teme reprezentovane kao potklase ovih klasa (slika 16).



Slika 16. Potklase klase *Computer_Networks_and_Communications*

Da bi se obezbedilo jednoznačno mapiranje u neku od *Skills* potklasa ishodi predmeta prikazani u tabeli 39 su, u skladu sa glagolima koji opisuju kategorije revidirane Blumove taksonomije, segmentirani na sledeće ishode:

- *poznaju* principe organizacije i projektovanja računarske mreže,
- *poznaju* rad mrežnih resursa (hardvera i osnovnih mrežnih protokola),
- umeju da *podese* osnovne mrežne postavke pod različitim operativnim sistemima,
- da *instaliraju* i *pokrenu* različite Internet servise,
- da *analiziraju* različite parametre mreže,
- da *dijagnostifikuju* i *otklone* kvarove u mreži.

Ovako particionisani ishodi su redom mapirani u sledeće klase (potklase klase *Skills*):

- *Recognize_the_principles_of_computer_network_organization_and_design* (potklasa klase *Remember-understand*),
- *Recognize_network_resources_work* (potklasa klase *Remember-understand*) (budući da postoji dodatan opis u zagradi, klasa sadrži labelu *Recognize hardware and basic network protocols*),
- *Adjust_basic_network_settings_under_different_operating_systems* (potklasa klase *Apply*),
- *Install_and_launch_a_variety_of_Internet_services* (potklasa klase *Apply*),
- *Analyze_different_network_parameters* (potklasa klase *Analyze*),
- *Diagnose_and_fix_faults_in_the_network* (potklasa klase *Apply*).

Ontološki model nastavničkog kurikuluma tehničkog fakulteta u Čačku nalazi se na adresi www.pef.uns.ac.rs/SerbianInformaticsTeacherEducationCurriculum/index.html.

Poglavlje V

5. Platforma za sinhronizaciju informatičkih kurikuluma

5.1. Pregled literature u oblasti usaglašavanja ontologija

5.1.1. Ontološko uparivanje

Uparivanje ontologija (engl. *Ontology matching*) podrazumeva proces pronalazjenja veza između entiteta različitih ontologija, odnosno pronalazjenje korespondencija između entiteta. Usaglašavanje ontologija (engl. *Ontology alignment*) nastaje kao rezultat različitih pristupa ontološkog uparivanja i predstavlja skup „korespondencija“ (Euzenat & Shvaiko, 2007) između ontologija. Korespondencija predstavlja relaciju, utvrđenu na osnovu pojedinog algoritma za uparivanje, između entiteta ontologija. U (Euzenat & Shvaiko, 2007) se naglašava važnost razlikovanja značenja pojmova korespondencija i mapiranje. Mapiranje označava „orijentisanu ili direktnu verziju usaglašavanja“ kojom se mapira entitet jedne ontologije na najviše jedan entitet druge ontologije, što je u skladu sa matematičkom definicijom mapiranja. U kontekstu ontološkog uparivanja, matematička definicija mapiranja bi, dakle, zahtevala da je mapiran objekat identičan originalu, odnosno da među njima postoji (samo) relacija ekvivalencije (Euzenat & Shvaiko, 2007). Međutim, kardinalnost veza među entitetima nastalim kao posledica primene ontološkog uparivanja, u opštem slučaju, nije ograničena na 1:1.

Ovakvo tumačenje usvojeno je i u ovoj disertaciji, po kom mapiranje predstavlja pravilo kojim se povezuje elemenat x jednog skupa X sa elementom y drugog skupa Y . U disertaciji je opisani vid tumačenja pojma mapiranja posebno uočljiv pri kreiranju ontoloških modela kurikuluma.

Ontološko uparivanje predstavlja važnu operaciju u tradicionalnim aplikacijama različite namene kao što su razvoj ontologija, integracija ontologija, integracija podataka ili skladištenje podataka (Shvaiko & Euzenat, 2011). Karakteristika ovih aplikacija je heterogenost modela baza podataka ili ontologija, pri čemu se uparivanje vrši manuelno ili poluautomatski „u fazi dizajna“. Uparivanja u „tradicionalnim“ aplikacijama obično predstavljaju preduslov za pokretanje sistema (Shvaiko & Euzenat, 2011). Na primer, u (Isaac et al., 2009) primenom ontološkog uparivanja omogućeno je pretraživanje, reindexiranje koncepata i generisanje zajedničkog leksikona sinonima između dve kolekcije knjiga Nacionalne biblioteke Holandije. Ove kolekcije su anotirane korišćenjem sopstvenih vokabulara i reprezentovane u SKOS (Simple Knowledge Organization System). Zablith (2009) predlaže sistem koji sadrži sve faze evolucije ontologije, pri čemu se u jednoj od faza primenjuju tehnike ontološkog uparivanja sa WordNet bazom i onlajn ontologijama.

Takođe, u (Shvaiko & Euzenat, 2011) se navodi da se ontološko uparivanje sve više koristi u aktuelnim aplikacijama za: „peer-to-peer“ (P2P) deljenje informacija, slaganje web servisa, pretraživanje i odgovaranje na upite. U ovom tipu aplikacija se, za razliku od tradicionalnih, ontološko uparivanje realizuje u fazi izvršavanja programa. U (Xiao & Cruz, 2006) se opisuje ontološki pristup za razrešenje interoperabilnosti podataka, sa naglaskom na problem procesuiranja upita u P2P heterogenom okruženju. Za reprezentovanje semantičkog mapiranja između metapodataka (ontologija) predložen je jezik za mapiranje – P2P Mapping Language (PML). U (Lopez et al., 2012) je prikazan “sistem za odgovaranje na pitanja” (engl. *Question Answering System*) zasnovan na ontologijama. Opisan sistem je sposoban da odgovori na upite lociranjem i integrisanjem informacija, koje su distribuirane u heterogenim semantičkim resursima.

5.1.2. Tehnike uparivanja ontologija

U ovoj sekciji prikazane su bazične metode za utvrđivanje sličnosti između entiteta ontologija, klasifikovane kao u (Euzenat & Shvaiko, 2007). Termin „bazične metode“ (Euzenat & Shvaiko, 2007) se odnosi na metode kojima se utvrđuje sličnost na osnovu specifičnog svojstva entiteta (poput naziva, pripadnosti određenoj strukturi, relacija između entiteta itd.).

5.1.2.1. Tehnika zasnovana na sličnosti naziva (*Terminološka sličnost*)

Pri poređenju naziva entiteta ontologija moguće je razmatrati stringove sadržane u imenima entiteta, njihovim labelama ili komentarima. Tehnike poređenja naziva se, po (Euzenat & Shvaiko, 2007), mogu podeliti na dva osnovna pristupa u zavisnosti od toga da li se uzimaju u obzir i značenja stringova, odnosno lingvistička znanja. To su: metode zasnovane na sličnosti stringova i metode zasnovane na jeziku.

5.1.2.1.1. Metode zasnovane na sličnosti stringova

Metode zasnovane samo na sličnosti stringova tumače string kao niz karaktera. Ove metode, dakle, ne podrazumevaju interpretaciju značenja stringova u prirodnom jeziku. One su, generalno, definisane pravilom za koje važi da što su stringovi sličniji to je veća verovatnoća da se oni odnose na isti koncept. Na primer, za sličnost klasa *Programiranje* i *Program*, primenom string baziranih tehnika, dobila bi se veća vrednost sličnosti nego za klase *Evaluacija* i *Ocenjivanje*. Pre poređenja stringova koji imaju značenje u „prirodnom jeziku“ u sistemima za ontološko usaglašavanje se često primenjuje neka od tehnika normalizacija stringova:

- uklanjanje praznog mesta,
- uklanjanje znakova interpunkcije,

- uklanjanje „stop reči“ (primeri „stop reči“ u engleskom jeziku su *the, is, at, which, on*),
- uklanjanje cifara,
- zamena velikih slova malim slovima i slično.

U nastavku su prikazane string-bazirane metode korišćene u aktuelnim sistemima za uparivanje ontologija.

Edit rastojanje odgovara minimalnim „troškovima“ operacija potrebnih da bi se od jednog stringa dobio drugi. Ovo rastojanje se često primenjuje u savremenim sistemima ontološkog uparivanja (Cruz et al., 2009; Lambrix & Tan, 2006; Li et al., 2009) za utvrđivanje sličnosti između stringova koji mogu imati pravopisne greške.

Levenshtein rastojanje (Levenshtein, 1965) odgovara minimalnom broju umetanja, brisanja ili zamene pojedinačnih karaktera potrebnom za transformisanje jednog stringa u drugi. Ovo rastojanje, dakle, odgovara *Edit* rastojanju pri čemu su svi „troškovi“ jednaki 1 (Euzenat & Shvaiko, 2007). Levenshtein rastojanje predstavlja često korišćenu metodu u sistemima za ontološko uparivanje (Cruz et al., 2009; Jean-Mary et al., 2009; Huber et al., 2011; Ngo & Bellahsene, 2012).

Podstring (engl. **Substring**) **sličnost**, korišćena u (Cruz et al., 2009), se definiše kao:

$$s_{sub}(s_1, s_2) = \frac{2|S_{max}|}{|s_1| + |s_2|}, \quad (5.1.1)$$

S_{max} je najveći zajednički podstring

***n*-gram sličnost**, primenjena u (Lambrix & Tan, 2006), predstavlja broj zajedničkih *n*-gram-a između stringova, pri čemu *n*-gram predstavlja sekvencu od *n* karaktera.

Jaro mera (Jaro, 1976, 1989) izračunava broj i „blizinu“ zajedničkih karaktera na sledeći način (Euzenat & Shvaiko, 2007).

$$s_{jaro}(s_1, s_2) = \frac{1}{3} \left(\frac{|c(s_1, s_2)|}{|s_1|} + \frac{|c(s_2, s_1)|}{|s_2|} + \frac{|c(s_1, s_2)| - |t(s_1, s_2)|}{|c(s_1, s_2)|} \right), \quad (5.1.2)$$

$s[i] \in c(s_1, s_2)$ ako i samo ako

$$\exists j \in \left[i - \frac{(\min(|s_1|, |s_2|))}{2}, i + \frac{(\min(|s_1|, |s_2|))}{2} \right],$$

$c(s_1, s_2)$ su zajednički (upareni) karakteri stringova s_1 i s_2 , $t(s_1, s_2)$ su elementi $c(s_1, s_2)$ koji se pojavljuju u drugačijem redosledu u s_1 i s_2 .

U (Winkler, 1999) metoda je unapređena tako da favorizuje uparivanje stringova sa dužim zajedničkim prefiksom.

Jaro-Winkler mera se definiše na sledeći način (Euzenat & Shvaiko, 2007).

$$s_{winkler}(s_1, s_2) = s_{jaro}(s_1, s_2) + PQ \frac{(1 - s_{jaro}(s_1, s_2))}{10} \quad (5.1.3)$$

P je dužina zajedničkog prefiksa, dok je Q konstanta.

U (Cruz et al., 2009) se korisnicima nudi opciono izračunavanje sličnosti Jaro-Winkler metodom, dok se u (Huber et al., 2011) ova metoda kombinuje sa drugim string-baziranim metodama.

Token-bazirana rastojanja se primenjuju u slučajevima kada se string posmatra kao dugačak tekst (koji se sastoji od mnogo reči). U (Jean-Mary et al., 2009) se izračunava sličnost između komentara poređenih entiteta kao varijacija Levenshtein rastojanja primenjena na tokene. Ova tehnika se primenjuje pri poređenju komentara, jer komentari obično predstavljaju „frazu ili rečenicu u prirodnom jeziku“ (Jean-Mary et al., 2009). Nakon preslikavanja komentara na uređen skup tokena izračunava se broj operacija tokena (umetanja, brisanja i zamena tokena) potrebnih da bi se transformisao jedan komentar u drugi. Ako su s_1 i s_2 komentari poređenih entiteta i $num(s_1, s_2)$ označava broj potrebnih token operacija, tada se sličnost, definisana u (Jean-Mary et al., 2009), predstavlja na sledeći način.

$$s_{token}(s_1, s_2) = 1 - \frac{num(s_1, s_2)}{\max(|tok(s_1)|, |tok(s_2)|)} \quad (5.1.4)$$

TFIDF (Frekvencija termina – inverzna frekvencija dokumenta) metrika (Robertson & Jones, 1976) se koristi za ocenjivanje relevantnosti dokumenta, odnosno skupa reči, za termin uzimajući u obzir frekventnost pojavljivanja termina u korpusu (Euzenat & Shvaiko, 2007). Neki oblik TFIDF mere primenjen je u sistemima poput (Cruz et al., 2009; Hu et al., 2008; Li et al., 2009). U (Li et al., 2009) se najpre za svaki entitet e (koncept ili svojstvo) formira dokument $D(e)$ koji se sastoji od reči sadržanih u metapodacima entiteta, svojstvima entiteta, potkonceptima (povezanim konceptima, u slučaju da entitet predstavlja svojstvo) i instancama entiteta. Nakon toga generiše se težinski vektor primenom formule $tf \cdot idf$, pri čemu je tf_i frekvencija pojavljivanja reči w_i u dokumentu $D(e)$, dok je idf inverzna frekvencija dokumenata koji sadrže w_i . Na taj način je svaki entitet prve ontologije i svaki entitet druge ontologije konvertovan u odgovarajući vektor $V(e_1)$ i $V(e_2)$ respektivno. Sličnost između entiteta e_1 i e_2 izračunava se kao kosinus dva vektora.

5.1.2.1.2. Metode zasnovane na jeziku

String-bazirane metode su pogodne za poređenje ontoloških modela u kojima su korišćeni slični stringovi za definisanje ekvivalentnih koncepata (Euzenat & Shvaiko, 2007). Dakle, one mogu dati visoku sličnost za stringove koji sadrže veliki broj ekvivalentnih karaktera. Ipak, procentualno mali broj karaktera razlike može uticati na suštinski (u prirodnom jeziku) različito ili čak suprotno značenje analiziranih stringova. Takođe, ukoliko se za iste koncepte u poređenim ontologijama koriste sinonimi opisani različitim nizom karaktera, string-bazirane metode neće dati visoku sličnost. Stoga, sistemi za usaglašavanje ontologija (Cruz et al., 2009; Hu et al., 2008; Huber et al., 2011; Jean-Mary et al., 2009; Jiménez-Ruiz & Cuenca Grau, 2011; Lambrix & Tan, 2006; Li et al., 2009) danas

podrazumevaju i metode zasnovane na jeziku i koriste neki od elektronski dostupnih leksikona sinonima. Metode zasnovane na jeziku se oslanjaju na tehnike obrade prirodnog jezika (engl. *Natural Language Processing* - NLP) i mogu se podeliti na „unutrašnje“ i „spoljašnje“ metode.

Prvi tip metoda odnosi se na lingvističku normalizaciju koja podrazumeva (Euzenat & Shvaiko, 2007; Lin & Sandkuhl, 2008):

- pronalaženje reči koja predstavlja osnovu, primenom morfoloških metoda (na primer, string „*računarski*“ se konvertuje u string „*računar*“),
- tokenizaciju - podelu stringa u niz tokena primenom metoda normalizacije stringova kojima se identifikuju prazna mesta, cifre, velika slova, znakovi interpunkcije i slično (na primer string „*programiranje grafičkog korisničkog interfejsa*“ postaje niz stringova {*“programiranje”, “grafickog”, “korisnickog”, “interfejsa”* }),
- uklanjanje stop reči (na primer „*programiranje u jeziku visokog nivoa*“ postaje „*programiranje jeziku visokog nivoa*“),
- ekstrakciju termina na osnovu ponavljanja morfološki sličnih fraza u tekstu i primene šablona (za ovu tehniku potrebna je značajna „količina“ teksta).

Primenom unutrašnjih metoda dobija se, dakle, niz tokena (termina) na koje je moguće primeniti string-bazirane metode.

Spoljašnje metode koriste rečnike i leksikone sinonima čime se omogućava uparivanje reči istog značenja. Analizom aktuelne literature o sistemima za usaglašavanje ontologija može se uočiti da većina sistema koristi WordNet leksičku bazu podataka (Cruz et al., 2009; Jean-Mary et al., 2009; Jiménez-Ruiz & Cuenca Grau, 2011; Lambrix & Tan, 2006; Li et al., 2009; Ngo & Bellahsene, 2012). Metode za izračunavanje sličnosti reči WordNet baze podataka mogu se klasifikovati u sledeće kategorije (Lin & Sandkuhl, 2008):

- *Metode zasnovane na putanji (ivici)*: semantička sličnost reči određena je „rastojanjem“ između reči i pozicijom reči u taksonomiji; što je kraća putanja između čvorova kojima pripadaju to su reči sličnije.
- *Statističke metode zasnovane na informacionom sadržaju* porede sadržaj čvorova da bi utvrdili sličnost reči; što više zajedničkih informacija imaju dva koncepta, to su oni sličniji. Primenom ovog pristupa se izbegava vremenski zahtevan proces pronalaženja uniformne putanje između dva čvora obuhvaćen prethodnom metodom.
- *Hibridne metode* predstavljaju kombinaciju navedenih metoda. One, na primer, mogu razmatrati poziciju čvorova u taksonomijskom grafu ali i njihov informacioni sadržaj.

Metode zasnovane na putanji

Leacock and Chodorow (1998) su predložili sledeću meru sličnosti između koncepta c_1 i c_2 .

$$s_{LC}(c_1, c_2) = -\log \frac{\text{length}(c_1, c_2)}{2D}, \quad (5.1.5)$$

gde $\text{length}(c_1, c_2)$ predstavlja minimalno rastojanje između koncepata, odnosno dužinu najkraćeg puta između koncepata računajući broj čvorova. D je maksimalna dubina taksonomije.

Wu and Palmer (1994) su definisali sličnost dva koncepta na osnovu njihovog rastojanja do “najnižeg” zajedničkog superkoncepta i rastojanja zajedničkog superkoncepta do korena taksonomije.

$$s_{WP}(c_1, c_2) = \frac{2N_3}{N_1 + N_2 + 2N_3}, \quad (5.1.6)$$

c_3 je najniži zajednički superkoncept koncepata c_1 i c_2 , N_1 je broj čvorova od c_1 do c_3 , N_2 je broj čvorova na putanji od c_2 do c_3 , dok je N_3 broj čvorova od c_3 do korenskog čvora.

Statističke metode zasnovane na informacionom sadržaju

Resnik (1995, 1999) predlaže metriku u kojoj je svaki synset c povezan sa verovatnoćom $p(c)$ pojavljivanja instance koncepta u korpusu. Obično, $p(c)$ predstavlja sumu pojavljivanja reči synset-a podeljenu sa ukupnim brojem koncepata. Resnik uzima u obzir informacioni sadržaj (odnosno entropiju) definisan kao:

$$IC(c) = -\log p(c), \quad (5.1.7)$$

tako da je sličnost opisana sledećom jednačinom

$$s_{resnik}(c_1, c_2) = \max_{c \in S(c_1, c_2)} [IC(c)], \quad (5.1.8)$$

$S(c_1, c_2)$ je skup zajedničkih superkonceptata (opštijih synset-a) koncepata c_1 i c_2 . Ova mera, dakle, razmatra samo zajednički superkoncept koji sadrži najveću količinu informacija. U (Resnik, 1999) se navodi da se u praksi češće utvrđuje sličnost reči nego sličnost koncepata. Ako $sen(w)$ odgovara skupu koncepata u taksonomiji koji predstavljaju smisao reči w , tada se sličnost reči definiše na sledeći način.

$$s_{resnik}(w_1, w_2) = \max_{c_1, c_2} [s_{resnik}(c_1, c_2)], \quad (5.1.9)$$

gde $c_1 \in sen(w_1)$ i $c_2 \in sen(w_2)$.

U (Lin, 1998) je proširena Resnik-ova metoda tako da uzima u obzir informacioni sadržaj koncepata c_1 i c_2 . Lin-ova sličnost definiše stepen verovatnoće preklapanja između dva synset-a.

$$s_{lin}(x_1, x_2) = \frac{2 \log p(c_0)}{\log p(c_1) + \log p(c_2)}, \quad (5.1.10)$$

gde $x_1 \in c_1$ i $x_2 \in c_2$, c_0 predstavlja „najspeficiji“ koncept koji uključuje i x_1 i x_2 .

Hibridne metode

Jiang and Conrath (1998) predlažu kombinovani model koji je izveden iz metode zasnovane na dužini putanje dodavanjem informacionog

sadržaja kao faktora odlučivanja. Snaga linka (engl. *Link strength* - LS) određene putanje predstavlja razliku vrednosti informacionog sadržaja potkoncepta c_i i njegovog roditeljskog koncepta p :

$$LS(c_i, p) = IC(c_i) - IC(p) \quad (5.1.11)$$

Jiang and Conrath (1998) razmatraju i druge faktore kao što su dubina čvora, lokalna “gustina”, tip linka. U slučaju da se uzima u obzir samo snaga linka, rastojanje se određuje na sledeći način.

$$dist_{JC}(w_1, w_2) = IC(c_1) + IC(c_2) - 2IC(c_3), \quad (5.1.12)$$

$c_1 = sen(w_1)$, $c_2 = sen(w_2)$, $sen(w)$ označava skup mogućih značenja reči w , c_3 je najniži superkoncept konceptata c_1 i c_2 .

5.1.2.2. Strukturne metode

U (Euzenat & Shvaiko, 2007) se strukturne metode klasifikuju na metode kojima se poredi interna struktura ontologija i metode kojima se poredi relaciona struktura ontologija.

5.1.2.2.1. Interna struktura

Metodama zasnovanim na internoj strukturi razmatra se skup svojstava, opseg svojstava, njihova kardinalnost, osobine tranzitivnosti ili simetrije (Euzenat & Shvaiko, 2007). Pri poređenju *datatype* svojstava može se razmatrati bliskost tipa podataka (opsega) (na primer tipovi podataka char i String su bliskiji nego tipovi podataka int i String) ili sličnost domena (skupa vrednosti). U (Jean-Mary et al., 2009) sličnost *objektnih* svojstava (relacija) se izračunava kao težinska suma sličnosti domena i sličnosti opsega poređenih ontologija.

Ipak, metode zasnovane na internoj strukturi se, po (Euzenat & Shvaiko, 2007), prvenstveno koriste pri uparivanju šema baza podataka, dok je za ontološka uparivanje mnogo značajnija relaciona struktura.

5.1.2.2.2. Relaciona struktura

Metode zasnovane na relacionoj strukturi najčešće razmatraju **taksonomijsku strukturu** ontologija, odnosno ontologija se posmatra kao graf sačinjen od relacija superklasa/potklasa. U literaturi koja se bavi sistemima za usaglašavanje ontologija opisane su brojne metode zasnovane na taksonomijskoj strukturi. Tako se u (Lambrix & Tan, 2006) strukturna sličnost određuje primenom iterativnog algoritma kojim se proverava da li se dva koncepta pojavljuju na sličnim pozicijama uzimajući u obzir *is-a* i *part-of* hijerarhijske relacije sa već uparenim konceptima. Ideja autora je da su koncepti, koji se nalaze na sličnim hijerarhijskim pozicijama u odnosu na uparene superkoncepte i potkoncepte, i sami slični. U (Jean-Mary et al., 2009) strukturna sličnost se određuje kao težinska suma vrednosti, dobijene na osnovu sličnosti svih roditelja, i vrednosti dobijene na osnovu sličnosti dece poređenih entiteta. Cruz et al. (2009) u ovoj fazi koriste dve metode

zasnovane na: sličnosti „potomaka“ (ukoliko su dva čvora uparena sa visokom sličnošću tada raste sličnost između njihovih „potomaka“ tako da veći uticaj na sličnost imaju direktni roditelji) i sličnosti „rođaka“ (uzimaju se u obzir relacije između koncepata istog nivoa – „rođaka“). Slično, u (Ngo & Bellahsene, 2012) se primenjuje jednostavna metoda po kojoj ako su dva koncepta poređenih ontologija slična, tada njihovi susedni koncepti mogu takođe biti slični. U (Euzenat & Shvaiko, 2007) se navodi da je strukturnu sličnost moguće utvrditi izračunavanjem broja čvorova (putanja) između klasa i primenom (lingvističkih) metoda zasnovanih na putanji poput (Leacock & Chodorow, 1998; Wu & Palmer, 1994).

Takođe, važan aspekt pri utvrđivanju strukturne sličnosti, pored taksonomijske strukture, predstavljaju *relacije između klasa*. U najopštijem slučaju utvrđivanje sličnosti između entiteta dve ontologije može biti zasnovano na relaciji sa drugim entitetima na sledeći način: što su poređeni entiteti sličniji to su sličniji entiteti sa kojima su oni povezani preko tih relacija (Euzenat & Shvaiko, 2007). Pri tome, opisan princip može biti primenjen i na skup relacija i na skup klasa.

Za razliku od taksonomijske strukture relacioni graf može biti cirkularan. U slučaju cirkularnih zavisnosti, „lokalno“ izračunavanje sličnosti nije više moguće, te je, po (Euzenat & Shvaiko, 2007), jedno od rešenja primena iterativnog algoritma poput „Similarity flooding“-a (Melnik et al., 2002). Metoda „Similarity flooding“-a zasnovana je na konverziji ulaznih šema u direktne označene grafove i propagaciji vrednosti sličnosti između susednih čvorova kroz više iteracija (Melnik et al., 2002). U (Li et al., 2009; Ngo & Bellahsene, 2012) koriste se varijacije ove metode.

5.1.2.3. Ekstenzione tehnike

Ekstenzione tehnike se odnose na poređenje instanci ontologija. U (Ngo et al., 2011) se primenjuju za:

- Utvrđivanje sličnih instanci iz poređenih ontologija,
- Otkrivanje novih parova koncepata/svojstava na osnovu liste uparenih instanci.

U prvom slučaju primenjuje se sledeći princip: ukoliko dve instance pripadaju uparenim klasama i imaju slične labele, tada se one razmatraju kao slične. Takođe, kao kandidati za uparivanje uzimaju se u obzir i instance koje imaju sličan tekstualni opis. Analogno, u drugom slučaju, za svaki par klasa iz dve ontologije važi da ako je većina njihovih instanci uparena, tada su i klase uparene.

5.1.2.4. Semantičke metode

Semantičke metode predstavljaju deduktivne metode koje se primenjuju na skup uparenih entiteta i ponašaju se kao „pojačavač“ dobijenih usaglašenosti (Euzenat & Shvaiko, 2007). One se mogu koristiti nakon metoda opisanih u prethodnim sekcijama kao i nakon metoda uparivanja ontologija uz pomoć eksternih resursa (posrednih

ontologija). „Posredne“ ontologije služe za definisanje zajedničkog konteksta ili „pozadinskog znanja“ poređenih ontologija. One obezbeđuju sveobuhvatan model domena od interesa ontologija koje su predmet uparivanja, čime je obezbeđeno da od više mogućih parova sličnih termina budu odabrani oni koji odgovaraju konkretnom kontekstu (Euzenat & Shvaiko, 2007). Tako se u (Lambrix & Tan, 2006) i (Groß et al., 2012) koriste eksterni resursi i ontologije u cilju raspolaganja većom količinom dostupnog znanja iz konkretnih domena.

Jean-Mary et al. (2009) su definisali više obrazaca koji se koriste u procesu semantičke verifikacije poput „korespondencije višestrukih entiteta“, „nekompletnost domena i opsega“, „nekompletnost veza potklasa i ekvivalencije“ itd. „Nekompletnost veza potklasa i ekvivalencije“ podrazumeva sledeće pravilo: ako za dobijena dva para klasa poređenih ontologija, klase jedne ontologije, sadržane u dobijenim parovima, su u vezi superklasa/potklasa, dok klase druge ontologije, koje pripadaju posmatranim parovima, nisu u superklasa/potklasa relaciji, tada usaglašavanje ne može biti verifikovano. U (Ngo & Bellahsene, 2012) se, na primer, primenjuje sledeći princip: ukoliko su dva objektna svojstva slična, ali nijedna klasa njihovog domena nije uparena i ako klase opsega, takođe, nisu uparene, tada se dobijeno uparivanje ovih svojstava tumači kao nekonzistentno i biće uklonjeno.

5.1.3. Strategije za kombinovanje različitih tehnika

Postoji dva generalna pristupa za kombinovanje različitih metoda uparivanja, koja se, po (Faria et al., 2013), koriste u većini sistema za usaglašavanje ontologija. To su:

- Sekvencijalna kompozicija, gde rezultati jednog uparivača (engl. *matcher*) predstavljaju ulaz za sledeći,
- Paralelna kompozicija, u kojoj se različiti uparivači pokreću nezavisno, i dobijeni rezultati se kombinuju korišćenjem specifičnog kriterijuma, pri čemu je moguća
 - Homogena kompozicija, u kojoj se isti ulaz koristi za sve (konkurentne) uparivače i
 - Heterogena kompozicija, u kojoj različiti tipovi podataka predstavljaju ulaz odgovarajućih uparivača (Euzenat & Shvaiko, 2007).

Paralelna kompozicija podrazumeva sumiranje rezultata više uparivača primenom težinske sume, proseka dobijenih vrednosti, određivanjem maksimuma i sl. Tako su u (Cruz et al., 2009) primenjeni posebni algoritmi za svako leksičko svojstvo entiteta (lokalno ime, labela i komentar). Ukupna leksička sličnost predstavlja linearnu težinsku sumu rezultata korišćenih algoritama za izračunavanje sličnosti labela, komentara i lokalnih naziva. Vrednost težinskih faktora izabrana je eksperimentalno. U istom izvoru se i ukupna sličnost svih uparivača (leksičkog, strukturnog i ekstenzionog) izračunava primenom težinskog proseka. Slično, u (Lambrix & Tan,

2006) rezultati terminoloških uparivača kombinuju se primenom težinske sume sa unapred definisanim težinskim faktorima dok se na izlazu sistema, pri izračunavanju ukupne sličnosti dobijene nakon primene svih algoritama, primenjuju korisnički definisani težinski faktori. U (Groß et al., 2012) kombinovanje rezultata svih uparivača se realizuje primenom unije na dobijene rezultate, a zatim izračunavanjem proseka. I u (Faria et al., 2013) spajaju se rezultati više uparivača, s tim što se u sledećoj fazi primenjuje kriterijum maksimalne vrednosti u slučaju da se parovi ponavljaju.

Metode za izračunavanje sličnosti mogu se kombinovati primenom principa mašinskog učenja kao u (Ngo & Bellahsene, 2012) gde je to realizovano u fazi izračunavanja terminološke sličnosti. Takođe, u (Huber et al., 2011) omogućeno je određivanje težinskih faktora korišćenjem algoritama mašinskog učenja pri izračunavanju težinske sume izabranih string baziranih metrika.

Težinske faktore je moguće utvrditi i dinamički, nakon dobijenih rezultata primenjenih uparivača, kao u (Ngo et al., 2011). U (Li et al., 2009) je opisan princip po kome se u fazi pretprocesiranja izračunavaju faktori sličnosti labela i struktura poređenih ontologija i u zavisnosti od dobijenih vrednosti se, dinamički, selektuju i kombinuju odgovarajuće metode uparivanja.

5.1.4. Učešće korisnika u usaglašavanju ontologija

Učešće korisnika u sistemima za usaglašavanje ontologija se može klasifikovati na tri generalna aspekta (Euzenat & Shvaiko, 2007):

- Obezbeđivanje povratne informacije o rezultatima usaglašavanja radi prilagođavanja rezultata stavovima korisnika,
- Dinamički izbor i kombinovanje uparivača,
- Obezbeđivanje inicijalnih uparivanja (i parametara).

U tradicionalnim aplikacijama (Shvaiko & Euzenat, 2011) učešće korisnika, po pravilu, podrazumeva prihvatanje ili odbijanje rezultata uparivanja dobijenih na izlazu sistema za usaglašavanje ontologija. Ipak, u istom izvoru se navodi da veliki broj podataka u poređenim ontologijama može učiniti ovaj zadatak teškim. U (Ngo & Bellahsene, 2012) korisnik može modifikovati, uklanjati netačne parove, kao i dodati nove parove pomoću grafičkog interfejsa. Slično, u (Lambrix & Tan, 2006) korisniku je omogućeno da odluči da li su dobijeni koncepti ekvivalentni, da li su u *is-a* relaciji ili je potrebno odbaciti dobijeni par kao netačan. Ukoliko korisnik odbaci par kao netačan, „dužan“ je da bar jedan koncept para izmeni/preimenuje.

Takođe, tokom procesa usaglašavanja u sistemu opisanom u (Lambrix & Tan, 2006) korisnik ima mogućnost izbora kombinacije uparivača. Ova opcija postoji i u sistemu predstavljenom u (Cruz et al. 2009) koji pruža sveobuhvatan interfejs pomoću kog je moguće vizuelizovati ontologije i menjati niz parametara usaglašavanja ontologija poput izbora uparivača, načina kombinovanja metoda,

izmene vrednosti praga, generisanja rečnika pojmova, prilagođavanja rezultata uparivanja i sl.

U (Ngo & Bellahsene, 2012) učešće korisnika podrazumeva i izbor metrike sličnosti, obezbeđivanje podataka inicijalnog usaglašavanja i izbor modela mašinskog učenja.

5.1.5. Evaluacija rezultata uparivanja

Da bi se evaluirali algoritmi za uparivanje ontologija potrebno je dobijene rezultate uporediti sa referentnim rezultatima ili testnim ontologijama. Mere preciznosti i odziva, koje potiču iz sistema za pronalaženje informacija, predstavljaju “najistaknutije kriterijume upoređivanja” (Euzenat, 2007). Ove mere su zasnovane na poređenju očekivanih i dobijenih rezultata analiziranog sistema. U kontekstu uparivanja ontologija, usaglašavanje dobijeno od sistema koje je predmet evaluacije (A) se upoređuje sa referentnim usaglašavanjem (R) (Euzenat, 2007).

Mera preciznosti P (engl. *Precision*) predstavlja odnos broja ispravno pronađenih korespondencija i ukupnog broja dobijenih korespondencija:

$$P(A, R) = \frac{|R \cap A|}{|A|} \quad (5.1.13)$$

Mera odziva R (engl. *Recall*) predstavlja odnos broja ispravno pronađenih korespondencija i ukupnog broja očekivanih korespondencija.

$$R(A, R) = \frac{|R \cap A|}{|R|} \quad (5.1.14)$$

U (Euzenat & Shvaiko, 2007) se navodi da je ponekad poželjna samo jedna vrednost kao rezultat poređenja sistema. Međutim, sistemi često nisu „uporedivi“ primenom samo mera preciznosti ili samo mera odziva. Na primer, sistem koji ima visok odziv može imati nisku preciznost i obrnuto. Stoga se pri evaluaciji sistema za usaglašavanje ontologija (Grau et al., 2013) najčešće primenjuje i F-mera. F mera kombinuje preciznost i odziv i definiše se na sledeći način:

$$F(P, R) = \frac{P(A, R)R(A, R)}{(1 - \alpha)P(A, R) + \alpha R(A, R)} \quad (5.1.15)$$

Za $\alpha = 0$, F-mera je jedna odzivu, dok za $\alpha = 1$ F-mera odgovara preciznosti. Veoma često (Euzenat & Shvaiko, 2007) je $\alpha = 0.5$ te je

$$F_{0.5}(P, R) = \frac{2P(A, R)R(A, R)}{P(A, R) + R(A, R)} \quad (5.1.16)$$

5.2. Opis izabranih tehnika za usaglašavanje ontoloških reprezentacija kurikuluma

Jezici za reprezentaciju ontologija najčešće opisuju sledeće tipove entiteta: *klase* (C) ili koncepte koji predstavljaju osnovne entitete

ontologije; *Individulas* (I) ili objekte koji predstavljaju instance definisanih klasa; tipove podataka (T); literale (L) koji označavaju konkretne vrednosti podataka; *relacije* ili svojstva (P) koje omogućavaju povezivanje instanci međusobno (objektna svojstva, *owl:ObjectProperty*) kao i povezivanje instanci i vrednosti podataka (*datatype* svojstva, *owl:DatatypeProperty*).

U sistemu predstavljenom u ovoj disertaciji se porede samo klase ontologija, budući da su objektna i *datatype* svojstva unapred predefinisana i ista u obe ontologije.

Tako, na primer, neka su date ontologije O_1 i O_2 i neka je C_{ik} klasa ontologije. U kontekstu sistema opisanog u ovom radu usaglašavanje se tada može formalno napisati na sledeći način.

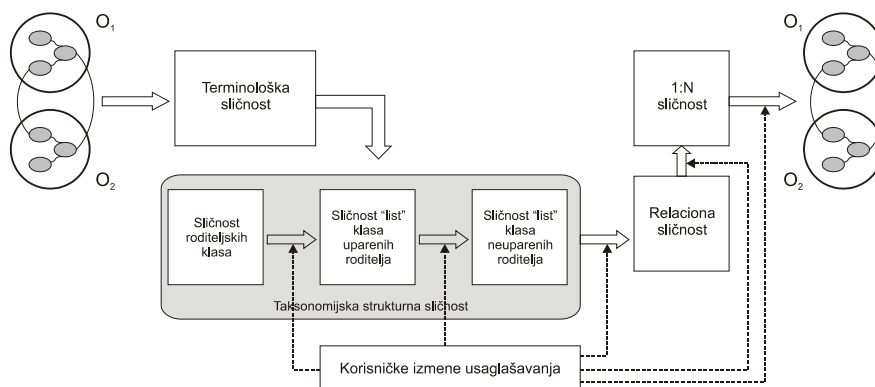
$$\text{Alignment}(O_1, O_2) = \left\{ (C_{i1}, C_{j2}, \text{conf}_i, \text{relation}_i) \mid C_{i1} \in O_1, C_{j2} \in O_2, \right. \\ \left. \text{conf}_i \in [0,1], \text{relation}_i \in \{=, \subseteq, \supseteq\} \right\} \quad (5.2.1)$$

Dakle, usaglašenost dve ontologije čine parovi klasa koje su međusobno povezane relacijama ekvivalencija (=), nadskup/nadklasa (\supseteq) i podskup/potklasa (\subseteq) sa stepenom podudarnosti conf_i koji uzima vrednost iz intervala $[0,1]$, odnosno u procentima od 0% do 100%. Sistem koji je predložen u ovom radu podržava veze jedan prema više, što znači da kao konačan rezultat usaglašavanja jedna klasa jedne ontologije može biti superklasa više klasa druge ontologije. Na slici 17 je prikazana šema sistema za uparivanje kurikuluma (srednjoškolskog i nastavničkog modela ili nastavničkih modela međusobno).

Uparivanje se vrši u dve faze.

U **prvoj fazi**, koja se može smatrati preprocesiranjem, određuje se terminološka sličnost primenom lingvistički zasnovane metode i String bazirane metode. Ovde se uzimaju u obzir lokalna imena i labele klasa.

U **drugoj fazi** se sekvencijalno primenjuju koraci kojima se, redom, utvrđuju: taksonomijska strukturalna sličnost, relacionalna sličnost i sličnosti „jedan prema više“. Pri tome je omogućena manuelna intervencija korisnika nad rezultatima dobijenim u svakom koraku druge faze.



Slika 17. Postupak uparivanja kurikuluma

Tri uparivača koji izračunavaju taksonomijsku strukturnu sličnost porede samo potklase klase *Knowledge*. Ukoliko se porede srednjoškolski i nastavnički kurikulum, razmatraju se samo klase u koje su mapirane teme domenskih (informatičkih) znanja. S druge strane, u slučaju poređenja nastavničkih modela posmatraju se sve *Knowledge* potklase ontologija, pri čemu se međusobno porede samo potklase istih klasnih struktura (potklase klase *General knowledge* jedne ontologije sa potklasama klase *General knowledge* druge ontologije; potklase klase *Informatics_domain_knowledge* jedne ontologije sa potklasama klase *Informatics_domain_knowledge* druge ontologije itd.) Provera struktuiranosti uparenih veština (potklase klase *Skills*) vrši se u kasnijoj fazi kada se prvenstveno upoređuju nivoi revidirane Blumove taksonomije kojima pripadaju uparene veštine (na primer, u slučaju poređenja srednjoškolskog i nastavničkog modela proverava se da li su srednjoškolske veštine na nižem ili istom nivou revidirane Blumove taksonomije sa uparenim nastavničkim veštinama). Dakle, u fazi izračunavanja taksonomijske strukturne sličnosti podatak da je neka veština potklasa odgovarajuće veštine/klase koja predstavlja revidiranu Blumovu taksonomiju (*Remember-understand, Apply, Evaluate, Analyze, Create*) ne utiče na rezultat uparivanja. Ukoliko ipak postoji struktuiranost u potklasama klasa revidirane Blumove taksonomije ona se uzima u obzir pri izračunavanju relacije sličnosti (sekcija 5.2.4.).

Rezultat primene svakog uparivača je matrica sličnosti koja sadrži sličnost svih klasa iz jedne ontologije sa svim klasama iz druge ontologije. Cilj primene uparivača je da se za svaku klasu iz jedne ontologije pronađe najbližija klasa/klase iz druge ontologije, pri čemu u slučaju kardinalnosti veza 1:1, jednoj klasi iz jedne ontologije odgovara najviše jedna klasa iz druge ontologije. U sistemu opisanom u ovom radu za utvrđivanje najboljih uparenih parova klasa koristi se *Greedy selection* algoritam opisan u (Wu et al., 2004), uz uslov da su dobijene vrednosti veće od zadatog praga.

5.2.1. Određivanje “uparenih entiteta” iz matrice sličnosti

Problem uparivanja je po (Wu et al., 2004) dobro izučavan u literaturi koja se bavi teorijom grafova, po kojoj je moguće primeniti nekoliko kriterijuma za utvrđivanje najboljih parova: maksimalna kardinalnost, maksimalna totalna “težina” i “stabilan brak”. Uparivanje ima maksimalnu kardinalnost ukoliko ima najveći broj mapiranja (uparenih polja); uparivanje ima maksimalnu totalnu težinu ako je suma težina njegovih mapiranja najveća; “stabilan brak” zahteva da ne postoje takve dve kombinacije uparenih polja (x, y) i (x_1, y_1) tako da x više “preferira” y_1 nego y i da y_1 “preferira” x više nego x_1 . Po (Wu et al., 2004), *Greedy* izbor se, za uparivanje kardinalnosti 1:1, može smatrati “monogamnom” verzijom “*perfectionist egalitarian polygamy*” metrike selekcije koja, po empirijskim rezultatima prikazanim u (Melnik et al., 2002), daje najbolje rezultate u uparivanju šema.

Postupak predstavljen u (Wu et al., 2004), primenjen na dvodimenzionalnu matricu sličnosti, može se opisati na sledeći način (Euzenat & Shvaiko, 2007; Jean-Mary et al., 2009).

- Izbor para entiteta $e_{m1} \in O_1$ i $e_{n2} \in O_2$ koji ima najveću vrednost sličnosti od svih parova entiteta.
- “Uklanjanje” vrsta i kolona u kojima se nalaze e_{m1} i e_{n2} tako da e_{m1} ne može biti uparen ni sa jednim $e_{j2} \in O_2$, $j \neq n$, i e_{n2} ne može biti uparen ni sa jednim $e_{i1} \in O_1$ tako da je $i \neq m$.
- Pronalaženje najveće sličnosti preostalih parova entiteta.
- Ponavljanje postupka dok ne ostane jedna vrednost u matrici.

U sistemu opisanom u ovom radu ovako dobijeni parovi entiteta postaju “najbolji upareni” ukoliko su veći od zadatog praga. Na slici 18 prikazan je primer utvrđivanja uparenih entiteta ovom metodom uz vrednost praga od 0.5.

	e_{12}	e_{22}	e_{32}
e_{11}	0.9	0.2	0.4
e_{21}	0.85	0.5	0.6
e_{31}	0.4	0.8	0.5

Slika 18. Primer utvrđivanja najboljih parova iz matrice sličnosti

U prvom koraku se bira najveća moguća sličnost sadržana u matrici; to je, u primeru sa slike, 0.9, te je prvi par uparenih entiteta: $\{e_{11}, e_{12}\}$. U narednom koraku neće biti izabran par $\{e_{21}, e_{12}\}$ iako je njihova sličnost 0.85, jer je e_{12} već uparen. Naredna najveća sličnost preostalih neuparenih entiteta je, dakle, 0.8, pa je $\{e_{31}, e_{22}\}$ sledeći najbolji par. U poslednjem koraku se uparuju $\{e_{21}, e_{32}\}$.

Opređenjenje da se u disertaciji koristi ova metoda za utvrđivanje “najboljih parova entiteta” proisteklo je iz činjenice da je *Greedy selection* metoda često korišćena u aktuelnim sistemima za usaglašavanje ontologija poput (Huber et al., 2011; Jean-Mary et al., 2009; Ngo et al., 2011). Takođe, u (Faria et al., 2013) se navodi da, na primer, uparivanje kojim se maksimizuje suma sličnosti selektovanih parova, ne predstavlja „optimalno“ rešenje za problem ontološkog usaglašavanja. Kao razlog za tu tvrdnju Faria et al. (2013) navode cilj ontološkog usaglašavanja da se maksimizuje broj tačnih parova i minimizuje broj netačnih parova. Stoga, u kontekstu usaglašavanja ontologija (uz pretpostavku da je vrednost sličnosti u direktnoj vezi sa verovatnoćom da je uparivanje tačno) selektovanje jednog para sa visokom sličnosti (na primer, iznad 90%) može biti ispravnije nego selektovanje dva para sa prosečnom vrednosti sličnosti (50-60%). Primer za takav slučaj prikazan je slici 19.

	e_{12}	e_{22}
e_{11}	0.95	0.6
e_{21}	0.6	0.2

Slika 19. Primer opravdanosti primene *Greedy selection* metode za usaglašavanje ontologija

Kao posledica primene *Greedy selection* metode (uz vrednost praga 0.5) dobio bi se par $\{e_{11}, e_{12}\}$ (ne i $\{e_{21}, e_{22}\}$ jer je vrednost sličnosti ispod praga), dok bi se primenom maksimalnog težinskog mapiranja dobilo 2 para entiteta $\{e_{21}, e_{12}\}$ i $\{e_{11}, e_{22}\}$.

5.2.2. Sličnost zasnovana na poređenju naziva (Terminološka sličnost)

Terminološka sličnost utvrđuje se, pre svega, na osnovu lingvističke sličnosti zasnovane na WordNet leksičkoj bazi podataka. Pre utvrđivanja sličnosti između stringova koji identifikuju određenu klasu, primenjuje se tokenizacija stringova korišćenjem metoda normalizacije stringova (identifikovanje cifara, praznih mesta, znakova interpunkcije, oznaka za novi red i sl.). Kao deo normalizacije uklanjaju se stop reči (za engleski jezik) i zamenjuju velika slova malim slovima. Takođe se primenjuje morfološka lingvistička normalizacija uz upotrebu WordNet baze podataka. Budući da se za dodatni opis koncepata u ontološkim modelima koriste labele, pri poređenju klasa dve ontologije primenom terminološkog uparivača uzimaju se u obzir lokalna imena klasa kao i njihove labele.

Sličnost između dva tokena koji pripadaju lokalnim imenima klasa se utvrđuje primenom Lin-ove „information-theoretic” sličnosti (Lin, 1998) ukoliko se oba tokena nalaze u WordNet rečniku. Ukoliko to nije slučaj, sličnost tokena se određuje primenom Jaro Winkler metode (Jaro, 1989; Winkler, 1999). Lin-ova sličnost je često korišćena metrika u sistemima za usaglašavanje ontologija (Jean-Mary et al., 2009, Li et al., 2009; Ngo & Bellahsene, 2012).

Primenom *Greedy selection* metode na matricu koju čine sličnosti svih mogućih parova tokena poređenih naziva klasa dobija se lista S_{ln} koju čine sličnosti „najboljih uparenih parova“ tokena.

Skills potklase reprezentuju veštine/ishode koji su najčešće opisani slobodnim tekstom. Veća razlika u broju reči sadržanih u ishodima može značajnije uticati na različito značenje ishoda. Stoga je potrebno uzeti u obzir broj tokena pri izračunavanju sličnosti lokanih naziva *Skills* potklasa. Sličnost $s_{ln}(C_{i1}, C_{j2})$ se izračunava na sledeći način.

$$s_{ln}(C_{i1}, C_{j2}) = \frac{2 \sum_{i=0}^m S_{ln}(i)}{(|tok_{i1}| + |tok_{j2}|)}, \quad (5.2.2)$$

$|tok_{ik}|$ – broj tokena u lokalnom imenu klase C_{ik} ,

m – broj elemenata liste S_{ln}

Normalizacija primenjena u (5.2.2), dakle, uzima u obzir broj tokena sadržanih u nazivima klasa tako da se umanjuje sličnost između klasa koje su opisane sa različitim brojem reči (tokena).

Za *Knowledge* potklase, koje su obično opisane manjim brojem tokena i predstavljaju nazive tema/oblasti, eksperimentalno je pokazano da se (za sve tri kombinacije ulaznih ontoloških modela) dobijaju tačniji rezultati ako se primeni različit princip izračunavanja sličnosti u zavisnosti od odnosa razlike u broju tokena naziva klasa

$\left| |tok_{i1}| - |tok_{j2}| \right|$ i minimalnog broja tokena $\min(|tok_{i1}|, |tok_{j2}|)$. Tako, ako razlika broja tokena nije manja od minimalnog broja tokena primenjuje se gornja formula (5.2.2). U suprotnom, ukupna sličnost lokanih naziva dve klase $s_{ln}(C_{i1}, C_{j2})$ se dobija kao prosečna vrednost elemenata liste S_{ln} .

Analogno se izračunava sličnost labela klase $s_{lb}(C_{i1}, C_{j2})$, kao i sličnost između lokalnog imena klase jedne ontologije i labela klase druge ontologije $s_{lnlb}(C_{i1}, C_{j2})$ i, inverzno, $s_{lbtn}(C_{i1}, C_{j2})$.

Ukupna terminološka sličnost je:

$$s_{term}(C_{i1}, C_{j2}) = \max(s_{ln}(C_{i1}, C_{j2}), s_{lb}(C_{i1}, C_{j2}), s_{lnlb}(C_{i1}, C_{j2}), s_{lbtn}(C_{i1}, C_{j2})) \quad (5.2.3)$$

5.2.3. Taksonomijska strukturna sličnost

Taksonomijska strukturna sličnost se izračunava u tri koraka:

- Izračunavanje sličnosti svih roditeljskih klasa (klase koje imaju potklase).
- Izračunavanje sličnosti neroditeljskih klasa (klase koje nemaju potklase – „list“ (engl. *leaf*) potklase) koje su potklase uparenih roditeljskih klasa.
- Izračunavanje sličnosti neroditeljskih klasa („list“ klasa) koje su potklase neuparenih roditeljskih klasa.

Ovakva kompozicija taksonomijskih algoritama omogućava korisniku manuelnu intervenciju nad rezultatima uparivanja nakon uparivanja superklasa, nakon uparivanja potklasa uparenih roditelja, itd. Budući da rezultat svakog narednog uparivača zavisi od rezultata prethodnog (sličnost „list“ potklasa zavisi od uparenosti roditeljskih klasa dve ontologije, relacioni uparivač se zasniva na uparenosti *Knowledge* potklasa, itd.), na ovaj način je omogućeno “rano ispravljanje” uparivanja.

Sličnosti predefinisanih klasa (*Knowledge*, *Competence*, *Informatics_domain_knowledge*, *General_knowledge* itd.) se ne uzimaju u obzir pri ovim izračunavanjima.

5.2.3.1. Izračunavanje sličnosti roditeljskih klasa

Izračunavanje sličnosti roditeljskih klasa je zasnovano na algoritmu opisanom u (Jean-Mary et al., 2009). Za dve roditeljske klase uzimaju se u obzir sličnosti njihovih roditeljskih klasa (superklasa), sličnosti njihovih potklasa („dece“) i njihova terminološka sličnost. Pri tome se posmatraju sličnosti *svih roditelja i dece*, ne samo direktnih. Tako, sličnost superklasa klasa C_{i1} i C_{j2} označena kao $s^{sup}(C_{i1}, C_{j2})$ se određuje po sledećem algoritmu.

```
/* Neka je  $A_{ij}$  klasa ontologije, pri čemu,  $A_{i1} \in O_1$  i  $A_{i2} \in O_2$ 
if  $\exists A_{i1} | C_{i1} \subseteq A_{i1}$  or  $\exists A_{i2} | C_{j2} \subseteq A_{i2}$  then
```

```

 $s^{sup}(C_{i1}, C_{j2})$  se ne uzima u obzir
else
    Neka je  $C_{i1} \subseteq \{A_{k1}\}, k = 1, n; n \geq 1$  and  $C_{j2} \subseteq \{A_{l2}\}, l = 1, m; m \geq 1$ 
    for k = 1 to n
        for l = 1 to m
/* u matrici sa n vrsta i m kolona postavljaju se vrednosti
sličnosti klase iz skupa  $\{A_{11}, A_{21} \dots A_{n1}\}$  sa klasama iz
skupa  $\{A_{12}, A_{22} \dots A_{m2}\}$ 
        matrix[k][l] =  $s_{term}(A_{k1}, A_{l2})$ 
/* lista najboljih parova superklasa  $S^{sup}$  se dobija primenom
Greedy selection metode na matrix
         $S^{sup} = \text{Greedy\_Selection\_Method}(\text{matrix})$ 
/*  $s^{sup}(C_{i1}, C_{j2})$  se postavlja na prosečnu vrednost sličnosti
uparenih superklasa
 $s^{sup}(C_{i1}, C_{j2}) = \frac{\sum_{i=0}^m S^{sup}(i)}{m}, m = \text{size of } S^{sup}$ 

```

Sličnost potklasa $s^{sub}(C_{i1}, C_{j2})$ se izračunava na analogan način formiranjem matrice koju čine sličnosti svake potklase C_{i1} sa svakom potklasom klase C_{j2} i izračunavanjem prosečne vrednosti *uparenih* potklasa.

Ukupna sličnost $s_{parent}(C_{i1}, C_{j2})$ klase C_{i1} i C_{j2} se izračunava kao srednja vrednost terminološke sličnosti i prethodno sračunatih sličnosti superklasa i potklasa uz uslov da svaka klasa para ima najmanje jednu potklasu, a ukoliko uslov nije zadovoljen ukupna sličnost je 0.

Najvažnija razlika u odnosu na algoritam predstavljen u (Jean-Mary et al., 2009) se odnosi na slučaj kada samo jedna klasa iz upoređivanog para roditeljskih klasa nema svoje superklase. U tom slučaju $s^{sup}(C_{i1}, C_{j2})$ se ne uzima u obzir pri izračunavanju ukupne sličnosti $s_{parent}(C_{i1}, C_{j2})$, dok se u (Jean-Mary et al., 2009) $s^{sup}(C_{i1}, C_{j2})$ postavlja na nulu.

Matrica sličnosti taksonomijskog uparivača označena sa S_{parent} je dimenzije $m \times n$ gde je m i n ukupan broj *Knowledge* potklasa u ontologijama O_1 i O_2 , respektivno. Lista uparenih klasa A_{parent} se dobija primenom *Greedy selection* algoritma na matricu S_{parent} .

5.2.3.2. Izračunavanje sličnosti „list“ klasa uparenih roditelja

U ovoj fazi se izračunava sličnost, pre svega, između „list“ klasa poređenih ontologija, pri čemu se pod „list“ klasama podrazumevaju klase koje nemaju potklase. Sličnost $s_{leaf}(C_{i1}, C_{j2})$ se izračunava na sledeći način:

```

/* Neka je  $A_{ij}$  klasa ontologije, pri čemu važi da  $A_{i1} \in O_1$  i
 $A_{j2} \in O_2$ .

```

```

/* Neka, dalje, važi sledeće:  $C_{i1}$  je "list" klasa ontologije  $O_1$ 
i  $C_{j2}$  je "list" klasa ontologije  $O_2$ , ili  $C_{i1}$  je "list" klasa
ontologije  $O_1$  i klasa  $C_{j2}$  ima samo "list" potklase, ili  $C_{j2}$  je
"list" klasa ontologije  $O_2$  i  $C_{i1}$  ima samo "list" potklase.
If  $\exists\{A_{i1}, A_{k2}\} \{A_{i1}, A_{k2}\} \in A_{parent}, A_{i1} \in \{A_{11} \dots A_{n1}\}, C_{i1} \subseteq \{A_{11} \dots A_{n1}\},$ 
 $A_{k2} \in \{A_{12} \dots A_{m2}\}, C_{j2} \subseteq \{A_{12} \dots A_{m2}\}$  then
     $s_{leaf}(C_{i1}, C_{j2}) = s_{term}(C_{i1}, C_{j2})$ 
else
     $s_{leaf}(C_{i1}, C_{j2}) = s_{parent}(C_{i1}, C_{j2})$ 

```

Uočljivo je da se ne porede samo „list“ klase već i klase koje imaju samo „list“ potklase i „list“ klasa. Na ovaj način se ostavlja mogućnost da je neka klasa u jednom od kurikuluma detaljnije opisana potklasama, što ne mora nužno da znači da nije ekvivalentna sa poređenom klasom. Ovaj izuzetak je obuhvaćen i superklasa/potklasa relacijom opisanom u 1:N algoritmu (sekcija 5.2.5).

5.2.3.3. Izračunavanje sličnosti „list“ klasa neuparenih roditelja

Dakle, prethodnim uparivačem (opisanim u sekciji 5.2.3.2.) sličnost „list“ klasa $s_{leaf}(C_{i1}, C_{j2})$ postaje nula ukoliko nijedan roditelj klase C_{i1} , primenom taksonomijskog uparivača (opisanim u sekciji 5.2.3.1.), nije uparen ni sa jednim roditeljem klase C_{j2} . Međutim, analizom kurikuluma može se uočiti da bi primenom te metode neki suštinski srodni koncepti (različitih roditelja) ostali neupareni. Primer za to su reprezentacije koncepata kompjuterske grafike i multimedije u kreiranim ontološkim modelima srednjoškolskog informatičkog kurikuluma i referentnog nastavničkog kurikuluma. Srednjoškolski kurikulum, na osnovu dostupne literature o ACM K12 modelu, koncepte računarske grafike (teme vektorske i rasterske grafike, na primer) predviđa u L2 nivou u okviru teme ‘Multimedia’. Stoga su koncepti računarske grafike mapirani kao potklase *Multimedia* klase. Sa druge strane, u mnogim nastavničkim kurikulumima, pojmovi koji se odnose na računarsku grafiku su, najčešće, prikazani u okviru posebnog kursa, dok se teme multimedije izučavaju ili u posebnom kursu ili u okviru oblasti upravljanja informacijom, interakcije čovek – računar i slično. Stoga su u referentnom nastavničkom kurikulumu koncepti računarske grafike predstavljeni kao potklase posebne klase *Graphics*, dok postoji i posebna roditeljska klasa *Multimedia* koja uopšte ne sadrži potklase koje odgovaraju temama računarske grafike. Dakle, primenom prethodnog algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja za, na primer, klase, u koje su mapirani koncepti rasterskih slika, dobila bi se vrednost sličnosti nula, budući da im roditelji nisu upareni (klasa *Multimedia* srednjoškolskog modela kurikuluma nije uparena sa klasom *Graphics* nastavničkog modela kurikuluma već sa klasom *Multimedia*). Ipak, ovi koncepti jesu srodni, odnosno instance klase na koje su mapirani ne bi bile disjunktne.

Svojstvo da dve klase jesu disjunktne se u OWL-u označava sa *owl:disjointWith*. Referentni nastavnički kurikulum u ovom radu je modelovan tako da su sve međusobno disjunktne oblasti kurikuluma mapirane u ontologiju primenom „disjoint“ opcije Protégé softverskog paketa. Tako, klase *Graphics* i *Multimedia* nisu mapirane kao disjunktne (“disjoint”).

Princip određivanja sličnosti klasa $s_{disj}(C_{i1}, C_{j2})$ je sledeći:

```

/* Neka je  $A_{leaf}$  lista parova klasa uparenih primenom algoritma
za izračunavanje sličnosti "list" klasa uparenih roditelja.
/* Neka važi:  $C_{i1} \subseteq \{A_{11}, A_{21}, \dots, A_{n1}\}, C_{j2} \subseteq \{B_{12}, B_{22}, \dots, B_{m2}\}$ 
If  $C_{i1}$  i  $C_{j2}$  su neuparene "list" klase and
 $\exists \{A_{k1}, A_{o2}\} | \{A_{k1}, A_{o2}\} \in A_{leaf}, A_{k1} \in \{A_{11}, A_{21}, \dots, A_{n1}\}$  and
 $\nexists A_{o2}, B_{l2}$  definisane kao "disjoint" klase  $\{A_{k1}, A_{o2}\} \in A_{leaf}, A_{k1} \in$ 
 $\{A_{11}, A_{21}, \dots, A_{n1}\}, B_{l2} \in \{B_{12}, B_{22}, \dots, B_{m2}\}$  and
 $\nexists \{A_{k1}, B_{l2}\} | \{A_{k1}, B_{l2}\} \in A_{leaf}, A_{k1} \in \{A_{11}, A_{21}, \dots, A_{n1}\}, B_{l2} \in \{B_{12}, B_{22}, \dots, B_{m2}\}$ 
then
 $s_{disj}(C_{i1}, C_{j2}) = s_{term}(C_{i1}, C_{j2})$ 
else
 $s_{disj}(C_{i1}, C_{j2}) = s_{leaf}(C_{i1}, C_{j2})$ 

```

Ovaj uparivač se u sekvencijalnoj kompoziciji nalazi iza uparivača kojim se utvrđuje sličnost „list“ klasa uparenih roditelja čime se privileguju klase čiji su roditelji upareni, ali se, takođe, proširuje prostor pretraživanja na druge, „ne-disjoint“ klase koje mogu predstavljati odgovarajuće koncepte. Ovde se takođe uzimaju u obzir parovi klasa od kojih je jedna „list“ klasa a druga roditelj samo neuparenim „list“ klasama.

Budući da se u svim taksonomijskim uparivačima proveravaju sličnosti potklasa klase *Knowledge* u obe ontologije, sličnosti potklasa klase *Skills* ostaju iste kao u matrici sličnosti nakon terminološkog uparivanja.

Sličnosti ostalih kombinacija klasa se utvrđuju na sledeći način:

- Sličnosti potklasa klase *Knowledge* prve ontologije i potklasa klase *Skills* druge ontologije postaju nula.
- Sličnosti potklase klase *Skills* prve ontologije i potklase klase *Knowledge* druge ontologije postaju nula.
- U slučaju poređenja srednjoškolskog modela i nastavničkog modela klase koje pripadaju neinformatičkim znanjima u nastavničkom kurikulumu ostaju neuparene.

5.2.4. Relaciona sličnost

Ishodi/ciljevi kurseva ili tematskih oblasti su, pri modelovanju ontološke reprezentacije, jednostavno mapirani u odgovarajuće *Skills* potklase kao direktne potklase klasa revidirane Blumove taksonomije. Te (pot)klase, obično, dalje nisu strukturno organizovane u predstavljenim ontološkim modelima. Stoga određivanje sličnosti

među klasama na osnovu njihove taksonomijske strukture nije odgovarajuće za ovaj deo ontologije. S druge strane, ishodi kurikuluma (mapirani u odgovarajuće *Skills* potklase) su najčešće opisani većim “slobodnim tekstom”, te primena samo terminološkog uparivača ne bi dala adekvatne rezultate.

Sličnost *Skills* potklasa se u sistemu izračunava na osnovu relacionog grafa ontologije. U radu je primenjena metoda izračunavanja sličnosti zasnovana na principu korišćenom u (Maedche & Staab, 2002), a, kasnije i u brojnim sistemima za usaglašavanje ontologija. Po tom principu: *ukoliko su dve klase koje predstavljaju domen objektnog svojstva (relacije) slične i ukoliko je i objektno svojstvo slično, tada su slične i klase opsega sa kojima su te domenske klase povezane.*

Relaciona sličnost $s_{rel}(C_{i1}, C_{j2})$ se određuje na sledeći način

```
/* Neka je  $A_{disj}$  lista parova klasa uparenih primenom algoritma
za izračunavanje sličnosti "list" klasa neuparenih roditelja
```

```
/* Neka je  $C_{Knowledge1}$  Knowledge klasa i neka je  $C_{Skills1}$  Skills
klasa
```

```
If  $C_{i1} \subseteq C_{Knowledge1}$  or  $C_{j2} \subseteq C_{Knowledge2}$  then
```

$$s_{rel}(C_{i1}, C_{j2}) = s_{disj}(C_{i1}, C_{j2})$$

```
else if  $C_{i1} \subseteq C_{Skills1}$  and  $C_{j2} \subseteq C_{Skills2}$  then
```

```
If  $C_{i1}$  povezana sa  $\{A_{11}, A_{21}, \dots, A_{n1}\} | \{A_{11}, A_{21}, \dots, A_{n1}\} \subseteq C_{Knowledge1}$  and
 $C_{j2}$  povezana sa  $\{A_{12}, A_{22}, \dots, A_{m2}\} | \{A_{12}, A_{22}, \dots, A_{m2}\} \subseteq C_{Knowledge2}$  then
```

```
If  $\{A_{k1}, A_{21}, \dots, A_{11}\}$  je skup svih superklasa i potklasa svih
klasa iz  $\{A_{11}, A_{21}, \dots, A_{n1}\}$ ,  $k = n + 1$  and  $\{A_{o2}, A_{22}, \dots, A_{p2}\}$  je skup
svih superklasa i potklasa svih klasa iz  $\{A_{12}, A_{22}, \dots, A_{m2}\}$ ,
 $o = m + 1$  then
```

```
If  $\exists \{A_{q1}, A_{r2}\} | \{A_{q1}, A_{r2}\} \in A_{disj}$ ,
```

```
 $A_{q1} \in \{A_{11}, A_{21}, \dots, A_{n1}\} \cup \{A_{k1}, A_{21}, \dots, A_{11}\}$ ,  $A_{r2} \in \{A_{12}, A_{22}, \dots, A_{m2}\} \cup$ 
```

```
 $\{A_{o2}, A_{22}, \dots, A_{p2}\}$  then
```

$$s_{rel}(C_{i1}, C_{j2}) = s_{term}(C_{i1}, C_{j2})$$

```
else
```

$$s_{rel}(C_{i1}, C_{j2}) = 0$$

Ukoliko u delu ontologije, kojima pripadaju potklase klase *Skills*, postoji struktuiranost, odnosno, ukoliko su neke veštine mapirane u klase koje, dalje, imaju potklase, tada se i za te potklase, pri izračunavanju relacione sličnosti, uzimaju u obzir relacije nasledene od njihovih superklasa.

Budući da je objektno svojstvo koje povezuje *Knowledge* i *Skills* poznato i isto u obe ontologije „cirkularnost“ koja može biti uzrokovana po (Euzenat & Shvaiko, 2007) primenom relacione metode se umanjuje - pored opisanog principa utvrđivanja sličnosti

opsega objektnog svojstva na osnovu sličnosti domena ovde se eksplicitno ne utvrđuje još i sličnost objektnog svojstva na osnovu sličnosti domena i opsega.

5.2.5. Superklasa - potklasa relacije između klasa dve ontologije

Prethodno opisanim algoritmima utvrđivanja sličnosti između klasa (terminološkim, taksonomijskim strukturnim i relacionim) određivalo se u kojoj meri su klase iz ontologije O_1 ekvivalentne sa klasama ontologije O_2 . Na osnovu toga se formirala matrica sličnosti iz koje su selektovane najbolje uparene klase, pri čemu je jedna klasa ontologije O_1 mogla biti uparena sa najviše jednom klasom ontologije O_2 . Dakle, ovim uparivačima se proverava relacija ekvivalencije uz kardinalnost 1:1. Naredna faza utvrđivanja usaglašenosti dve ontologije omogućava uparivanje jedne klase ontologije sa više klasa druge ontologije preko relacije superklasa/potklasa.

Ova metoda zasnovana je na sledećem.

```

/* Neka je Arel lista parova klasa uparenih primenom algoritma
za izračunavanje relacione sličnosti
If {Ci1, Cj2} ∈ Arel and ∄ Ai1 | Ai1 ⊆ Ci1 and ∃ Ak2 | Ak2 ⊆ Cj2 then
  If ∄ {Ai1, Ak2} | {Ai1, Ak2} ∈ Arel, Ai1 ∈ O1, Ak2 ∈ {A12, ..., An2}, {A12, ..., An2} ⊆ Cj2,
  n ≥ 1 then
    For k = 1 to n
      Ak2 ⊆ Ci1
      s1N(Ci1, Ak2) = srel(Ci1, Cj2)

```

Za “suprotan smer” (u kome se određuje da li je klasa C_{j2} superklasa klasama iz O_1) primenjuje se analogan postupak.

Nakon primene ove metode, dakle, jedna klasa može biti povezana sa više klasa druge ontologije i to relacijama superklasa i relacijama ekvivalencije. S druge strane, jedna klasa može biti potklasa klase ontologije kojoj pripada, kao i klase druge ontologije.

5.2.6. Izmene rezultata uparivanja ontologija od strane korisnika sistema

Specifičnost zadatka (upoređivanje kurikuluma predstavljenih ontologijom) i priroda podataka u populisanoj ontologiji (dominantno slobodan tekst) nameću potrebu za podrškom manuelnim intervencijama korisnika-eksperata nad rezultatima uparivanja. Softverska platforma prikazana u ovom radu omogućuje sledeće manuelne intervencije:

1. Uklanjanje veze između uparenih klasa.
2. Promenu stepena podudarnosti uparenih klasa.
3. Izmenu klase u paru.
4. Kreiranje novog uparenog para klasa.
5. Izmene vrednosti praga u *greedy* algoritmu.

Učešće korisnika u izmeni rezultata uparivanja ontoloških reprezentacija kurikuluma je moguća nakon svake faze uparivanja, izuzev terminološkog uparivanja. Pri manuelnim intervencijama uzima se u obzir kardinalnost prethodnog uparivača. Tako, nakon poslednje faze utvrđivanja sličnosti, opisane u sekciji 5.2.5., korisnik može da definiše tip usaglašenosti kao jednu od sledećih relacija $\{=, \subseteq, \supseteq\}$ (ekvivalencija, superklasa, potklasa). Nakon ostalih uparivača podrazumeva se da je izmenjeno ili kreirano mapiranje tipa ekvivalencije, budući da je kardinalnost tih veza 1:1.

Pri kreiranju novih podudarnosti mogući su sledeći slučajevi:

1. Kreiranja novog para $\{C_{i1}, C_{j2}\}$, pri čemu su obe klase neuparene.
2. Kreiranje novog para klasa $\{C_{i1}, C_{j2}\}$, pri čemu je jedna od klasa (C_{i1} ili C_{j2}) već uparena, odnosno: $\exists\{C_{i1}, C_{m2}\} \mid m \neq j$
 $\vee \exists\{C_{k1}, C_{j2}\} \mid k \neq i$.
3. Kreiranje novog para $\{C_{i1}, C_{j2}\}$, pri čemu su obe klase već uparene, odnosno: $\exists\{C_{i1}, C_{m2}\} \mid m \neq j \wedge \exists\{C_{k1}, C_{j2}\} \mid k \neq i$.

Pri kreiranju novog para klasa uzima se u obzir tip nove kao i tip postojećih veza (za slučaj 2 i 3) sa klasom iz druge ontologije. Tako, na primer, za gorenavedeni slučaj 2),

- ukoliko korisnik definiše vezu ekvivalencije $\{C_{i1}, C_{j2}\}$, pri čemu $\exists\{C_{k1}, C_{j2}\} \mid k \neq i$ onda
- briše se veza $\{C_{k1}, C_{j2}\}$ i uspostavlja nova $\{C_{i1}, C_{j2}\}$, budući da jedna klasa može da učestvuje u najviše jednoj vezi ekvivalencije sa klasom druge ontologije.

Analogan algoritam dodavanja novih parova klasa se primenjuje i za ostale slučajeve.

5.2.7. Način prikazivanja rezultata

Krajnji cilj sistema definisanog u ovom radu je utvrđivanje da li nastavnik po okončanju određenog studijskog programa (modelovanog kao što je opisano u 4.3) poseduje kompetencije da predaje informatičke kurseve u srednjoj školi, odnosno (pri poređenju nastavničkih modela) da li su kompetencije izabranog nastavničkog kurikuluma usaglašene sa referentnim nastavničkim kurikulumom. Stoga je, na primer, pri poređenju srednjoškolskog modela i nastavničkog modela, potrebno utvrditi da li postoje teme koje se izučavaju u srednjoj školi a da nisu obuhvaćene nastavničkim kurikulumom kao i da li srednjoškolski kurikulum predviđa sticanje veština koje ne postoje u nastavničkom kurikulumu. Pri tome je važan i nivo revidirane Blumove taksonomije kojima pripadaju veštine – stečena veština predviđena nastavničkim kurikulumom mora biti na istom ili višem nivou od odgovarajuće veštine zahtevane srednjoškolskim informatičkim kurikulumom.

Tako, kao primer poslednjeg principa važi sledeće:

- ako je nastavničkim kurikulumom predviđena samo sposobnost analize/ocenjivanja nekog informatičkog koncepta **i**,
- ako srednjoškolski kurikulum uključuje sposobnost kreiranja istog informatičkog koncepta **i**,
- ako su klase koje reprezentuju te dve veštine opisanim sistemom uparene, **tada**
- nastavnički kurikulum nije dobro usaglašen sa srednjoškolskim kurikulumom, jer analiza pripada nižem nivou revidirane Blumove taksonomije od kreiranja.

U skladu sa opisanim principima, pri poređenju srednjoškolskog modela i nastavničkog modela, sistem prikazuje:

- Uparene klase, tip relacije među njima kao i vrednost sličnosti.
- Broj i procenat uparenih klasa srednjoškolskog kurikuluma:
 - Broj i procenat uparenih potklasa klasa *Knowledge*,
 - Broj i procenat uparenih potklasa klasa *Skills*,
 - Ukupan broj i procenat svih uparenih klasa.
- Listu *Knowledge* potklasa srednjoškolskog kurikuluma koje nisu uparene.
- Listu *Skills* potklasa srednjoškolskog kurikuluma koje nisu uparene.
- Listu klasa srednjoškolskog kurikuluma koje su uparene u relaciji potklasa sa klasama nastavničke ontologije.
- Listu uparenih potklasa klase *Skills* nastavničkog kurikuluma čija superklasa reprezentuje niži nivo revidirane Blumove taksonomije od superklase odgovarajuće veštine srednjoškolskog kurikuluma.

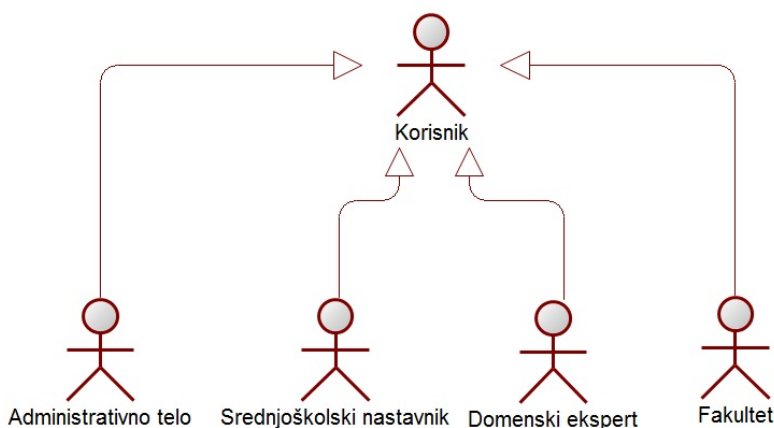
Analogno, pri poređenju modela izabranog nastavničkog kurikuluma i modela referentnog nastavničkog kurikuluma sistem prikazuje:

- Uparene klase, tip relacije među njima kao i vrednost sličnosti.
- Broj i procenat uparenih klasa u oba nastavnička kurikuluma:
 - Broj i procenat uparenih potklasa klasa *Knowledge*,
 - Broj i procenat uparenih potklasa klasa *Skills*,
 - Ukupan broj i procenat svih uparenih klasa.
- Listu klasa oba nastavnička kurikuluma koje nisu uparene.
- Listu klasa oba nastavnička kurikuluma koje su uparene u relaciji potklasa ili superklasa sa klasama poređene ontologije.
- Listu uparenih potklasa klase *Skills* izabranog nastavničkog kurikuluma čije superklase reprezentuju različit nivo revidirane Blumove taksonomije od superklasa odgovarajuće veštine modela referentnog nastavničkog kurikuluma.

5.3. Implementacija softverske platforme za sinhronizaciju informatičkih kurikuluma

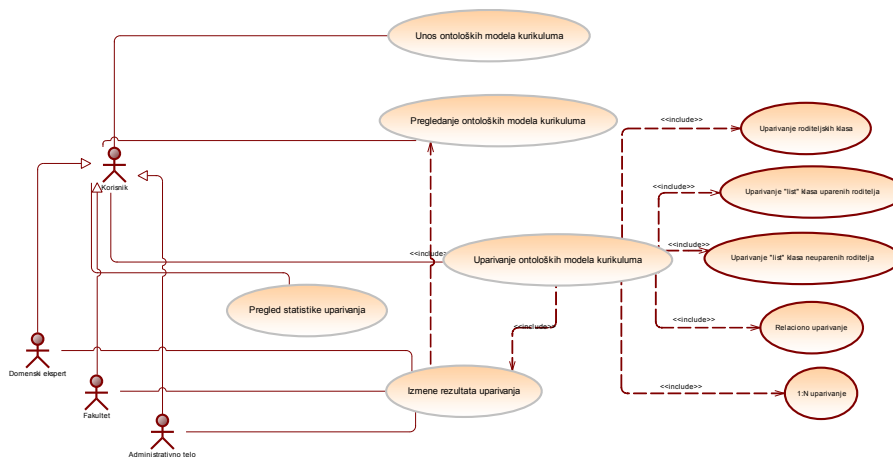
5.3.1. Specifikacija zahteva

U ovoj sekciji opisani su slučajevi korišćenja softverske platforme za sinhronizaciju informatičkih kurikuluma korišćenjem UML (<http://www.uml.org/>) dijagrama slučajeva korišćenja. UML dijagrami su kreirani upotrebom PowerDesigner CASE alata (<http://www.sybase.com/products/modelingdevelopment/powerdesigner/>). Softverska platforma ima četiri grupe korisnika: *Administrativno telo*, *Srednjoškolski nastavnik*, *Domenski ekspert* i *Fakultet* (slika 20).



Slika 20. Grupe korisnika softverske platforme za sinhronizaciju informatičkih kurikuluma

Administrativno telo može označavati Školsku upravu, nadležno Ministarstvo, tela za akreditaciju, univerzitetska stručna tela i sl. *Srednjoškolski nastavnik* podrazumeva zaposlene srednjoškolske nastavnike informatike. *Domenski ekspert* se odnosi pre svega na eksperte iz informatičke (domenske) oblasti i oblasti obrazovanja nastavnika informatike. Grupa korisnika *Fakultet* podrazumeva visokoškolske institucije koje realizuju studijske programe za nastavnike informatike.



Slika 21. Dijagram slučajeva korišćenja

Na slici 21 je dat dijagram slučajeva korišćenja sistema za sinhronizaciju informatičkih kurikuluma. U nastavku su opisani svi slučajevi korišćenja.

Unos ontoloških modela kurikuluma

Kratak opis: Korisnik sistema unosi ontološke modele kurikuluma

Učesnici: Korisnik

Preduslovi: Nema

Opis:

1. Korisnik odabira izvornu i ciljnu ontologiju koje će biti predmet upoređivanja.
2. Korisnik otvara ontološke modele.
3. Korisnik definiše tip ulazne i ciljne ontologije. Moguće kombinacije ulaznih ontoloških modela su:
 - srednjoškolski informatički kurikulum i referentni nastavnički kurikulum,
 - srednjoškolski informatički kurikulum i izabrani nastavnički kurikulum,
 - izabrani nastavnički kurikulum i referentni nastavnički kurikulum.
4. Prikazuju se ontološki modeli u formi stabla.

Izuzetak: Nema

Postuslovi: Prikazani su ontološki modeli odabrane izvorne i ciljne ontologije u formi stabla

Uparivanje ontoloških modela kurikuluma

Kratak opis: Korisnik pokreće automatsko uparivanje ontoloških modela kurikuluma

Učesnici: Korisnik

Preduslovi: Uneti su ontološki modeli kurikuluma

Opis:

1. Korisnik unosi vrednost praga.
2. Korisnik pokreće uparivanje.
3. Uparivanje uključuje uparivače opisane u sekcijama 5.2.2. – 5.2.5.
4. Prikazuju se svi parovi uparenih klasa sa tipom relacije, vrednosti sličnosti i, za *Skills* potklase, usaglašenosti kategorija revidirane Blumove taksonomije.

Izuzetak: Nema

Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

Uparivanje roditeljskih klasa

Kratak opis: Korisnik pokreće automatsko uparivanje roditeljskih klasa

Učesnici: Korisnik

Preduslovi: Uneti su ontološki modeli kurikuluma

Opis:

1. Korisnik unosi vrednost praga.
2. Korisnik pokreće uparivanje zasnovano na algoritmu za utvrđivanje sličnosti roditeljskih klasa opisanim u sekciji 5.2.3.1.

3. Prikazuju se svi parovi uparenih klasa sa tipom relacije i vrednosti sličnosti.

Izuzetak: Nema

Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

Uparivanje „list“ klasa uparenih roditelja

Kratak opis: Korisnik pokreće automatsko uparivanje „list“ klasa uparenih roditelja

Učesnici: *Korisnik*

Preduslovi: Realizovano je uparivanje roditeljskih klasa

- Opis:*
1. Korisnik (opciono) unosi novu vrednost praga.
 2. Korisnik pokreće uparivanje zasnovano na algoritmu za utvrđivanje sličnosti „list“ klasa uparenih roditelja opisanim u sekciji 5.2.3.2.
 3. Prikazuju se svi parovi uparenih klasa sa tipom relacije i vrednosti sličnosti.

Izuzetak: Nema

Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

Uparivanje „list“ klasa neuparenih roditelja

Kratak opis: Korisnik pokreće automatsko uparivanje „list“ klasa neuparenih roditelja

Učesnici: *Korisnik*

Preduslovi: Realizovano je uparivanje „list“ klasa uparenih roditelja

- Opis:*
1. Korisnik (opciono) unosi novu vrednost praga.
 2. Korisnik pokreće uparivanje zasnovano na algoritmu za utvrđivanje sličnosti „list“ klasa neuparenih roditelja opisanim u sekciji 5.2.3.3.
 3. Prikazuju se svi parovi uparenih klasa sa tipom relacije i vrednosti sličnosti

Izuzetak: Nema

Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

Relaciono uparivanje

Kratak opis: Korisnik pokreće automatsko relaciono uparivanje

Učesnici: *Korisnik*

Preduslovi: Realizovano je uparivanje „list“ klasa neuparenih roditelja

- Opis:*
1. Korisnik (opciono) unosi novu vrednost praga.
 2. Korisnik pokreće uparivanje zasnovano na algoritmu za utvrđivanje relacione sličnosti opisanim u sekciji 5.2.4.
 3. Prikazuju se svi parovi uparenih klasa sa tipom relacije, vrednosti sličnosti i, za *Skills* potklase, usaglašenosti kategorija revidirane Blumove taksonomije.

Izuzetak: Nema
Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

1:N uparivanje

Kratak opis: Korisnik pokreće automatsko 1:N uparivanje
Učesnici: *Korisnik*
Preduslovi: Realizovano je relaciono uparivanje
Opis:

1. Korisnik (opciono) unosi novu vrednost praga.
2. Korisnik pokreće uparivanje zasnovano na algoritmu za utvrđivanje superklasa - potklasa relacije između klasa dve ontologije opisanim u sekciji 5.2.5.
3. Prikazuju se svi parovi uparenih klasa sa tipom relacije, vrednosti sličnosti i, za *Skills* potklase, usaglašenosti kategorija revidirane Blumove taksonomije

Izuzetak: Nema
Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma

Pregledanje ontoloških modela kurikuluma

Kratak opis: Korisnik pregleda ontološke modele
Učesnici: *Korisnik*
Preduslovi: Uneti su ontološki modeli kurikuluma
Opis:

1. Korisnik pretražuje klase izabrane ontologije prikazane u formi stabla.
2. Za odabranu klasu prikazuje se njena struktura, labele, komentari i klase sa kojima je preko objektnih svojstava *hasKnowledge/hasSkills* povezana.
3. Ako se ovaj slučaj korišćenja koristi nakon slučaja korišćenja *Uparivanje ontoloških modela kurikuluma* tada će, za selektovanu klasu, biti prikazana i lista klasa sa kojima je ona uparena.

Izuzetak: Nema
Postuslovi: Prikazane su svojstva i struktura odabrane klase

Pregled statistike uparivanja

Kratak opis: Korisnik pregleda statističke rezultate dobijene nakon uparivanja ontologija
Učesnici: *Korisnik*
Preduslovi: Realizovana je neka od faza uparivanja
Opis:

1. Korisnik bira prikaz statističkih rezultata.
2. Za izvornu i ciljnu ontologiju prikazuju se:
 - Procenat uparenih *Knowledge* potklasa,
 - Procenat uparenih *Skills* potklasa,
 - Ukupan procenat uparenih klasa,
 - Procenat i listu klasa uparenih u relaciji superklasa ili potklasa,

- Procenat i lista *Skills* potklasa koje ne reprezentuju odgovarajući nivo revidirane Blumove taksonomije.

Izuzetak: Nema

Postuslovi: Prikazana je statistika uparivanja

Izmene rezultata uparivanja

Kratak opis: Administrativna tela, eksperti u domenskoj oblasti ili fakulteti vrše izmene rezultata uparivanja

Učesnici: *Administrativno telo, Domenski ekspert i Fakultet*

Preduslovi: Realizovana je neka od faza uparivanja

- Opis:*
1. Korisnik (opciono) unosi novu vrednost praga.
 2. Učesnici pronalaze dobijene parove klasa koje nisu tačno uparene kao i parove ekvivalentnih klasa koje sistem nije pronašao.
 3. Učesnici vrše izmene rezultata uparivanja iz prozora koji sadrži rezultate usaglašavanja ili iz prozora koji prikazuje ontološke modele kurikuluma, pri čemu je moguće:
 - Ukloniti par klasa koji nije ispravno uparen,
 - Promeniti klasu u paru,
 - Dodati novi par klasa,
 - Izmeniti vrednosti sličnosti para klasa.
 3. Učesnici snimaju izmene.
 4. Nakon izvršenih izmena sistem pokreće *Greedy selection* algoritam za utvrđivanje najboljih parova klasa.

Izuzetak: Nema

Postuslovi: Prikazani su rezultati usaglašavanja ontoloških modela kurikuluma nakon manuelnih intervencija učesnika

5.3.2. Arhitektura sistema

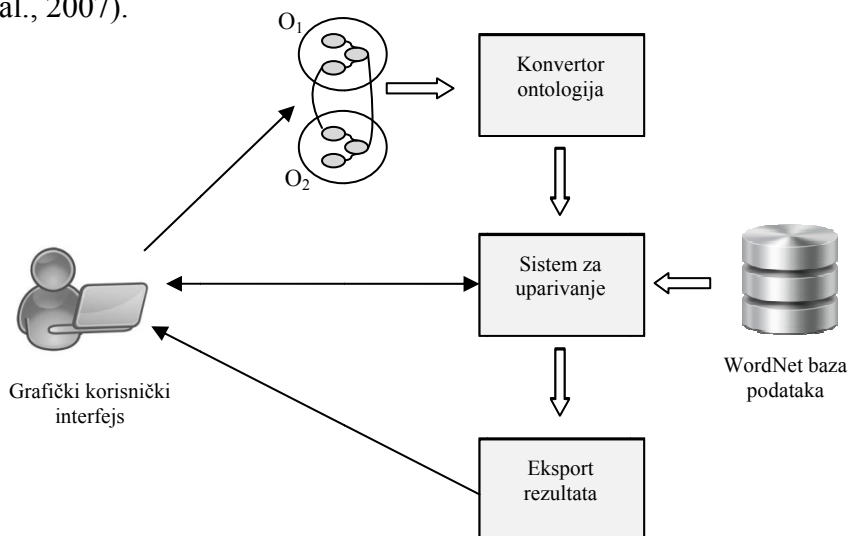
Na slici 22 je prikazana arhitektura sistema za sinhronizaciju kurikuluma zasnovanog na tehnologijama Semantičkog Weba. Softverska platforma je razvijena u programskom jeziku Java. Arhitektura sistema je bazirana na arhitekturi AgreementMaker sistema predloženoj u (Cruz et al., 2007). Osnovne komponente softvera predstavljaju:

- Konvertor ontologija,
- Sistem za uparivanje,
- Eksport rezultata i
- Grafički korisnički interfejs.

5.3.2.1. Konvertor ontologija

Ova komponenta omogućava konvertovanje OWL/RDF ulaznih ontoloških modela kurikuluma u odgovarajuće klase u programskom jeziku Java. Konvertor ontologija, dakle, obezbeđuje čitanje i korišćenje ulaznih ontologija u okviru softverske platforme.

Ontologije su mapirane primenom Protégé OWL API-ja (http://protegewiki.stanford.edu/wiki/ProtegeOWL_API_Programmers_Guide), Jena API-ja (<https://jena.apache.org/>) i OWL *Pellet*-a (Sirin et al., 2007).



Slika 22. Arhitektura sistema za sinhronizaciju kurikuluma

5.3.2.2. Sistem za uparivanje

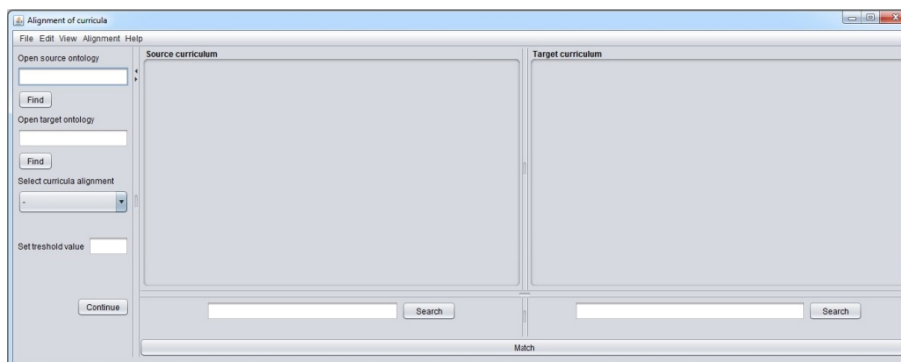
Sistem za uparivanje obezbeđuje usaglašavanje ulaznih ontoloških modela primenom algoritama opisanih u sekciji 5.2. uz korišćenje eksternog WordNet rečnika.

5.3.2.3. Ekspert rezultata

Ova komponenta daje tabelarni i statistički prikaz rezultata opisan u sekciji 5.2.7. Ekspert rezultata treba da omogući konvertovanje rezultata u neki od preporučenih formata u (Euzenat & Shvaiko, 2007) za prikazivanje rezultata usaglašavanja (poput *Alignment format*-a, OWL-a itd.), što u prototipu softvera nije realizovano.

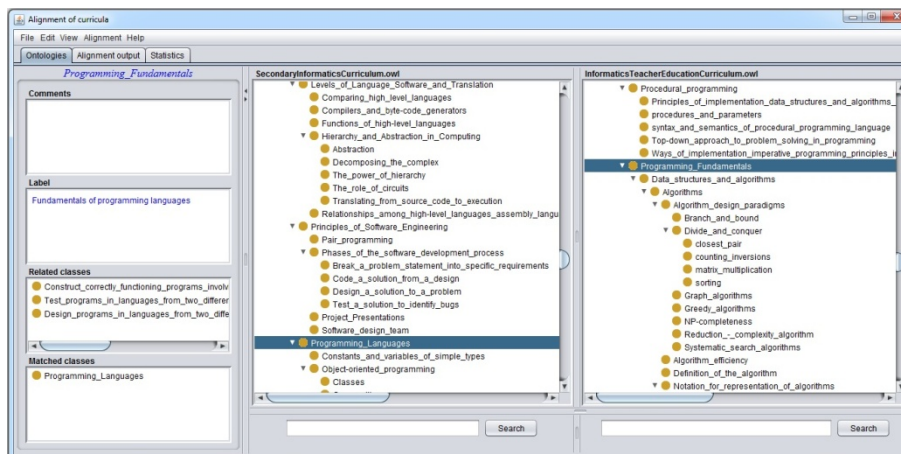
5.3.2.4. Grafički korisnički interfejs

Grafički korisnički interfejs omogućava korisniku uvid u strukturu ontologija, svojstva klasa kao i u rezultate uparivanja. Pri pokretanju softvera dobija se prozor kao na slici 23.



Slika 23. Početni ekran softverske platforme

Učitani ontološki modeli se prikazuju u formi stabla, dok se u zasebnim poljima prikazuju svojstva odabrane klase (slika 24).



Slika 24. Prikaz klase *Programming_Fundamentals*

Nakon svake faze uparivanja u posebnim jezičcima (engl. *tab*) se prikazuju rezultati uparivanja u formi tabele (slika 25) i statistika dobijenih rezultata. Izmene nad rezultatima uparivanja je moguće izvršiti u jezičku koji sadrži tabelarni prikaz rezultata (promenom sadržaja polja u tabeli) kao i u jezičku koji prikazuje poredene ontološke modele (izborom odgovarajuće opcije iz padajućeg menija). Takođe, korisniku se olakšava utvrđivanje tačnosti rezultata uparivanja tako što se odabrani par klasa označava u jezičku koji prikazuje učitane ontologije (slika 24).

R...	Source class	Target class	Type of relation	Similarity value
5	Databases	Database	Equivalence	93.84%
6	Ethical_and_Social_Issues	Computer_ethics	Equivalence	88.36%
7	Ethical_and_security_Issues	Social_and_Professional_Issues_and_Inf...	Equivalence	87.63%
8	Fundamentals_of_Hardware_Design	Computer_design	Equivalence	87.4%
9	Hierarchy_and_Abstraction_in_Computing	Computing_methodologies	Equivalence	70.5%
10	Human_Computer_Interaction	Human_Computer_Interaction	Equivalence	92.09%
11	Internet_concepts	Internet_basics	Equivalence	88.37%
12	Internet_elements	Standard_Internet_elements	Equivalence	90.4%
13	Levels_of_Language_Software_and_Translati...	Programming_paradigms	Equivalence	68.69%
14	Limits_of_Computing	Limits_of_computing	Equivalence	99.58%
15	Models_of_Intelligent_Behavior	Artificial_intelligence	Equivalence	77.13%
16	Multimedia	Multimedia	Equivalence	88.95%
17	Object-oriented_programming	Object-oriented_programming	Equivalence	95.64%
18	Operating_Systems	Operating_systems	Equivalence	96.02%
19	Parts_of_a_Computer	Computer_architecture_and_organization	Equivalence	83.52%
20	Phases_of_the_software_development_proce...	Models_and_phases_of_the_software_de...	Equivalence	95.96%
21	Principles_of_Software_Engineering	Software_engineering	Equivalence	91.41%
22	Principles_of_computer_organization	Architecture_and_Organization	Equivalence	88.79%
23	Privacy_and_Security	Privacy_and_security	Equivalence	88.34%
24	Problem_Solving_and_Algorithms	Problem_solving	Equivalence	98.24%
25	Problem_solving	Problem_solving_phases	Equivalence	96.36%
26	Programming_Languages	Programming_Fundamentals	Equivalence	93.19%
27	Representing_Information_Digitally	Data_representation	Equivalence	85.25%

Slika 25. Tabelarni prikaz rezultata uparivanja

5.4. Evaluacija sistema za usaglašavanje nastavničkog referentnog i srednjoškolskog informatičkog kurikuluma

Razvijena softverska aplikacija je verifikovana utvrđivanjem usaglašenosti kreiranih ontoloških modela opisanih u sekcijama 4.3.5, 4.3.6. i 4.3.7. Validaciju softverske platforme dodatno je realizovao tim obrazovnih eksperata, formiran tako da reprezentuje profile mogućih korisnika sistema. Tim je sačinjen od: 4 univerzitetska nastavnika (izabrana u zvanje informatika sa metodikom nastave), 2 radnika Školske uprave (Ministarstva prosvete, nauke i tehnološkog razvoja) i 2 srednjoškolska nastavnika informatike. Njihov zadatak bio je određivanje očekivanih parova klasa (referentno usaglašavanje) za sve kombinacije ulaznih ontoloških modela kurikuluma. Na osnovu referentnih usaglašavanja i dobijenih izlaza sistema utvrđeni su preciznost, odziv i F-mera sistema, prikazani u odgovarajućim sekcijama (5.4.6, 5.5.6 i 5.6.6).

U ovoj sekciji prikazani su rezultati upoređivanja kurikuluma (srednjoškolskog i referentnog nastavničkog) opisanih u odeljcima 4.3.5. i 4.3.6. Prikaz je organizovan tako da prati korake (uparivače) koji se izvršavaju nakon terminološkog usaglašavanja i sadrži analizu dobijenih rezultata.

5.4.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa

Na slici 26 je, kao primer ispisa rada sistema, prikazan deo uparenih klasa poređenih kurikuluma nakon primene prvog taksonomijskog strukturnog algoritma koji utvrđuje sličnost klasa koje imaju makar jednu potklasu.

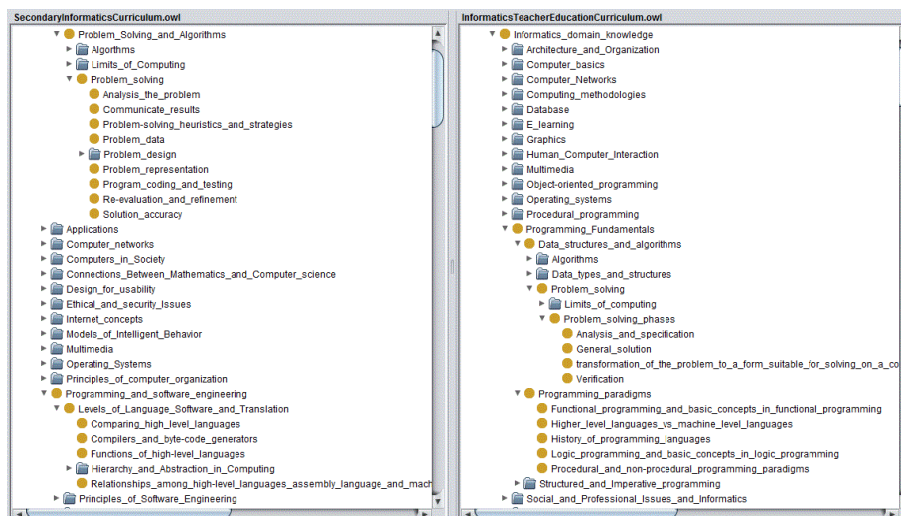
Kolona “Source class” i “Target class” sadrže lokalne nazive klasa ontoloških reprezentacija srednjoškolskog i referentnog nastavničkog kurikuluma; kolona “Type of relation” označava tip veze između klasa (“Equivalence”, “Superclass” i “Subclass”), dok “Similarity value” označava vrednost sličnosti između uparenih klasa. Prag je, eksperimentalno, postavljen na 68%.

Row	Source class	Target class	Type ...	Similar...
1	Computers_in_Society	Informatics_basics	Equiv...	85.26%
2	Connections_Between_Mathematics_and_Com...	Mathematical_basis_of_informatics	Equiv...	84.84%
3	Data_Structures	Data_types_and_structures	Equiv...	97.98%
4	Ethical_and_Social_Issues	Computer_ethics	Equiv...	88.36%
5	Fundamentals_of_Hardware_Design	Computer_design	Equiv...	87.4%
6	Hierarchy_and_Abstraction_in_Computing	Computing_methodologies	Equiv...	70.5%
7	Human_Computer_Interaction	Human_Computer_Interaction	Equiv...	92.09%
8	Levels_of_Language_Software_and_Translation	Programming_paradigms	Equiv...	68.69%
9	Models_of_Intelligent_Behavior	Artificial_intelligence	Equiv...	77.13%
10	Multimedia	Multimedia	Equiv...	88.95%
11	Object-oriented_programming	Object-oriented_programming	Equiv...	95.64%
12	Phases_of_the_software_development_process	Models_and_phases_of_the_software...	Equiv...	95.96%
13	Principles_of_Software_Engineering	Software_engineering	Equiv...	91.41%
14	Principles_of_computer_organization	Architecture_and_Organization	Equiv...	88.79%
15	Privacy_and_Security	Privacy_and_security	Equiv...	88.34%
16	Problem_Solving_and_Algorithms	Problem_solving	Equiv...	98.24%
17	Problem_solving	Problem_solving_phases	Equiv...	96.36%
18	Programming_Languages	Programming_Fundamentals	Equiv...	93.19%
19	Representing_Information_Digitally	Data_representation	Equiv...	85.25%
20	Structured_programming	Structured_and_Imperative_programming	Equiv...	94.23%
21	The_major_component_parts_of_the_micropro...	Central_processing_unit_-_CPU	Equiv...	79.86%
22	Tools_for_problem_design	Notation_for_representation_of_algorith...	Equiv...	88.05%

Slika 26. Deo uparenih klasa nakon primene algoritma za utvrđivanje sličnosti roditeljskih klasa

Sa slike 26 se može zapaziti da su pojedine klase identičnog naziva uparene sa vrednostima sličnosti ispod 100%, kao i da su neke klase uparene iako nemaju slične nazive. Razlozi za to mogu biti postojanje dodatnog opisa u labeli za neke klase, uzimanje u obzir sinonima pri izračunavanju terminološke sličnosti, kao i učešće sličnosti superklasa/potklasa, ukoliko postoje, u izračunavanju ukupne sličnosti klasa.

Pored toga, uočljivo je da neke klase koje imaju ista imena u srednjoškolskom i referentnom nastavničkom kurikulumu (na primer, “Problem solving”) nisu međusobno uparene, već su uparene klase *Problem solving* srednjoškolskog kurikulumu i *Problem solving phases* referentnog nastavničkog kurikulumu (red 17) što se može smatrati ispravnim, jer potklase obeju uparenih klasa predstavljaju faze pri algoritamskom rešavanju problema (slika 27).



Slika 27. Deo hijerarhijske strukture srednjoškolskog modela i referentnog nastavničkog modela

Takođe, posmatrajući samo nazive uparenih klasa sa slike 26, uparivanje klasa *Levels_of_Language_Software_and_Translation* i *Programming_paradigms* (red 8) bi se moglo protumačiti kao netačno. Međutim, sa slike 27 je uočljivo da su u njihove potklase mapirane srodne teme srednjoškolskog i referentnog nastavničkog kurikuluma (veze i poređenje jezika visokog nivoa i mašinskih jezika, nivoi programskih jezika i sl.).

Na ovom nivou primene strukturnog uparivača postoje parovi netačno uparenih klasa. Primer je par klasa: *{Hierarchy_and_Abstraction_in_Computing, Computing_methodologies}* (red 6, slika 26). Budući da njihove roditeljske klase nisu uparene ovaj par klasa može imati uticaja na sličnosti njihovih potklasa koja se izračunava narednim uparivačima (algoritam opisan u 5.2.3.2.).

Procenat uparenih *Knowledge* potklasa u ovoj fazi je 14.34 %.

5.4.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja

Na slici 28 prikazan je deo uparenih klasa dobijenih nakon primene taksonomijskog strukturnog algoritma kojim se utvrđuje sličnost između klasa ontologija koje nemaju potklase.

...	Source class	Target class	Ty...	Similari...
1	Analysis_the_problem	Analysis_and_specification	Eq...	100.0%
2	Audio_file_types	Audio_formats	Eq...	100.0%
3	Binary_counting_and_switching	Counting_with_Different_Number_Systems	Eq...	80.25%
4	Binary_numbers	Binary_number_system	Eq...	100.0%
5	Challenges_of_modeling_information_digitally	Representation_of_the_different_types_of_inf...	Eq...	71.78%
6	Code_a_solution_from_a_design	Software_deployment	Eq...	100.0%
7	Conversion_among_decimal_binary_and_hex_number_s...	Conversion_among_different_number_systems	Eq...	100.0%
8	Creating_and_maintaining_a_database	Motivation_for_databases	Eq...	87.75%
9	Current_legislation_and_-_or_litigation	Legislation_in_computing	Eq...	83.39%
10	Data_collision_and_network_failure	Network_reliability	Eq...	97.26%
11	Data_communication_over_networks	Connecting_a_local_network_with_a_global_n...	Eq...	73.38%
12	Design_a_Web_site	Principles_of_web_design	Eq...	81.79%
13	Encoded_data_and_integrated_circuits	Characteristics_of_digital_integrated_circuits	Eq...	77.44%
14	Expressions	Graphs	Eq...	71.66%
15	File_systems_and_organization	File_System	Eq...	100.0%
16	How_technology_has_changed_ethical_and_legal_issues	Aspects_of_socially_and_ethically_proper_us...	Eq...	82.12%
17	Identity_-_IP_addresses	IP_addressing_scheme_in_local_network	Eq...	76.31%
18	Information_storage_and_retrieval	The_difference_between_IT_literacy_and_Info...	Eq...	71.53%
19	Intellectual_property_copyright_and_fair_use	Protection_of_intellectual_property_rights	Eq...	100.0%
20	Interface_evaluation	Measures_for_evaluation_in_HCI	Eq...	96.8%
21	Knowledge-based_Systems	Semantic_Web_and_knowledge_representation	Eq...	80.16%
22	Method_-_functions_and_parameters	Method_parameters	Eq...	100.0%
23	Multiple_software_applications_can_be_used_to_compl...	Appropriate_software_applications_for_specifi...	Eq...	83.26%
24	Presentation_software	Software_applications_for_presentations	Eq...	100.0%
25	Repetition	Iteration	Eq...	100.0%
26	Web_development_tools	Tools_for_web_development_and_maintanance	Eq...	100.0%
27	What_is_Intelligence	Definition_of_artificial_intelligence	Eq...	100.0%
28	Word_processing	Software_applications_for_text	Eq...	77.62%

Slika 28. Primer uparenih klasa nakon primene druge faze taksonomijskog strukturnog algoritma

Sličnost uparenih klasa dobijenih u ovoj fazi određena je pre svega terminološkom sličnošću njihovih lokalnih imena i labela uz uslov da su im neke od (direktnih ili nedirektnih) roditeljskih klasa uparene primenom algoritma za *izračunavanje sličnosti roditeljskih klasa*. Tako klase *Repetition* i *Iteration* imaju uparene nedirektne roditelje *Programming_Languages* i *Programming_fundamentals* (slika 26, red 18), te predstavljaju kandidate za uparivanje u ovoj fazi. Dalje, budući da glagoli ‘repeat’ i ‘iterate’ predstavljaju sinonime u WordNet bazi,

primenom terminološkog uparivača dobijena je visoka sličnost klasa *Repetition* i *Iteration*. Stoga su ove klase uparene primenom algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja (slika 28, red 25).

Daljom analizom oba ontološka modela kurikuluma može se zapaziti da, iako terminološki i strukturno slična, postoje i suštinski netačna uparivanja (redovi 8, 11, 14 i 18, slika 28). Ovi parovi klasa predstavljaju prave kandidate za manuelnu intervenciju korisnika.

Primer uparivanja „list“ klase sa klasom koja predstavlja roditelja „list“ klasama je par {*Knowledge-based Systems*, *Semantic_Web_and_knowledge_representation*} (slika 28, red 21). Klasa *Knowledge-based_Systems* nema potklase i predstavlja potklasu klase *Models_of_Intelgent_Behavior*. Klasa *Semantic_Web_and_knowledge_representation* je potklasa klase *Artificial_intelligence* koja ima potklase a uparena je sa klasom *Models_of_Intelgent_Behavior* primenom algoritma za izračunavanje sličnosti roditeljskih klasa (slika 26, red 9).

Procenat uparenih *Knowledge* potklasa u ovoj fazi je 70.22%.

5.4.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja

Prethodnim uparivačem neke od potklasa klase *Multimedia* srednjoškolskog kurikuluma (*Bit-mapped_representation_of_images_and_resolution*, *Vector_versus_bit-mapped_images*, *Image_file_types_and_comperession*) nisu uparene sa potklasama klase *Multimedia* referentnog nastavničkog kurikuluma. Primenom algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja ove klase su uparene sa potklasama klase *Graphics* (slika 29).

...	Source class	Target class	Typ...	Simila...
1	Bit-mapped_representation_of_images_and_resolution	Resolution	Equ...	76.6%
2	Image_file_types_and_compression	Image_Compression	Equ...	81.41%
3	Vector_versus_bit-mapped_images	Methods_of_presenting_static_images_in_com...	Equ...	100.0%

Slika 29. Uparene „list“ klase čiji roditelji nisu upareni

Detaljnijom analizom rezultata dobijenih nakon primene poslednje faze taksonomijskog algoritma i ontoloških modela, i na ovom nivou se mogu zapaziti klase koje nisu uparene. Klase koje ostaju neuparene nakon primene taksonomijskog strukturnog algoritma mogu, ali ne moraju, da ukazuju na nedostajuća znanja u nastavničkom kurikulumu ili na nekompatibilnu strukturu poređenih ontoloških modela kurikuluma i mogu da se iskoriste za unapređenje modela. U nastavku su navedeni neki karakteristični primeri.

Neke od neuparenih klasa reprezentuju znanja koja nisu uopšte sadržana u modelu referentnog nastavničkog kurikuluma (faze mašinskog ciklusa, robotika, tehnike dokumentovanja itd.). One ukazuju na potrebu dopunjavanja referentnog nastavničkog kurikuluma novim znanjima.

Sa druge strane, klasa *Functions_including_parameters_and_mathematical_notation* u srednjoškolskom kurikulumu je potklasa klase *Connections_Between_Mathematics_and_Computer_*

science, a u referentnom nastavničkom kurikulumu odgovarajuća znanja su mapirana na potklasu klase *General_knowledge* koja ne pripada domenskim znanjima. Stoga iz činjenice da ne postoji uparivanje nije moguće izvesti zaključak da ova znanja ne postoje u referentnom nastavničkom kurikulumu, već da je potrebno revidirati strukturu domenskih ontologija ili u uparivanje uključiti i nedomenske ontologije.

Konačno, razlike u struktuiranosti ontologija koje proističu iz dubine izučavanja pojedinih tema u srednjoškolskom i nastavničkom kurikulumu (dubina izučavanja u okviru nastavničkog kurikuluma je očekivano veća) mogu da rezultuju neuparenostima koje ne ukazuju na neadekvatan nastavnički kurikulum. Na primer, tematska oblast srednjoškolskog kurikuluma 'Interdisciplinary Utility of Computers and Problem Solving in the Modern World' (Verno et al., 2006) sadrži fokuse koji se odnose na različite aspekte primene računara ('Embedded systems', 'Education and Training', 'Modeling and design' uključujući CAD i simulacije i sl.). U ontološkom modelu ti fokusi su mapirani na „list“ potklase klase *Interdisciplinary_Utility_of_Computers_and_Problem_Solving_in_the_Modern_World* koje nakon primene taksonomijskog strukturog uparivača nisu uparene ni sa jednom klasom ontološkog modela referentnog nastavničkog kurikuluma. Tako je fokus 'Education and training' (strana 21, Verno et al., 2006) mapiran na jednu od „list“ potklasa klase *Interdisciplinary_Utility_of_Computers_and_Problem_Solving_in_the_Modern_World* srednjoškolskog kurikuluma. Iako referentni nastavnički kurikulum sadrži klase koje odgovaraju ovom fokusu, poput *Educational_technology* i *E-learning*, te klase nisu uparene sa „list“ klasom *Education_and_training*, jer, u skladu sa detaljnijim izučavanjem u nastavničkom kurikulumu, predstavljaju kompleksne klasne strukture (sadrže potklase koje dalje imaju svoje potklase), te ih predloženi uparivači ne razmatraju kao kandidate za uparivanje sa „list“ klasom *Education_and_training*.

Procenat uparenih *Knowledge* potklasa nakon ove faze je 87.13%.

5.4.4. Primena algoritma za izračunavanje relacione sličnosti

Kako su u pitanju nisko struktuirane potklase klase *Skills* (praktično samo struktura kojom se modeluje revidirana Blumova taksonomija), gde naslovi i labele potklasa najčešće sadrže veliki broj reči, terminološko uparivanje značajno utiče na konačan rezultat. Da bi se izbeglo izostavljanje potencijalno korisnih uparivanja koja se mogu iskoristiti za manuelne intervencije, ovde je postavljen niži kriterijum (prag) pri utvrđivanju uparenih klasa (60%). Deo rezultata dobijenih relacionim uparivačem kojim se utvrđuje sličnost između potklasa klase *Skills* prikazan je na slici 30. Procenat uparenih klasa je 80.88%.

R.	Source class	Target class	Ty..	Simil..	Bloom
1	Code_a_program_that_corresponds_to_a_set_of_specifications	Design_programs_in_languages_from_two_different_programming_parad...	Eq..	85.0%	T
2	Convert_between_decimal_binary_and_hexadecimal_numbers	Apply_arithmetic_operations_in_different_number_systems	Eq..	70.04%	T
3	Convert_between_image_formats	Contrast_vector_and_raster_graphics	Eq..	77.24%	T
4	Create_a_Web_site_given_design_specifications	Design_web_pages	Eq..	69.19%	T
5	Create_a_user-centered_design	Design_interactive_user_interfaces_for_diverse_applications	Eq..	74.59%	T
6	Define_intellectual_property_and_state_the_impact_of_provisions_to_protect_it	Discuss_intellectual_property	Eq..	64.16%	T
7	Define_parallel_processing	Use_design_patterns	Eq..	61.94%	T
8	Describe_the_major_applications_of_artificial_intelligence_and_robotics	Apply_Artificial_Intelligence_applications	Eq..	70.7%	T
9	Describe_the_role_of_the_OS_as_an_intermediary_between_application_programs_and_h...	Explain_the_objectives_and_functions_of_modern_operating_systems	Eq..	64.59%	T
10	Design_a_multi-table_relational_database	Project_relational_data_model	Eq..	75.0%	T
11	Determine_if_a_given_algorithm_successfully_solves_a_stated_problem	Select_basic_language_instructions_to_accomplish_a_given_straightfow...	Eq..	64.34%	⊥
12	Display_a_multimedia_object_within_a_Web_page_or_document	Set_the_multimedia_on_the_web	Eq..	61.21%	T
13	Evaluate_algorithms_by_their_efficiency_correctness_and_clarity	Analyze_algorithms_using_complexity_efficiency_aesthetics_and_correctne...	Eq..	69.93%	⊥
14	Evaluate_computer_components_in_terms_of_features_and_price	Understand_machine_level_components_and_related_issues_of_complex...	Eq..	65.63%	⊥
15	Express_the_design_of_a_Web_site_using_standard_tools	Use_web_design_tools	Eq..	72.73%	T
16	List_ways_to_increase_computer_performance	Propose_options_to_improve_computer_performance	Eq..	67.85%	T
17	Name_and_explain_the_steps_in_the_problem-solving_process	List_problem_solving_phases	Eq..	72.91%	T
18	Name_the_different_phases_of_the_software_development_process	Use_one_or_more_software_development_models	Eq..	69.61%	T
19	Use_modeling_and_simulation_to_represent_and_understand_natural_phenomena	Use_Modeling_and_simulation_to_solve_real_world_problems	Eq..	74.53%	T
20	Utilize_advanced_OS_user_interface_elements_and_features	Use_interactive_graphic_OS	Eq..	62.91%	T
21	Write_an_algorithm_to_solve_an_assigned_problem_using_a_specified_set_of_commands	Design_algorithms_and_programming_solutions_to_problems_in_differen...	Eq..	72.58%	T
22	Write_conditional_statements_that_include_simple_and_complex_Boolean_expressions_to...	Create_complex_logical_expressions_using_Boolean_operators_and_fun...	Eq..	75.93%	T

Slika 30. Deo uparenih veština srednjoškolskog i referentnog nastavničkog kurikuluma

Kolona “*Bloom*” u tabeli sa slike 30 sadrži τ oznaku ako je nivo veštine u nastavničkom kurikuluma viši ili jednak od zahtevanog nivoa u srednjoškolskom kurikulumu, odnosno oznaku \perp ako to nije slučaj.

Tako, na primer, za par veština $\{Evaluate_algorithms_by_their_efficiency_correctness_and_clarity, Analyze_algorithms_using_complexity_efficiency_aesthetics_and_correctness\}$ (slika 30, red 13) vrednost je \perp , jer srednjoškolska veština pripada višem nivou Blumove taksonomije (‘Evaluate’) od odgovarajuće nastavničke veštine (‘Analyze’).

Tabela sa slike 30 sadrži rezultate koji su potencijalno netačni. To su parovi klasa prikazani u redovima 3, 7, 11 i 14. Primeri klasa koje nisu uparene zato što ne postoji odgovarajuća klasa u referentnom nastavničkom kurikulumu su:

- *Explain_the_relationship_between_a_web_server_a_web_page_and_a_browser,*
- *Describe_the_difference_in_the_processing_of_arrays_stacks_and_queues.*

Uočljivo je da su neki ishodi u srednjoškolskom kurikulumu reprezentovani većim brojem potklasa klase *Skills* nego odgovarajući ishodi referentnog nastavničkog kurikuluma. Stoga, neke veštine srednjoškolskog kurikuluma ostaju neuparene iako referentni nastavnički kurikulum sadrži klase koje (implicitno) uključuju ove veštine. Primeri klasa srednjoškolskog kurikuluma koje iz navedenih razloga nisu uparene su:

- *Code_a_program_to_solve_a_stated_problem_using_variables_and_at_least_one_decision_or_loop,*
- *Use_advanced_search_engine_options_and_refine_searches_to_locate_information.*

5.4.5. Primena 1:N algoritma

Primer primene 1:N algoritma je uparenost potklasa klase *Semantic_Web_and_knowledge_representation* i klase *Knowledge-based_Systems*. Budući da klasa *Semantic_Web_and_knowledge_representation* sadrži neuparene „list“ potklase, a da je *Knowledge-*

based_System „list“ klasa uparena sa *Semantic_Web_and_knowledge_representation* (slika 28, red 21), sistem će sugerisati 1:N relaciju, odnosno da je klasa *Knowledge-based_Systems* superklasa svim potklasama klase *Semantic_Web_and_knowledge_representation* kao što su: *Ontology*, *Predicate_logic*, *Web_ontology_language*, itd. (slika 31).

Na kraju primene sistema ukupan procenat uparenih *Knowledge* potklasa je 87%.

...	Source class	Target class	Type of r...	Similar...
1	Knowledge-based_Systems	Knowledge_representation_in_education	Superclass	80.16%
2	Knowledge-based_Systems	Ontology	Superclass	80.16%
3	Knowledge-based_Systems	Predicate_logic	Superclass	80.16%
4	Knowledge-based_Systems	Propositional_logic	Superclass	80.16%
5	Knowledge-based_Systems	Resource_Description_Framework_-_RDF	Superclass	80.16%
6	Knowledge-based_Systems	Semantic_web_-_basic_notions	Superclass	80.16%
7	Knowledge-based_Systems	Web_ontology_language	Superclass	80.16%

Slika 31. Uparene klase u „Superclass“ relaciji

5.4.6. Analiza rezultata

Primenom opisanog sistema uparen je visok procenat *Knowledge* potklasa (preko 85%) i *Skills* potklasa (80%) modela srednjoškolskog informatičkog kurikuluma uz preciznost od 0.64, odziv od 0.76, i F-meru od 0.695. Rezultati evaluacije sistema kojom je dobijena veća vrednost odziva od vrednosti preciznosti (uz zadovoljavajuće visoku vrednost preciznosti) su saglasni sa referencom (Stoilos, 2005) gde se „najveća važnost“ daje upravo vrednosti odziva kada je ontološko uparivanje poluautomatski proces. U (Stoilos, 2005) se navodi da je „napor“ koji je potrebno uložiti za brisanje identifikovanih pogrešnih parova minimalan u poređenju sa „ukrštanjem“ heterogenih ontologija (koje mogu imati veliki broj koncepata i atributa) i identifikovanjem odgovarajućih parova, te da je stoga mera odziva „mnogo važnija“. Iz dobijenih kvantitativnih rezultata o procentu uparenosti klasa i evaluaciji sistema može se zaključiti da je model referentnog nastavničkog kurikuluma zadovoljavajuće usaglašen sa modelom zasnovanim na ACM K12 standardu. Ipak analiza prikazana u prethodnim sekcijama ukazuje na to da neka znanja ACM K12 modela nisu mapirana u model referentnog nastavničkog kurikuluma. Tako, važne teme koje se odnose na faze mašinskog ciklusa, tehnike dokumentovanja (softvera), robotike, web dizajn prilagođen korisniku nisu mapirane u model referentnog nastavničkog kurikuluma. Neuparene su ostale i klase koje reprezentuju principe pretraživanja Interneta (unošenja parametara pretraživanja) i principe ocenjivanja web sajta. Iako su, na primer, principi kreiranja web sajta mapirani u referentni nastavnički kurikulum, on ne reprezentuje posebno i klasu koja predstavlja principe ocenjivanja ove oblasti. Stoga je potrebno model referentnog nastavničkog kurikuluma unaprediti tako da reprezentuje i ova nedostajuća znanja. Ipak, neke neuparenosti, kao što je opisano u sekciji 5.4.3., ne znače da referentni nastavnički kurikulum ne sadrži odgovarajuća znanja, već one proizilaze iz različite klasne strukture poređenih klasa ili specifičnosti definisanih

pravila u korišćenim algoritmima. U prvom slučaju klase koje reprezentuju specifične primene računara su ostale neuparene. U drugom slučaju ostale su neuparene pojedine klase u koje su mapirana znanja oblasti „veze računarskih nauka i matematike“. Slično, sistem je ukazao da tema robotike nije obuhvaćena nastavničkim kurikulumima, iako model referentnog nastavničkog kurikuluma sadrži klasu koja reprezentuje primenu robotike u realizovanju nastave programiranja. Međutim, metodika programiranja ne pripada informatičkom domenu, te njene potklase nisu kandidati za uparivanje. Ove neuparenosti bi se mogle rešiti proširivanjem prostora pretraživanja primenjenih algoritama. Ipak, to je potrebno detaljno razmotriti jer bi se tako, potencijalno, dobio i veći broj netačnih parova. Na primer, neuparena klasa srednjoškolskog kurikuluma koja reprezentuje principe softverskog dokumentovanja bi tako mogla biti uparena sa klasom dokumentovanja koja se odnosi na dokumentovanje kao podršku nastavnoj praksi.

Preciznost i, posebno, odziv sistema su očekivano niži (0.59 i 0.52, respektivno) pri uparivanju potklasa klasa *Skills* s obzirom na nisku struktuiranost i nazive/labela klasa koji sadrže, uglavnom, slobodan tekst. Ipak, na osnovu dobijenih rezultata se može zaključiti da model referentnog nastavničkog kurikuluma u zadovoljavajućem procentu podrazumeva ciljeve učenja definisane u ACM K12 modelu. Takođe, *Skills* potklase modela referentnog nastavničkog kurikuluma su većinom na višem ili istom nivou kognitivne dimenzije Blumove taksonomije u odnosu na odgovarajuće klase modela srednjoškolskog kurikuluma. Sistem je ukazao da je pojedine neuparene ishode srednjoškolskog kurikuluma, poput „objašnjenje razlike između web strane, servera i Internet pretraživača“, potrebno mapirati u model referentnog nastavničkog kurikuluma i povezati sa odgovarajućom *Knowledge* potklasom. Ipak, uočljivo je da su neke *Skills* potklase srednjoškolskog modela ostale neuparene iako u modelu referentnog nastavničkog kurikuluma postoje klase koje bi se mogle definisati kao njene superklase. Ovaj tip neuparenosti bi mogao biti razrešen na dva načina. Jedan bi bio redefinisavanje algoritma (pre svih 1:N) tako da obuhvata i ove slučajeve. Drugi pristup bi podrazumevao unapređenje referentnog nastavničkog kurikuluma tako da podrazumeva detaljnost opisa veština odgovarajući detaljnosti opisa ciljeva učenja opisanih u temama ACM K12 modela.

5.5. Evaluacija sistema za usaglašavanje odabranog nastavničkog kurikuluma iz Republike Srbije i srednjoškolskog informatičkog kurikuluma

U ovoj sekciji upoređene su ontološke reprezentacije srednjoškolskog informatičkog kurikuluma zasnovanog na ACM K12 modelu (sekcija 4.3.5.) i ontološke reprezentacije kurikuluma integrisanih akademskih studija „Tehnika i informatika“ Fakulteta tehničkih nauka u Čačku (sekcija 4.3.7.).

5.5.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa

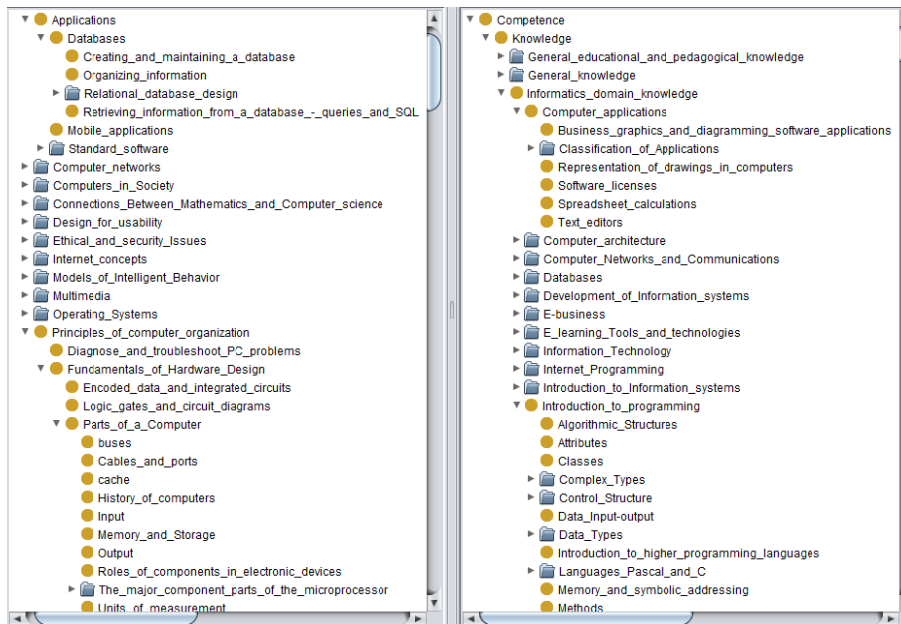
Na slici 32 prikazan je deo uparenih roditeljskih klasa posle primene prvog taksonomijskog algoritma. Klase smeštene u kolonu "Source class" pripadaju srednjoškolskom modelu kurikuluma, dok klase u "Target class" koloni pripadaju modelu nastavnčkog kurikuluma. Prag je, eksperimentalno, postavljen na 70%.

...	Source class	Target class	Ty...	Similar...
1	Algorithms_and_data_structures	Data_types	Eq...	87.91%
2	Applications	Computer_applications	Eq...	79.2%
3	Computer_networks	Computer_Networks_and_Comm...	Eq...	87.87%
4	Connections_Between_Mathematics_and_Co...	Mathematics_of_Informatics	Eq...	84.14%
5	Data_Structures	Complex_Types	Eq...	72.96%
6	Databases	Databases	Eq...	90.08%
7	Fundamentals_of_Hardware_Design	Memory_Systems_and_Technolo...	Eq...	85.19%
8	Internet_elements	Internet	Eq...	79.44%
9	Multimedia	Multimedia_systems	Eq...	79.01%
10	Object-oriented_programming	Object-Oriented_Programming	Eq...	94.22%
11	Operating_Systems	Operating_systems	Eq...	88.32%
12	Parts_of_a_Computer	Practicum_in_computer_architect...	Eq...	75.64%
13	Principles_of_computer_organization	Computer_architecture	Eq...	82.05%
14	Programming_Languages	Programming_Languages	Eq...	85.15%
15	Programming_and_software_engineering	Introduction_to_programming	Eq...	73.55%
16	Relational_database_design	Relational_databases	Eq...	82.23%
17	Representing_Information_Digitally	Information_Technology	Eq...	70.99%
18	Standard_software	Classification_of_Applications	Eq...	75.21%
19	Structured_programming	Control_Structures	Eq...	82.22%
20	The_major_component_parts_of_the_micropr...	Languages_Pascal_and_C	Eq...	72.12%

Slika 32. Deo uparenih roditeljskih klasa

I ovde se može zapaziti da sličnost parova klasa istog ili vrlo sličnog naziva nije nužno bliska ili jednaka 100%. Primeri za to su parovi klasa prikazani na slici 32 u redovima 2, 6, 9, 11, 14. Razlog za to je učešće sličnosti potklasa i superklasa, ukoliko postoje, u ukupnoj sličnosti poređenih roditeljskih klasa. Tako je za par klasa {*Applications*, *Computer_applications*} (red 2, slika 32) dobijena sličnost ispod 80% zbog razlika u strukturi i nazivu potklasa (slika 33); baze podataka su, na primer, u nastavnčkom kurikulumu predstavljene posebnim kursom, te nisu modelovane kao potklasa klase *Computer_applications*.

S druge strane, klase *Fundamentals_of_Hardware_Design* i *Memory_Systems_and_Technologies*, iako različitog značenja naziva, su ispravno uparene (slika 32, red 7). Uvidom u hijerarhijsku strukturu poređenih ontologija može se uočiti da klasa *Fundamentals_of_Hardware_Design* sadrži potklase koje odgovaraju potklasama klase *Memory_Systems_and_Technologies*. Analogan zaključak se može izvesti i za par klasa prikazan u redu 12 slike 32.



Slika 33. Deo hijerarhijske strukture srednjoškolskog i nastavničkog modela

Procenat uparenih *Knowledge* potklasa srednjoškolskog kurikuluma posle primene algoritma za izračunavanje sličnosti roditeljskih klasa je 8.98%. Udeo roditeljskih potklasa klase *Knowledge* u ukupnom broju *Knowledge* potklasa u srednjoškolskom modelu kurikuluma je “samo” 16.11%, te u ovoj fazi uparivanja nije moguće dobiti procenat uparenosti viši od navedene vrednosti. Ipak, uočljivo je da je dobijen značajno niži procenat uparenosti u odnosu na prethodnu kombinaciju ulaznih ontoloških modela (sekcija 5.4.1). To bi se moglo objasniti različitim pristupom primenjenim pri definisanju taksonomijskih struktura poređenih ontoloških modela kurikuluma (struktuiranost u modelu nastavničkog kurikuluma je niža). Ipak, niži procenat uparenosti ukazuje prvenstveno na to da izabrani nastavnički kurikulum ne predviđa izučavanje tematskih oblasti odgovarajućih neuparenim roditeljskim klasama srednjoškolskog modela.

Pri tome postoje dva osnovna slučaja neuparenosti roditeljskih klasa: postojanje neuparenih klasa čiji su roditelji upareni sa odgovarajućom klasom nastavničkog kurikuluma i neuparenost klasa čiji roditelji nisu upareni (ili ne postoje). U prvom slučaju dobijena neuparenost roditeljske klase neće uticati na to da njene potklase ostanu neuparene. S druge strane, neuparenost roditeljskih klasa čiji ni roditelji nisu upareni (ili ne postoje) može uzrokovati neuparenost njenih potklasa nakon primene svih algoritama.

Primeri neuparenih roditeljskih klasa srednjoškolskog kurikuluma čiji su roditelji upareni su:

- *The machine cycle*
(roditelji ove klase su prikazani u redovima 7 i 13 i koloni “Source class” na slici 32),
- *Levels_of_Language_Software_and_Translation*

(potklasa klase *Programming_and_software_engineering*, red 15, slika 32).

Neuparene roditeljske klase čiji su roditelji takođe neupareni ili ne postoje su:

- *Computers_in_Society*,
- *Careers_in_Computing*,
- *Ethical_and_security_Issues*,
- *Ethical_and_Social_Issues*,
- *Privacy_and_Security*,
- *Models_of_Intelligent_Behavior*,
- *Design_for_usability*,
- *Human_Computer_Interaction*.

5.5.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja

Slika 34 prikazuje deo uparenih „list“ klasa čiji su roditelji upareni prethodnim algoritmom za izračunavanje sličnosti roditeljskih klasa.

...	Source class	Target class	Typ...	Similar...
1	Classes	Classes	Equ...	100.0%
2	Client-server_architecture	The_Client-server_model	Equ...	76.88%
3	Conditionals_and_Boolean_logic	Boolean_Expressions	Equ...	70.56%
4	Constants_and_variables_of_simple_types	Variable_and_value_arguments	Equ...	89.95%
5	Control_unit	Control_Units	Equ...	100.0%
6	File_systems_and_organization	File_management	Equ...	96.25%
7	Inheritance	Inheritance	Equ...	100.0%
8	Method_-_functions_and_parameters	Methods	Equ...	100.0%
9	Multi-tasking	Task_management	Equ...	74.5%
10	Network_topologies	Network_topology	Equ...	100.0%
11	Objects	Object	Equ...	100.0%
12	Programming_style	Semantics_of_programming_languag...	Equ...	79.46%
13	Relationship_between_Boolean_Algebra_...	Boolean_Algebra	Equ...	100.0%
14	Selection	Selection	Equ...	100.0%
15	Software_components	Ways_of_addressing	Equ...	70.41%
16	Software_to_support_multimedia	Work_in_leading_software_packages	Equ...	70.6%
17	Spreadsheet_as_a_table	Spreadsheet_calculations	Equ...	72.35%
18	Subprograms	Program_Modularization	Equ...	100.0%
19	Using_the_clipboard	Applications_support	Equ...	73.93%
20	Word_processing	Text_editors	Equ...	100.0%
21	Cache	Cache	Equ...	100.0%

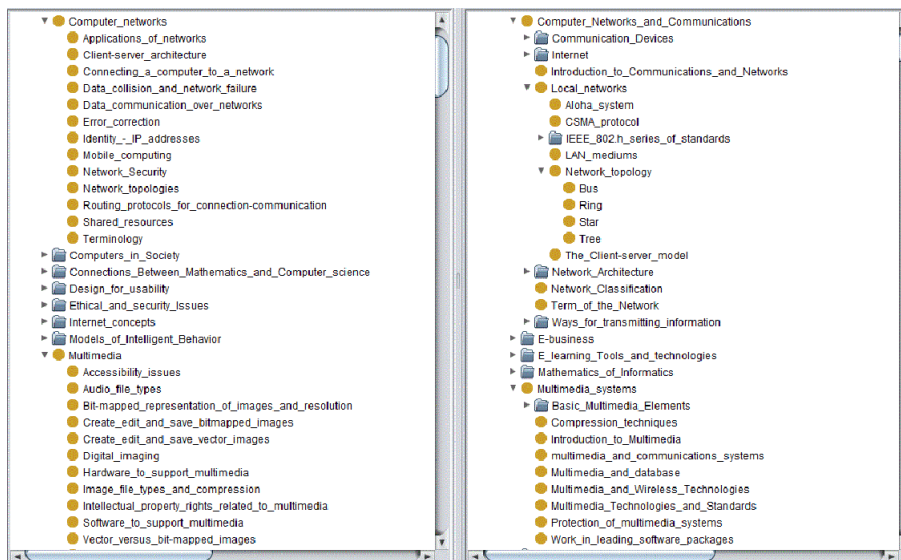
Slika 34. Parovi „list“ klasa uparenih roditelja

Za razliku od slike 32, uočljivo je da slika 34 sadrži uparene klase sa 100% sličnosti. Ovo je i očekivano s obzirom da na izračunavanje sličnosti ne utiče sličnost superklasa i potklasa kao u prethodnom algoritmu, već samo terminološka sličnost poređenih klasa (uz uslov roditeljske sličnosti). Stoga, ako klase sadrže identične termine (dobijene nakon primenjene normalizacije) i ukoliko su roditelji klasa upareni primenom algoritma za izračunavanje sličnosti roditeljskih klasa, tada je sličnost klasa 100%, poput parova klasa prikazanih u redovima 1, 5, 7, 11, 14 i 21 (slika 34). Par klasa prikazan u redu 18 ima sličnost od 100% jer je klasa *Program_modularization*

nastavničkog modela dodatno opisana labelom *Functions*, koja u WordNet bazi predstavlja sinonim pojmu *Subprograms*.

Slika 34 sadrži i parove klasa koji se mogu tumačiti kao netačni; to su parovi klasa prikazani u redovima 15 i 19. Budući da za, na primer, izvornu klasu *Using_the_clipboard* (red 19), koja pripada hijerarhijskoj strukturi klase *Operating_Systems*, nije pronađen ekvivalentan par u odgovarajućem delu ontološke strukture modela nastavničkog kurikuluma (struktura klase *Operating_systems*), sistem je pronašao „prvu“ terminološki najbliskiju klasu u odgovarajućem delu ciljne ontologije. Vrednost sličnosti izabranog para klasa je iznad zadatog praga, te su one (netačno) postale „najbolje uparene klase“. Stoga je potrebno obratiti posebnu pažnju na parove klasa čija je vrednost sličnosti bliskija vrednosti praga i, ukoliko je potrebno, manuelno ispraviti rezultate.

S druge strane, iz naziva klase nastavničkog kurikuluma *Work_in_leading_software_packages* može se zaključiti da ona odgovara radu sa softverskim aplikacijama uopšteno, te bi se, uz dobijenu nešto nižu vrednost sličnosti, rezultat algoritma prikazan u redu 16 slike 34 mogao tumačiti kao potencijalno netačan. Ipak, sa slike 35 se vidi da je klasa *Software_to_support_Multimedia* potklasa klase *Multimedia*, dok klasa *Work_in_leading_software_packages* pripada ekvivalentnoj ontološkoj strukturi – ona je potklasa klase *Multimedia_systems*. Roditeljske klase *Multimedia* i *Multimedia_systems* su uparene prethodnim uparivačem (red 9, slika 32), te se par (red 16, slika 34) može tumačiti kao ispravan rezultat primene algoritma - klasa *Work_in_leading_software_packages* zaista reprezentuje rad sa vodećim softverskim aplikacijama iz oblasti multimedije.



Slika 35. Deo ontološke strukture poređenih modela

Primer uparenih „list“ klase i roditeljske klase čije su sve potklase „list“ prikazan je u redu 10. Sa slike 35 se vidi da je „list“ klasa *Network_topologies* srednjoškolskog modela, potklasa klase *Computer_networks*. S druge strane, u ontološkom modelu nastavničkog kurikuluma postoji klasa *Network_topology* (potklasa

klase *Computer_Networks_and_Communications*) koja nije „list“ klasa, ali ima samo „list“ potklase. Budući da su im roditeljske klase uparene (red 3, slika 32), klase *Network_topologies* i *Network_topology* su, takođe, uparene.

Procenat uparenih *Knowledge* potklasa srednjoškolskog kurikuluma u ovoj fazi je 38.67%.

5.5.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja

Budući da nastavnički kurikulum (Fakultet tehničkih nauka, 2013) ne sadrži informacije o tome koji su kursevi disjunktne, u ontološki model nisu mapirane grupe potencijalno disjunktne kurseva kao „disjoint“ klase. Stoga algoritam prikazan u sekciji 5.2.3.3. nije bilo moguće primeniti na ovako definisane ontološke modele srednjoškolskog i nastavničkog kurikuluma. Ipak, s obzirom na mogućnost da pojedine klase jesu „slične“ iako im roditelji nisu upareni, neophodno je proširiti prostor pretraživanja radi pronalazjenja adekvatnih parova. Sličnost klasa C_{i1} i C_{j2} , $s_{leaf2}(C_{i1}, C_{j2})$ se utvrđuje u ovoj fazi na sledeći način.

```

/* Neka je  $A_{ij}$  klasa ontologije, pri čemu važi da  $A_{i1} \in O_1$  i
 $A_{j2} \in O_2$ .
/* Neka je  $A_{parent}$  lista parova klasa uparenih primenom
algoritma za izračunavanje sličnosti roditeljskih klasa.
/* Neka su  $T_i$  vrednosti zadatog praga  $|T_i \in [0,1], i \in \{1,2\}$ 
/* Neka, dalje, važi sledeće:  $C_{i1}$  je „list“ klasa ontologije  $O_1$ 
i  $C_{j2}$  je „list“ klasa ontologije  $O_2$ , ili  $C_{i1}$  je „list“ klasa
ontologije  $O_1$  i klasa  $C_{j2}$  ima samo „list“ potklase, ili  $C_{j2}$  je
„list“ klasa ontologije  $O_2$  i  $C_{i1}$  ima samo „list“ potklase.
  If  $\exists \{A_{l1}, A_{k2}\} \{A_{l1}, A_{k2}\} \in A_{parent}$  and
 $s_{term}(C_{i1}, C_{j2}) > T_1$  and  $\exists \{A_{o1}, A_{p2}\} | s_{parent}(A_{o1}, A_{p2}) > T_2$ ,  $A_{o1} \in \{A_{11},$ 
 $A_{21} \dots A_{n1}\}$ ,  $A_{l1} \in \{A_{11}, A_{21} \dots A_{n1}\}$ ,  $C_{i1} \subseteq \{A_{11}, A_{21} \dots A_{n1}\}$ ,  $A_{p2} \in$ 
 $\{A_{12}, A_{22} \dots A_{m2}\}$ ,  $A_{k2} \in \{A_{12}, A_{22} \dots A_{m2}\}$ ,  $C_{j2} \subseteq \{A_{12}, A_{22} \dots A_{m2}\}$  then
 $s_{leaf2}(C_{i1}, C_{j2}) = s_{term}(C_{i1}, C_{j2})$ 
else
 $s_{leaf2}(C_{i1}, C_{j2}) = s_{leaf}(C_{i1}, C_{j2})$ 

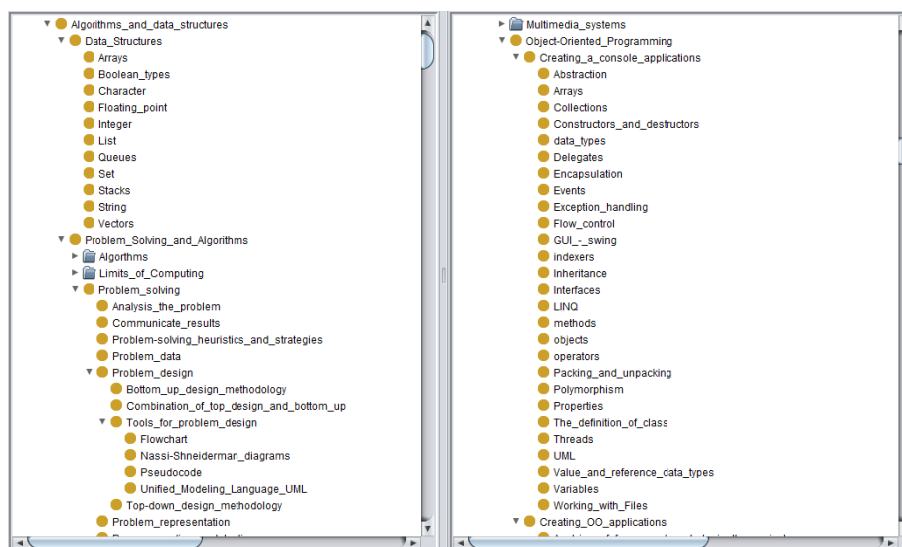
```

Dakle, u ovom koraku posmatraju se „list“ klase (čija je sličnost veća od praga T_1) čiji roditelji nisu upareni ali postoji par roditelja koji su „dovoljno slični“. Preciznije, postoji par roditelja čija je sličnost, određena uparivačem za utvrđivanje sličnosti roditeljskih klasa, veća od zadatog praga T_2 . Vrednosti T_1 i T_2 su eksperimentalno postavljene na 0.88 (88%) i 0.55 (55%), respektivno. Ovde se takođe uzimaju u obzir parovi klasa od kojih je jedna „list“ klasa a druga roditelj samo „list“ klasama. Deo uparenih „list“ klasa dobijenih nakon primene opisanog algoritma prikazan je na slici 36.

R...	Source class	Target class	Typ...	Simila...
1	Arrays	Arrays	Equ...	100.0%
2	Documentation_techniques	Development_and_system_documentation	Equ...	89.94%
3	Hypertext_Language_HTML_tags	HTML	Equ...	98.07%
4	Unified_Modeling_Language_UML	UML	Equ...	100.0%

Slika 36. Uparene „list“ klase neuparenih roditelja

Karakteristični primeri koji opravdavaju uvođenje opisanog algoritma za proširivanje “prostora pretraživanja” prikazani su u redovima 1 i 4. Tako, iako obe klase u redu 4 predstavljaju ekvivalentnu temu, one nisu uparene prethodnim algoritmom jer nijedan od roditelja klase *Unified Modeling Language UML* modela srednjoškolskog kurikulumuma nije uparen ni sa jednim roditeljem klase *UML* modela nastavničkog kurikulumuma; sličnost poređenih klasa primenom prethodnog algoritma (kojim se utvrđuje sličnost „list“ klasa uparenih roditelja) je postavljena na nulu. Objašnjenje neuparenosti njihovih roditeljskih klasa dato je u nastavku. Pojam UML-a, u nastavničkom kurikulumu, predstavlja sadržaj kursa ‘Objektno orijentisano programiranje’. Kurs ‘Objektno orijentisano programiranje’ je mapiran u klasu *Object-Oriented Programming*. Primenom algoritma za utvrđivanje sličnosti roditeljskih klasa, klasa *Object-Oriented Programming* uparena je sa klasom *Object-oriented programming* srednjoškolskog modela (red 10, slika 32). U skladu sa principom kreiranja modela nastavničkog kurikulumuma opisanim u sekciji 4.3.7, klasa UML je potklasa i klase *Creating_a_console_applications* (slika 37) koja nije uparena ni sa jednom roditeljskom klasom srednjoškolskog modela. S druge strane, u modelu srednjoškolskog kurikulumuma, objedinjeni jezik modeliranja (UML) je mapiran na potklasu sledećih roditeljskih klasa: *Tools_for_problem_design*, *Problem_design*, *Problem_solving*, *Problem Solving and Algorithms*, *Algorithms_and_data_structures* (slika 37). Ipak, proširivanjem prostora pretraživanja i na klase neuparenih roditelja koje pripadaju “sličnim” klasnim strukturama, par klasa {*Unified Modeling Language UML*, *UML*} je ispravno uparen (slika 37).



Slika 37. Hijerarhijske strukture klasa “neuparenih roditelja”

Red 3 predstavlja primer uparivanja roditeljske klase modela srednjoškolskog kurikuluma čije su sve potklase „list“ klase i „list“ klase nastavničkog kurikuluma. Takođe, i u ovoj fazi, dobijeni su potencijalno netačni parovi klasa (poput para prikazanog u redu 2).

Procenat uparenih *Knowledge* potklasa nakon ove faze je 58.2%. Može se zapaziti da je procenat uparenosti *Knowledge* potklasa modela srednjoškolskog kurikuluma niži pri poređenju sa modelom izabranog (pojedinačnog) nastavničkog kurikuluma od procenta dobijenog pri uparivanju sa referentnim modelom nastavničkog kurikuluma. To se može smatrati očekivanim rezultatom sistema budući da je model referentnog nastavničkog kurikuluma nastao na osnovu analize preporuka međunarodnih akreditacionih tela i organizacija i analize više od 20 svetskih i domaćih kurikuluma.

5.5.4. Primena algoritma za izračunavanje relacione sličnosti

Neki od parova *Skills* potklasa dobijenih primenom algoritma za utvrđivanje relacione sličnosti prikazani su na slici 38. Prag je eksperimentalno postavljen na 55%.

...	Source class	Target class	Ty...	Simila...	BL...
1	Based_upon_the_file_extension_determine_if_a_given_file_type_is_audi...	Create_and_process_various_multimedia_content_in_leading_softwar...	Eq...	65.02%	T
2	Compare_FCS_with_other_electronic_devices	Configure_macro_architecture_of_a_computer_system	Eq...	58.73%	↓
3	Connect_a_computer_to_a_network	Diagnose_and_fix_faults_in_the_network	Eq...	68.61%	T
4	Create_a_Web_site_given_design_specifications	Web_design_for_standard_and_mobile_platform	Eq...	68.43%	T
5	Create_user-friendly_and_functional_Web_sites_and_programs_that_ap...	Develop_web_applications_in_Java_web_technology_with_the_help...	Eq...	64.96%	T
6	Describe_how_various_types_of_data_are_stored_in_a_computer_syst...	Use_pointers_and_functions	Eq...	67.73%	T
7	Describe_the_principal_components_of_computer_organization	Identify_different_types_of_buses_in_a_computer_system	Eq...	75.1%	T
8	Describe_the_role_of_the_OS_as_an_intermediary_between_application...	Identify_strengths_and_weaknesses_and_contrasts_of_popular_oper...	Eq...	66.72%	T
9	Design_a_multi-table_relational_database	Database_design	Eq...	57.14%	T
10	Evaluate_computer_components_in_terms_of_features_and_price	Evaluate_the_elements_of_computer_architecture	Eq...	66.55%	T
11	Illustrate_the_is-a_and_has-a_object-oriented_concepts	Recognize_the_concept_of_object-oriented_aproach	Eq...	73.8%	↓
12	List_ways_to_increase_computer_performance	Recognize_the_performance_and_classes_of_storage_device	Eq...	65.73%	T
13	Select_appropriate_data_types	Recognize_the_basic_types_and_data_structures	Eq...	72.28%	↓
14	State_the_hardware_requirements_for_adding_a_computer_to_a_network	Recognize_the_principles_of_computer_network_organization_and_d...	Eq...	70.39%	T
15	Use_at_least_two_Internet_elements	Install_and_launch_a_variety_of_Internet_services	Eq...	69.86%	T
16	Use_integrated_software_productively	Use_computer_applications	Eq...	72.15%	T
17	Use_modeling_and_simulation_to_represent_and_understand_natural_ph...	Understand_the_basics_of_modeling_data_dictionary_and_the_role_a...	Eq...	67.47%	↓
18	Utilize_advanced_OS_user_interface_elements_and_features	Compare_the_main_characteristics_and_management_functions_of_o...	Eq...	63.78%	T
19	Write_a_computer_program_that_implements_an_algorithm	Apply_the_logic_of_object-oriented_programming	Eq...	72.86%	↓
20	Write_structured_program_code	Work_with_arrays_and_structures	Eq...	63.36%	↓

Slika 38. Primeri uparenih *Skills* potklasa modela srednjoškolskog kurikuluma i odabranog nastavničkog kurikuluma

Sa slike 38 se može zapaziti da i ova faza uparivanja daje parove klasa koji se mogu tumačiti kao neodgovarajući. To se odnosi, pre svih, na parove klasa prikazane u redovima 2, 3, 12 i 17. Klase *Connect a computer to a network* i *Diagnose and fix faults in the network* (red 3, slika 38) su povezane sa *Knowledge* potklasama *Computer networks* i *Computer Networks and Communications*, respektivno, uparenih algoritmom za utvrđivanje sličnosti roditeljskih klasa (red 3, slika 32). Stoga se u ovoj fazi za izvornu klasu *Connect a computer to a network* odgovarajuća klasa dobija pretraživanjem skupa *Skills* potklasa sa kojim je klasa *Computer Networks and Communications* modela nastavničkog kurikuluma povezana. Kao terminološki najbližnja klasa iz tog skupa dobijena je klasa *Diagnose and fix faults in the network*. Budući da je vrednost terminološke sličnosti ovih klasa iznad zadatog praga, par klasa je uparen. Ipak, značenje veština/ishoda na koje se klase odnose ukazuje na to da je potrebna manuelna intervencija korisnika nad ovim rezultatom. Takođe, sa slike 38 može se zaključiti da je moguće

razmotriti tačnost uparivanja i parova klasa prikazanih u redovima 1, 7, 8 i 14.

Primeri parova klasa u kojima klasa srednjoškolskog modela reprezentuje ishod višeg nivoa Blumove taksonomije od nivoa kojem pripada odgovarajuća veština (klasa) nastavničkog kurikuluma prikazani su u redovima 11, 13, 19 i 20. Tako za par klasa prikazan u redu 11 važi da je izvorna klasa (modela srednjoškolskog kurikuluma) potklasa *Apply* klase, jer odgovara veštini *primene* konceptata objektno-orijentisane paradigme, dok se klasa modela nastavničkog kurikuluma odnosi na poznavanje konceptata objektno-orijentisanog pristupa, te predstavlja potklasu klase *Remember-understand*. Slično, selektovanje pripada kategoriji višeg nivoa Blumove taksonomije od poznavanja (red 13, slika 38).

Procenat uparenih *Skills* potklasa je 51.66%. I ovde je, kao za *Knowledge* potklase, dobijena niža vrednost uparenosti od vrednosti dobijene pri upoređivanju sa modelom nastavničkog kurikuluma nastalog na osnovu analize međunarodnih preporuka i brojnih kurikuluma.

Nepostojanje odgovarajućih klasa za neuparene *Skills* potklase srednjoškolskog kurikuluma je često posledica toga što za *Knowledge* klase, sa kojima su one povezane preko objektnog svojstva *hasKnowledge*, ne postoje odgovarajuće klasne strukture u modelu nastavničkog kurikuluma. Tako su, na primer, dobijene neuparene *Skills* potklase koje su povezane sa neuparenom klasom *Models_of_Intelligent_Behavior*, poput klasa:

- *Describe_what_distinguishes_humans_from_machines_focusing_on_human_intelligence_versus_machine_intelligence_and_ways_we_can_communicate,*
- *Describe_the_major_applications_of_artificial_intelligence_and_robotics* itd.,

kao i klase povezane sa neuparenom klasom *Computers_in_Society*:

- *Name_and_describe_the_contributions_of_two_or_more_computer_scientists,*
- *Behave_ethically_when_using_the_computer,*

zatim klase povezane sa neuparenom klasom *Design_for_usability*:

- *Create_a_user-centered_design,*
- *Evaluate_the_usability_of_a_user_interface,*

kao i klase povezane sa neuparenom klasom *Ethical_and_Social_Issues*, poput:

- *Distinguish_between_ethical_and_legal_issues,*
- *Define_intellectual_property_and_state_the_impact_of_provisions_to_protect_it,*
- *Define_software_piracy_and_discuss_its_effect_on_software_company_profits_and_the_price_of_software_to_the_consumer,*
- *Identify_at_least_two_benefits_and_two_drawbacks_of_using_commercial_public_domain_open_source_and_shareware.*

Takođe, za neke *Skills* potklase ne postoje odgovarajuće klase u modelu nastavničkog kurikuluma, iako su one povezane sa *Knowledge* potklasama koje su adekvatno uparene sa ekvivalentnim *Knowledge* potklasama nastavničke ontologije. To su, na primer:

- *Evaluate algorithms by their efficiency correctness and clarity,*
- *Determine if a given algorithm successfully solves a stated problem,*
- *Explain the differences advantages and disadvantages between vector and bitmapped images,*
- *List activities in which humans excel over computers and activities in which computers excel over humans, itd.*

Uočljivo je i da klase koje se odnose na veštinu programiranja, pisanja programskog koda i slično ili nisu uparene (poput klase: *Test and execute a program that corresponds to a set of specifications, Code a program to solve a stated problem using variables and at least one decision or loop, Write a program to process a range or all elements in one and two-dimensional arrays*) ili su odgovarajući ishodi u nastavničkom kurikulumu formulisani tako da predstavljaju niži nivo Blumove taksonomije (red 20 slika 38). Primeri takvih ishoda nastavničkog kurikuluma (mapiranih na potklase klase *Apply* ili *Remember-understand*) su: „Zna za osnovne tipove i strukture podataka”, “razume i primenjuje rad sa nizovima i strukturama” (kursa ‘Uvod u programiranje’), „Poznaje koncept objektno orjentisanog pristupa, i koristi klase i objekte u programskom jeziku C++” (kurs ‘Programski jezici’), “u rešavanju zadataka i svakodnevnom životu primenjuje OOP logiku” (kurs ‘Objektno orijentisano programiranje’). Iz opisanih neusaglašenosti moguće je izvesti dva generalna zaključka koji se odnose na unapređenje kurikuluma i, eventualno, unapređenje modela. U kontekstu unapređenja ontološke reprezentacije moguće je razmotriti način reprezentacije pojedinih ishoda tako da, pri kreiranju modela, glagoli koji opisuju ishod ne predstavljaju ključni kriterijum za mapiranje u odgovarajuću potklasu Blumove taksonomije. U primeru ishoda “koristi klase i objekte u programskom jeziku C++”, potrebno je, dakle, razmotriti da li *korišćenje* klasa i objekata u određenom objektnom programskom jeziku suštinski predstavlja *programiranje* (koje pripada najvišem nivou Blumove taksonomije). Ukoliko bi se na taj način protumačilo značenje ishoda on bi tada bio mapiran na odgovarajuću potklasu klase *Create*. Drugi zaključak koji je moguće izvesti iz opisanih neusaglašenosti, a koji je, po autoru ovog rada, neophodno svakako primeniti, je unapređenje kurikuluma tako da sadrži ishode koji, u oblasti programiranja, odgovaraju najvišem nivou Blumove taksonomije.

5.5.5. Primena 1:N algoritma

Parovi klasa u kome jedna klasa predstavlja „list“ klasu, dok je druga roditelj samo „list“ klasama, predstavljaju kandidate za Superclass/Subclass relaciju. Klasa *Network_topologies* ne sadrži potklase i uparena je sa klasom *Network_topology* primenom algoritma opisanog u 5.2.3.2. (red 10, slika 34). Klasa *Network_topology* sadrži samo „list“ potklase (*Bus*, *Ring*, *Star*, *Tree*) koje su primenom 1:N algoritma postale i potklase klase *Network_topologies* (slika 39). Analogno, klasa *Hypertext_Language_HTML_tags* srednjoškolskog modela je uparena sa „list“ klasom *HTML* nastavnčkog modela (slika 36, red 3) i sadrži samo „list“ potklase. Stoga su sve potklase klase *Hypertext_Language_HTML_tags* postale potklase klase *HTML* (slika 39).

...	Source class	Target class	Type of re...	Similarit...
1	Hypertext_Language_HTML_tags	HTML	Equivalence	98.07%
2	form	HTML	Subclass	98.07%
3	graphics	HTML	Subclass	98.07%
4	hyperlinks	HTML	Subclass	98.07%
5	multimedia	HTML	Subclass	98.07%
6	tables	HTML	Subclass	98.07%
7	texts	HTML	Subclass	98.07%
8	Network_topologies	Network_topology	Equivalence	100.0%
9	Network_topologies	Bus	Superclass	100.0%
10	Network_topologies	Ring	Superclass	100.0%
11	Network_topologies	Star	Superclass	100.0%
12	Network_topologies	Tree	Superclass	100.0%

Slika 39. Primeri sva tri tipa relacija

Konačan procenat uparenih *Knowledge* potklasa je 62.99%.

„List“ klase za koje ne postoji odgovarajuće klase u modelu nastavnčkog kurikuluma su sledeće potklase neuparenih roditeljskih klasa poput:

- potklasa neuparene roditeljske klase *Limits_of_Computing*:
 - *Computationally_hard_problems*,
 - *Computationally_intensive_problems*,
 - *Computers_versus_humans*,
 - *Unsolvable_problem_for_the_computer*.
- *Problem_solving* potklase:
 - *Problem-solving_heuristics_and_strategies*,
 - *Bottom_up_design_methodology*,
 - *top-down_design_techniques*.
- potklase klase *Models_of_Intelligent_Behavior*:
 - *Knowledge-based_Systems*,
 - *Machine_learning*,
 - *What_is_Intelligence_itd*.
- potklase klase *Ethical_and_security_Issues*:
 - *Intellectual_property_copyright_and_fair_use*,

- *Software_licensing_agreement,*
- *How_viruses_are_spread,*
- *Types_of_malicious_software itd.*
- potklase klase *Human_Computer_Interaction:*
 - *Fundamental_HCI_concepts,*
 - *Interface_evaluation,*
 - *User-centered_design, itd.*

Navedeni rezultati upućuju na potrebu proširivanja nastavničkog kurikuluma znanjima koja su reprezentovana neuparenim klasama.

Takođe, rezultati uparivanja su ispravno ukazali da u modelu ne postoje ni klase koje neposredno odgovaraju znanjima o relevantnosti web izvora, principima kreiranja i ocenjivanja korisnički prilagođenih web sajtova i sl. (iako model nastavničkog kurikuluma sadrži kompleksne klasne strukture poput *E-bussines*, *Internet_programming* i *Web_Technologies* koje reprezentuju napredne koncepte web dizajna). I koncepti kompjuterske grafike koji se odnose na vektorsku i rastersku grafiku nisu sadržani u nastavničkom kurikulumu iako u modelu nastavničkog kurikuluma postoji više klasa koje reprezentuju grafičke koncepte (svrstane u različite klasne strukture, odnosno pripadaju različitim kursevima).

S druge strane, poput rezultata dobijenih u sekciji 5.4.3, i ovde niz klasa koje reprezentuju binarne brojeve, funkcije, skupove, grafove nisu pronađene u potklasama domenskih znanja modela nastavničkog kurikuluma (pre svih u klasi *Mathematics_of_Informatics*). Ipak, ove teme jesu sadržane u nastavničkom kurikulumu kao deo matematičkih kurseva (diskretne matematike). Slično, klasa *Robotics* je ostala neuparena, jer se u nastavničkom kurikulumu teme robotike izučavaju u okviru kurseva 'Uvod u tehničke sisteme' i 'Metodički praktikum iz robotike' koje su reprezentovane potklasama klasa *General_knowledge* i *Knowledge_of_Informatics_teaching_methods*. Ove klasne strukture se ne uzimaju u obzir pri poređenju sa srednjoškolskim modelom. Ovi rezultati ukazuju na potrebu razmatranja unapređenja algoritama za usaglašavanje ontologija pretraživanjem i klasa koje ne pripadaju informatičkim znanjima.

5.5.6. Analiza rezultata

Procenat uparenih *Knowledge* i *Skills* potklasa modela srednjoškolskog kurikuluma je značajno niži pri poređenju sa modelom izabranog nastavničkog kurikuluma nego pri poređenju sa referentnim modelom nastavničkog kurikuluma. To se može objasniti principom kreiranja modela referentnog nastavničkog kurikuluma opisanim u sekciji 4.3.6. Referentni model je nastao na osnovu analize većeg broj kurikuluma iz različitih zemalja. Stoga je i opravdano da referentni model sveobuhvatnije reprezentuje domen kompetencija nastavnika informatike od izabranog, pojedinačnog kurikuluma. Ipak, iz dobijene neuparenosti *Knowledge* potklasa (uz preciznost od 0.63, odziv od 0.58 i F-meru od 0.60) se može izvesti zaključak da

analizirani nastavnički kurikulum iz Republike Srbije ne predviđa izučavanje značajnih tema srednjoškolskog nivoa ACM K12 standarda. Da bi izabrani nastavnički kurikulum bio usaglašen sa ACM K12 modelom potrebno je da sadrži kurseve koji se bave ograničenjem računara, principima veštačke inteligencije, etičkim i bezbednosnim pitanjima, principima kreiranja korisniku prilagođenog interfejsa, kao i da u okviru novog kursa ili postojećih kurseva previđa dublje izučavanje principa rešavanja problema i algoritama. Deo neuparenih klasa je posledica različite taksonomijske strukture srednjoškolskog modela i modela analiziranog nastavničkog kurikuluma. To se posebno odnosi na teme robotike i veze između matematike i računarskih nauka, koje su u modelu izabranog nastavničkog kurikuluma mapirane na klase koje odgovaraju nedomenskim znanjima. Rezultati dobijeni primenom sistema za usaglašavanje ontologija su saglasni i sa analizom prikazanom u sekciji 2.7.1. gde je utvrđivanje usaglašenosti realizovano jednostavnim metodom prebrojavanja srodnih tema podeljenih po oblastima ACM K12 standarda i izračunavanjem procentualne zastupljenosti (tabela 6, kolona SRB2). Važno je istaći da je u trećoj fazi taksonomijskog strukturalnog algoritma primenjen drugačiji princip pronalazjenja novih parova klasa (u odnosu na algoritam primenjen pri poređenju srednjoškolskog modela i referentnog nastavničkog modela), budući da algoritam zasnovan na „disjoint“ opciji nije bio primenljiv. Dobijeni rezultati ukazuju da je algoritam za proširivanje prostora pretraživanja omogućio pronalazjenje srodnih „list“ potklasa „sličnih roditelja“.

Iz dobijenih rezultata o neuparenim *Skills* potklasama može se zaključiti da analizirani nastavnički kurikulum ne sadrži ishode kao što su: opisivanje najvažnijih primena veštačke inteligencije, razlikovanja etičkih i pravnih pitanja, ocenjivanja algoritama na osnovu njihove efikasnosti, ispravnosti i jasnoće, ocenjivanje korisnosti korisničkog interfejsa, testiranje programskog koda i sl. Stoga je potrebno kreirati nove i/ili izmeniti postojeće kurseve analiziranog nastavničkog kurikuluma tako da obuhvataju i nedostajuće ciljeve učenja predviđene ACM K12 modelom standarda. Uparene *Skills* potklase modela izabranog nastavničkog kurikuluma koje reprezentuju neodgovarajući nivo Blumove taksonomije se najčešće odnose na veštine programiranja. One su u nastavničkom kurikulumu često opisane glagolima koji ne odgovaraju najvišem nivou kognitivne dimenzije revidirane Blumove taksonomije.

5.6. Evaluacija sistema za usaglašavanje referentnog nastavničkog kurikuluma i odabranog nastavničkog kurikuluma iz Republike Srbije

U ovoj sekciji primenjeni su algoritmi opisani u sekcijama 5.2.3. – 5.2.5, pri čemu se razmatraju sve klase sadržane u oba ontološka modela, jer se porede kurikulumi istog nivoa (visokoškolskog). Pri tome se međusobno upoređuju samo potklase klasa koje odgovaraju istoj oblasti nastavničkog kurikuluma. Stoga se potklase klase

Informatics_domain_knowledge modela odabranog nastavničkog kurikuluma porede samo sa potklasama klase *Informatics_domain_knowledge* modela referentnog nastavničkog kurikuluma; potklase klase *General_knowledge* izvornog modela porede se samo sa potklasama klase *General_knowledge* ciljnog modela. Analogan princip važi i za potklase klase *General_educational_and_pedagogical_knowledge*, *Knowledge_of_teaching_practice*, *Knowledge_of_informatics_teaching_methods*. Za ostale kombinacije parova potklasa vrednost sličnosti postavlja se na nulu. Tako, na primer, sličnost sledećih parova potklasa postaje nula:

- Potklasa klase *Informatics_domain_knowledge* i potklasa klase *General_educational_and_pedagogical_knowledge*,
- Potklase klase *General_educational_and_pedagogical_knowledge* i potklase klase *General_knowledge* itd.

Pri uparivanju nastavničkih kurikuluma posmatraju se neuparene klase u oba modela kurikuluma. Iz neuparenosti klase modela odabranog nastavničkog kurikuluma mogu se izvesti zaključci o potrebi unapređenja ontološkog modela referentnog nastavničkog kurikuluma. S druge strane, neuparenost klase referentnog modela može ukazivati na to da odgovarajuće kompetencije nisu obuhvaćene analiziranim nastavničkim kurikulumom.

5.6.1. Primena algoritma za izračunavanje sličnosti roditeljskih klasa

Za sve slike prikazane u sekciji 5.6. važi da kolona „Source class“ sadrži klase nastavničkog kurikuluma iz Republike Srbije, dok kolona „Target class“ odgovara klasama modela referentnog nastavničkog kurikuluma. Prag je, eksperimentalno, postavljen na 70%. Karakteristični parovi roditeljskih klasa poređenih ontologija prikazani su na slici 40.

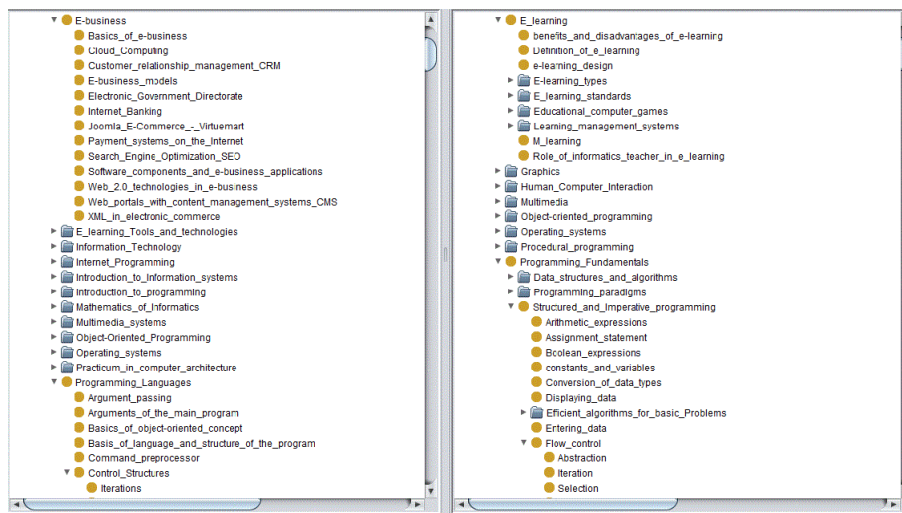
...	Source class	Target class	Type ...	Simila...
1	Communication_Devices	Computer_network_components	Equiv...	83.08%
2	Complex_Types	Abstract_data_types_and_structures	Equiv...	84.99%
3	Computer_Networks_and_Communications	Computer_Networks	Equiv...	88.97%
4	Computer_applications	Software_basics	Equiv...	82.02%
5	Computer_architecture	Computer_architecture_and_organ...	Equiv...	82.68%
6	Control_Structures	Flow_control	Equiv...	91.43%
7	Data_Types	Data_types_and_structures	Equiv...	87.91%
8	E-business	E_learning	Equiv...	71.03%
9	E_learning_Tools_and_technologies	E-learning_types	Equiv...	91.98%
10	Educational_technology	Educational_technology	Equiv...	80.64%
11	Internet_Programming	Web_programming	Equiv...	90.48%
12	Introduction_to_programming	Procedural_programming	Equiv...	76.86%
13	Mathematics_1	Algebra	Equiv...	80.87%
14	Mathematics_2	Mathematics	Equiv...	79.4%
15	Mathematics_of_Informatics	Mathematical_basis_of_informatics	Equiv...	86.06%
16	Methods_of_Teaching_Techniques_and_Inform...	Methods_of_teaching_Informatics	Equiv...	80.08%
17	Multimedia_systems	Multimedia	Equiv...	83.94%
18	Object-Oriented_Programming	Object-oriented_programming	Equiv...	82.9%
19	Operating_systems	Operating_systems	Equiv...	87.41%
20	Pedagogy	Didactics	Equiv...	83.68%
21	Practicum_in_computer_architecture	Hardware_basics	Equiv...	78.3%
22	Programming_Languages	Programming_Fundamentals	Equiv...	90.04%
23	Psychology_of_education_and_upbringing	Educational_psychology	Equiv...	73.95%
24	Relational_databases	Relational_model	Equiv...	93.09%
25	SQL	Structured_Query_Language_-_SQL	Equiv...	83.02%
26	Vocational_school_practice	School_practice	Equiv...	73.2%
27	Ways_for_transmitting_information	Protocols_of_transport_layer	Equiv...	75.15%
28	Web_Technologies	Web_technologies_and_developm...	Equiv...	87.72%

Slika 40. Deo uparenih roditeljskih klasa modela nastavničkih kurikuluma

Može se zapaziti da je uparen i deo klasa koje reprezentuju neinformatička znanja (redovi 10, 13, 14, 16, 20, 23, 26), u skladu sa goreopisanim principom pronalaženja parova u modelima nastavničkih kurikuluma. Tako se može videti da kursevi 'Matematika 1' i 'Matematika 2' nastavničkog kurikuluma odgovaraju znanjima reprezentovanim klasnim strukturama *Mathematics* i *Algebra* referentnog modela (redovi 13 i 14, slika 40). Kao i u sekcijama 5.4.1. i 5.5.1. i ovde se može zapaziti uticaj klasne strukture (potklasa i superklasa ako postoje) na ukupnu sličnost poređenih klasa. Tako, dobijena vrednost sličnosti klasa nije bliska 100% iako su klase identičnog naziva (poput parova klasa prikazanih u redovima 10, 18 i 19) ili leksički vrlo bliskog naziva (poput parova u redovima 3, 5, 11, 14, 25). Sličnosti su najčešće ispod 90%. Izuzetak je, na primer, par u redu 22, čija je sličnost nešto veća - iznad 90%, što se može objasniti strukturom ovih klasa koja je prikazana na slici 41. Posebno je uočljiv uticaj sličnosti potklasa na sličnost roditeljskih klasa u uparenim klasama opštih obrazovnih i pedagoških znanja. Tako, model referentnog nastavničkog kurikuluma sadrži posebne roditeljske klase: *General_Pedagogy* i *Didactics*. Model odabranog nastavničkog kurikuluma sadrži roditeljsku klasu *Pedagogy*, pa bi opravdana pretpostavka mogla biti da će sistem upariti tu klasu i klasu *General_Pedagogy*. Ipak, kurs 'Pedagogija' nastavničkog kurikuluma sadrži teme koje su najbližije sadržajima reprezentovanim potklasama klase *Didactics* referentnog modela, te je sistem upario klase *Pedagogy* i *Didactics* (red 20, slika 40). Takođe, parovima uparenih klasa ne pripada par *{Methodical_practicum_in_*

programming basics, Methods_of_teaching_programming}. Zbog nedovoljne sličnosti potklasa dobijena je vrednost podudarnosti koja nije veća od zadatog praga.

Par sadržan u redu 8 slike 40 može se protumačiti kao netačan. Deo strukture uparenih klasa *E-bussines* i *E-learning* prikazan je na slici 41.



Slika 41. Deo strukture klasa modela nastavničkih kurikuluma

Nakon primene algoritma za izračunavanje sličnosti roditeljskih klasa u modelu odabranog nastavničkog kurikuluma neuparene su klase:

- *Descriptive_geometry,*
- *Discrete_Mathematics,*
- *Communicology,*
- *Docimology,*
- *Introduction_to_Information_systems,*
- *Information_Technology,*
- *Methodical_practicum_in_Computer_Architecture_and_Orga
nization,*
- *Methodical_practicum_in_computer_graphics,*
- *Methodical_practicum_in_programming_basics,*
- *Methodical_practicum_in_Robotics.*

U modelu referentnog nastavničkog kurikuluma primeri neuparenih roditeljskih klasa su:

- *Artificial_intelligence,*
- *Graphics,*
- *Human_Computer_Interaction,*
- *Social_and_Professional_Issues_and_Informatics,* kao i njene roditeljske potklase:
 - *Computer_ethics,*
 - *Informatics_basics,*
 - *Privacy_and_security,*

- *Methods_of_teaching_programming,*
- *Research,*
- *Problem_solving,*
- *Algorithms,*
- *Modeling_and_simulation.*

Procenat uparenih *Knowledge* potklasa modela nastavničkog kurikuluma je 4.67%, dok je procenat uparenih *Knowledge* potklasa modela referentnog nastavničkog kurikuluma 5.27%.

5.6.2. Primena algoritma za izračunavanje sličnosti „list“ klasa uparenih roditelja

Primeri “najboljih parova” „list“ klasa uparenih roditelja prikazani su na slici 42.

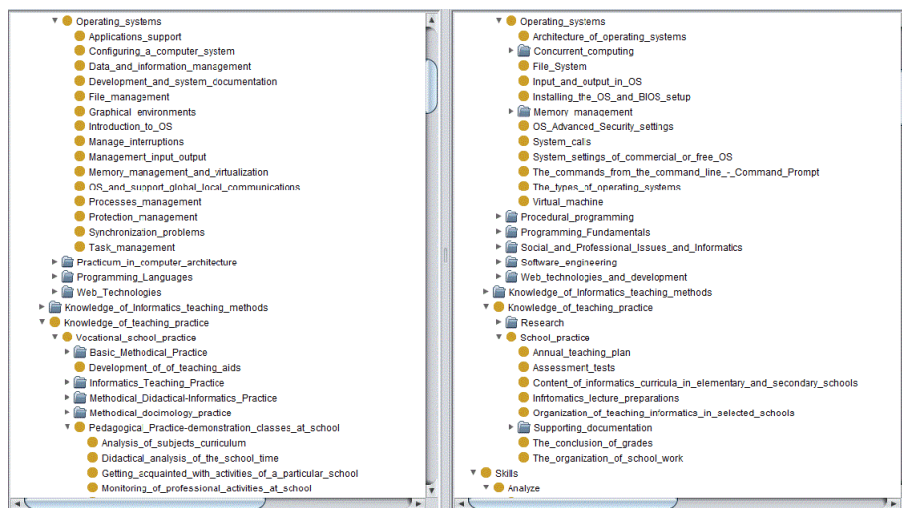
...	Source class	Target class	Ty...	Simila...
1	Basic_Theoretical_Principles_of_Educational_Technologi...	The_concept_of_educational_technology	Eq...	82.36%
2	Characteristics_of_function_that_measures_the_Inform...	Units_for_measuring_the_amount_of_infor...	Eq...	74.24%
3	Class_hour	School_hour	Eq...	77.0%
4	Communication_and_collaborative_tools	Computer-supported_collaborative_learnin...	Eq...	75.15%
5	Concepts_and_approaches_in_web_design	Principles_of_web_design	Eq...	87.28%
6	Curriculum_programs_technical-technological_and_infor...	Planning_of_teaching_and_informatic_curri...	Eq...	75.73%
7	Didactical_analysis_of_the_school_time	The_organization_of_school_work	Eq...	73.4%
8	Entity-relationship_model	Entity-Relationship_model_-_ER	Eq...	100.0%
9	File_management	File_System	Eq...	96.25%
10	HTML5_and_CSS3_standards_and_related_technologies	HTML_Document_Naming_Scheme	Eq...	74.87%
11	Internal_subprograms	Procedures_and_parameters	Eq...	74.07%
12	Inverse_matrix	Matrix_inversion	Eq...	96.83%
13	Java_script	Client-side_scripting	Eq...	100.0%
14	Machine_Programming	Assembler_programming	Eq...	100.0%
15	Management_input_output	Input_and_output_in_OS	Eq...	70.77%
16	Memory_management_and_virtualization	Virtual_memory	Eq...	97.78%
17	P2P_network	Peer-to-peer	Eq...	100.0%
18	Planning_and_preparation_of_teaching	Infrtomatics_lecture_preparations	Eq...	76.29%
19	Processes_management	Synchronization_of_processes	Eq...	100.0%
20	Software_licenses	Software_distribution_and_download	Eq...	100.0%
21	Specifics_of_the_measurement_and_assessment_of_a...	Evaluation_of_student_achievement_in_inf...	Eq...	70.62%
22	Subqueries	Nested_query	Eq...	94.33%
23	Teaching_Methods_of_Techniques_and_Informatics_as...	Teaching_strategies_and_instruction_in_in...	Eq...	84.09%
24	Text_editors	Software_applications_for_text	Eq...	75.69%
25	The_practical_implementation_of_architectural_features...	The_organization_and_working_principle_...	Eq...	74.71%
26	Theories_of_Learning_and_Teaching	Teaching_and_learning_strategies	Eq...	83.13%
27	Types_of_e-learning	Synchronous_and_asynchronous_e_learn...	Eq...	81.57%
28	Working_with_Files	Input_and_output_streams	Eq...	72.34%
29	XML_Standards_and_Technologies	XML_technology	Eq...	100.0%
30	Quality_of_Service	Cloud_computing_services	Eq...	76.19%

Slika 42. Uparene „list“ klase modela nastavničkih kurikuluma

Za razliku od rezultata uparivanja prethodnih kombinacija ontoloških modela (sekcija 5.4.2. i 5.5.2) ovde su uparene i „list“ klase koje odgovaraju neinformatičkim znanjima. Tako parovi klasa prikazani u redovima 7 i 18 odgovaraju temama koja pripadaju znanjima nastavne prakse; parovi klasa u redovima 1, 3 i 26 pripadaju opštim obrazovnim i pedagoškim znanjima; par klasa u redu 12 pripada opštim znanjima (matematičke oblasti), dok parovi klasa u redovima 6, 21 i 23 odgovaraju znanjima metodike nastave informatike. Analizom samo naziva uparenih klasa (i/ili njihovih labela) i vrednosti dobijene sličnosti može se pretpostaviti sa velikom verovatnoćom da su parovi klasa u redovima 8, 9, 12, 13, 14, 17, 19, 20, 22 i 29 ispravno upareni (što se uvidom u strukturu ontologija i potvrđuje). S druge strane,

primenjujući isti kriterijum za procenu tačnosti uparivanja (samo nazive uparenih klasa i vrednost sličnosti) za parove u redovima 2, 4, 11, 21, 25 i 28, ispravnost uparenosti nije u toj meri uočljiva. Ipak, detaljnim uvidom u hijerarhijske strukture može se zaključiti da su i klase sadržane u navedenim redovima ispravno uparene.

Tako, za klase sadržane u redu 21 važi da predstavljaju ekvivalentne sadržaje kurikuluma, jer pripadaju uparenim klasnim strukturama (red 16, slika 40). Slično, analizirajući samo nazive klasa u redovima 15 i 16 može se zapaziti da se klase modela odabranog nastavničkog kurikuluma (kolona „Source class“) mogu odnositi i na druge oblasti informatičkih znanja od onih opisanih ciljnim klasama. Na primer, upravljanje ulazom i izlazom ne mora nužno da odgovara ulazu i izlazu u operativnim sistemima (red 15). Ipak uvidom u ontološke strukture kojima pripadaju (koje reprezentuju kurseve/oblasti operativnih sistema) (slika 43), zapaža se da ove klase jesu ispravno uparene.



Slika 43. Deo struktura klasa koje reprezentuju operativne sisteme i školsku praksu

Na slici 42 postoje parovi klasa koji se mogu protumačiti kao netačni (redovi 7, 10 i 30). Objašnjenje prikazanih netačnosti sistema dato je u nastavku. Klase prikazane u redu 7 (*Didactical analysis of the school time* i *The organization of school work*) pripadaju hijerarhijskim strukturama (slika 43) koje su uparene prethodnim algoritmom za utvrđivanje roditeljske sličnosti (red 26, slika 40). Algoritam je u ovoj fazi u tim ontološkim strukturama pretraživao terminološki „najsličnije list klase“ iznad praga, te je dobijen par klasa {*Didactical analysis of the school time*, *The organization of school work*} koji suštinski ne reprezentuje ekvivalente teme kurikuluma. Analogno su dobijeni parovi klasa sadržani u redovima 10 i 30. Par klasa {*Quality of Service*, *Cloud computing services*} (red 30, slika 42) je dobijen kao posledica uparenosti roditeljskih klasa prikazanih u redu 28 slike 40. Pri tome za klasu *Cloud computing services* modela referentnog nastavničkog kurikuluma postoji odgovarajuća klasa u modelu nastavničkog kurikuluma (*Cloud Computing*). Međutim, klasa *Cloud Computing*

modela nastavničkog kurikuluma predstavlja potklasu klase *E-business* koja nije uparena ni sa jednom roditeljskom klasom klase *Cloud_computing_services* referentnog modela. Stoga, pripadnost klasa *Cloud_Computing* i *Cloud_computing_services* različitim hijerarhijskim strukturama uz dobijenu uparenost $\{Quality_of_Service, Cloud_computing_services\}$ može dalje prouzrokovati ili da klasa *Cloud_Computing* modela nastavničkog kurikuluma ostane neuparena ili da bude netačno uparena.

Primeri uparivanja „list“ klase sa klasom koja je superklasa samo „list“ klasama su prikazani u redovima 8 i 20 slike 42.

Procenat uparenih *Knowledge* potklasa nakon ove faze u slučaju ontološkog modela odabranog nastavničkog kurikuluma je 23,37%, dok je za ontološki model referentnog nastavničkog kurikuluma ta vrednost 26.34%.

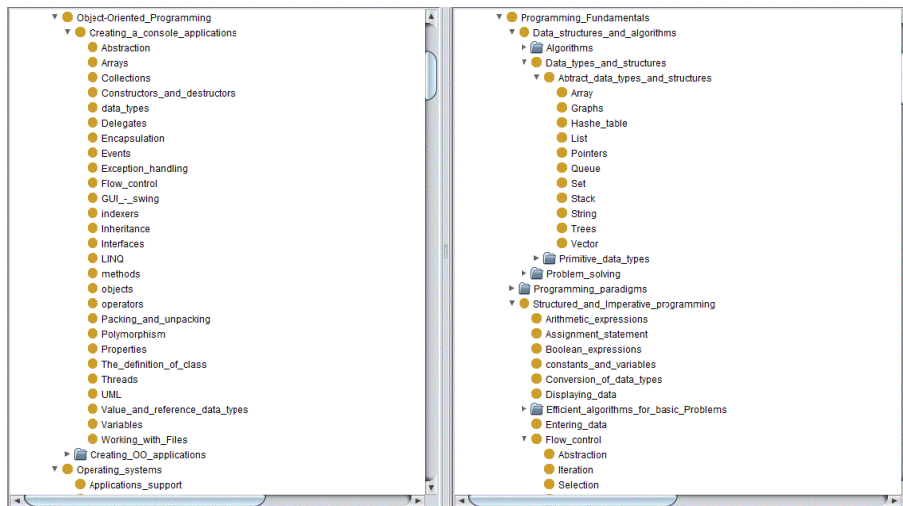
5.6.3. Primena algoritma za izračunavanje sličnosti „list“ klasa neuparenih roditelja

S obzirom da su u modelu referentnog nastavničkog kurikuluma definisane „disjoint“ klase, ovde je primenjen algoritam za utvrđivanje sličnosti „list“ klasa neuparenih roditelja zasnovan na proširivanju prostora pretraživanja u „ne-disjunktnim“ klasnim strukturama (sekcija 5.2.3.3.). Karakteristični parovi klasa prikazani su na slici 44.

...	Source class	Target class	Ty...	Simila...
1	Abstraction	Abstraction	Eq...	100.0%
2	Adult_education	Multicultural_education	Eq...	80.51%
3	Applications_support	System_and_application_software	Eq...	97.78%
4	Areas_of_upbringing	Components_of_upbringing	Eq...	71.61%
5	Arrays	Array	Eq...	100.0%
6	Control_Units	Control_Unit_-_CU	Eq...	100.0%
7	Defining_the_type	Reference_type	Eq...	81.02%
8	Examples_of_Use_of_Educational_Technologies	Examples_of_educational_software	Eq...	76.51%
9	Fundamentals_of_object-oriented_languages	Syntax_of_object_oriented_language	Eq...	87.63%
10	Graphical_environments	GUI_components_of_OS	Eq...	100.0%
11	Internetworking	Internet	Eq...	93.78%
12	Objective_and_outcomes_of_upbringing	Aims_and_objectives_of_upbringing	Eq...	72.43%
13	Principles_of_Creation_and_implementation_of_educ...	The_principles_of_creating_educational_soft...	Eq...	78.19%
14	The_system_of_Discipline	The_system_of_pedagogical_disciplines	Eq...	85.71%
15	Threads	Threads	Eq...	100.0%

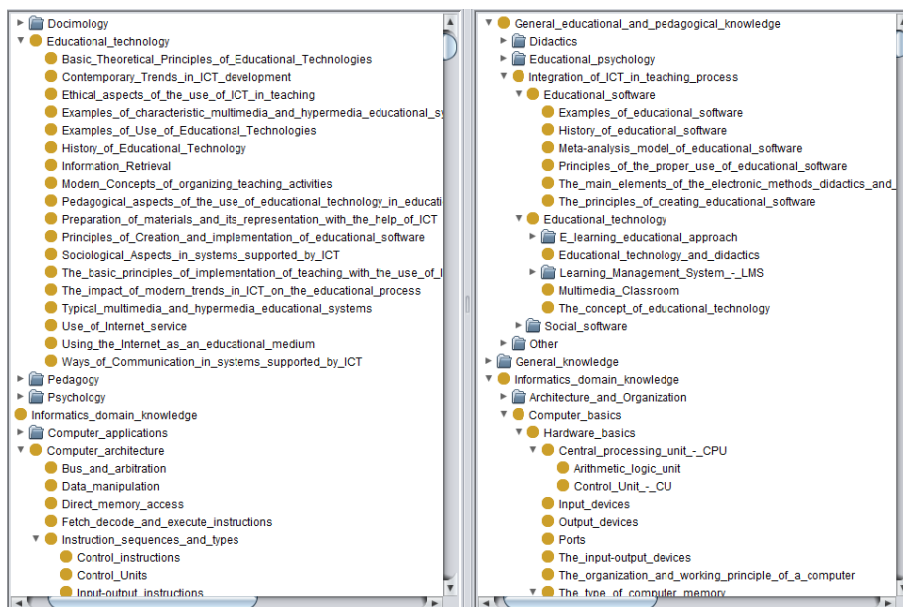
Slika 44. Parovi „list“ klasa neuparenih roditelja modela nastavničkih kurikuluma

Klase *Abstraction* i *Arrays* modela nastavničkog kurikuluma pripadaju klasnoj strukturi roditeljske klase *Object-Oriented_Programming* dok njima odgovarajuće klase referentnog modela u redovima 1 i 5 („Target class“ kolona) pripadaju klasnoj strukturi roditeljske klase *Programming_Fundamentals* (slika 45). Sa slike 40 se može zaključiti da ove klasne strukture nisu uparene. Tako roditelji klasa *Abstraction* i *Arrays* modela nastavničkog kurikuluma (red 18, slika 40) nisu upareni sa superklasama klasa *Abstraction* i *Array* referentnog modela (redovi 2, 7, 22, slika 40). Ipak klasa *Object-oriented_programming* referentnog modela nije definisana kao „disjoint“ klasi *Programming_Fundamentals* te je u ovom koraku algoritam pretraživao i potklase klase *Programming_Fundamentals*.



Slika 45. Međusobno neuparene klasne strukture nastavničkih modela

Analogno su dobijeni i ostali parovi klasa u ovoj fazi. Red 6 ukazuje i na uparenost potklase roditeljskih klasa *Computer_architecture*, *Instruction_sequences_and_types* i potklase roditeljskih klasa *Computer_basics*, *Hardware_basics*, *Central_processing_unit_-_CPU* (slika 46), iako ove klasne strukture nisu uparene primenom algoritma za određivanje sličnosti roditeljskih klasa. Slično, redovi 8 i 13 ukazuju na uparenost potklasa klase *Educational_technology* modela nastavničkog kurikuluma i potklasa klase *Educational_software* modela referentnog nastavničkog kurikuluma (slika 46), iako roditeljske klase *Educational_technology* i *Educational_software* nisu uparene.



Slika 46. Međusobno neuparene klasne strukture modela nastavničkih kurikuluma

Sa slike 44 se može videti da se ovaj algoritam, kao i prethodni, primenjuje i na klase koje reprezentuju ne-informatička (nedomenska) znanja (redovi 2, 4, 8, 12, 13, 14). To je posledica principa

kreiranja referentnog nastavničkog modela u koga su mapirane i informacije o mogućoj (ne)disjunktnosti neinformatičkih znanja.

U redovima 2 i 7 prikazani su parovi klasa koji se mogu smatrati netačnim.

Procenat uparenih *Knowledge* potklasa modela nastavničkog kurikuluma je 42.65%, dok je kod modela referentnog nastavničkog kurikuluma taj procenat veći i iznosi 48.08%.

5.6.4. Primena algoritma za izračunavanje relacione sličnosti

Pri poređenju *Skills* potklasa modela nastavničkih kurikuluma prag je, eksperimentalno, postavljen na 60%. Kolona “*Bloom*” (slika 47) prikazuje rezultate poređenja nivoa revidirane Blumove taksonomije uparenih *Skills* potklasa. Pri tome, oznaka τ ukazuje na to da je veština reprezentovana klasom modela referentnog nastavničkog kurikuluma na višem ili jednakom nivou od odgovarajuće veštine modela nastavničkog kurikuluma. Oznaka \perp ukazuje na suprotan slučaj.

...	Source class	Target class	Ty...	Simila...	...
1	Allocate_resources-memory_management	Apply_memory_management_principles	Eq...	76.6%	τ
2	Analyze_and_select_educational_software_and_technology_for_use_in...	Select_the_most_suitable_educational_software_for_use_in_certain...	Eq...	86.89%	τ
3	Analyze_different_network_parameters	Implement_local_area_networks	Eq...	69.41%	\perp
4	Analyze_the_course_contents_in_the_technique_and_informatics_field	Identify_the_national_high_school_CS_curriculum_intending_to_teach...	Eq...	71.05%	\perp
5	Apply_basic_mathematical_concepts_ideas_and_results_in_vocational_c...	Understand_concepts_and_assertions_of_mathematical_analysis_a...	Eq...	63.43%	\perp
6	Apply_didactic_principles_as_criteria_of_validity_of_implemented_instruc...	Use_assessment_results_to_capture_student_learning_provide_rem...	Eq...	60.98%	τ
7	Conducts_classes	Teach	Eq...	100.0%	τ
8	Create_different_solutions_for_e-learning	Create_e_learning_content	Eq...	80.17%	τ
9	Create_multimedia_and_or_hypermedia_lessons_by_using_educational_s...	Create_multimedia_lessons_using_educational_software	Eq...	85.71%	τ
10	Creatively_develop_the_educational_process_and_extracurricular_activiti...	Plan_instruction_involving_students_independently_using_computing...	Eq...	71.36%	τ
11	Database_design	Design_databases	Eq...	100.0%	τ
12	Determine_the_objectives_and_outcomes_of_technical_and_informatics...	Reflect_on_the_delivered_instructional_activity_suggesting_modifica...	Eq...	62.63%	τ
13	Identify_different_types_of_buses_in_a_computer_system	Identify_the_purpose_of_the_major_hardware_components_of_a...	Eq...	71.33%	τ
14	Install_and_launch_a_variety_of_internet_services	Use_of_internet_in_a_safe_and_efficient_manner	Eq...	67.18%	τ
15	Manipulate_multimedia_content	Set_the_multimedia_on_the_web	Eq...	70.5%	τ
16	Organize_classes_tailored_to_students_with_different_individual_charac...	Teach_CS_lessons_involving_students_independently_using_compu...	Eq...	65.31%	\perp
17	Prepare_and_implement_various_docimology_procedures	Develop_assessment_procedures_to_determine_successful_perfor...	Eq...	66.45%	τ
18	Recognize_the_WWW_and_internet_protocol_services_and_networking	Understand_elements_of_the_Web_and_their_interactions	Eq...	62.96%	τ
19	Recognize_the_basic_types_and_data_structures	Select_abstract_data_structure	Eq...	73.54%	τ
20	Recognize_the_principles_of_computer_network_organization_and_design	Understand_computer_networks_supporting_communication_and_c...	Eq...	63.75%	τ
21	Select_the_most_suitable_e-learning_product	Select_appropriate_e_learning_approach_to_teach_a_specific_cont...	Eq...	66.53%	τ
22	Use_classes_and_objects_in_the_programming_language	Select_design_patterns_for_object-oriented_software	Eq...	72.76%	τ
23	Web_design_for_standard_and_mobile_platform	Design_web_pages	Eq...	68.32%	τ
24	Work_on_different_operating_systems	Use_various_OS	Eq...	70.7%	τ
25	Write_simple_machine_programs	Understand_assembler_programming	Eq...	69.21%	\perp

Slika 47. Uparene *Skills* potklase modela nastavničkih kurikuluma

Neki od parova *Skills* potklasa koje odgovaraju različitim nivoima Blumove taksonomije prikazani su u redovima 4, 5, 16 i 25. Tako nastavnički kurikulum, u okviru kursa 'Arhitektura računara' predviđa ishod „piše jednostavne mašinske programe“ reprezentovan klasom *Write_simple_machine_programs*, potklasom klase *Create*. Primenom relacionog uparivača ova klasa uparena je sa klasom *Understand_assembler_programming* (red 25, slika 47), potklasom klase *Remember-understand* referentnog modela, koja reprezentuje niži nivo Blumove taksonomije.

Parovi klasa sa slike 47 koji se mogu tumačiti kao netačni prikazani su u redovima 3, 6, 10, 12 i 13.

Procenat uparenih *Skills* potklasa modela nastavničkog kurikuluma je 58.14%, dok je procenat uparenih *Skills* potklasa referentnog modela 58.4%.

Neuparenost *Skills* potklasa najčešće je posledica neuparenosti *Knowledge* klasnih struktura. Tako su u modelu odabranog nastavničkog kurikuluma ostale neuparene klase:

- povezane sa klasom *Descriptive_geometry* poput:

- *Resolve cross-sections of the body and the plane,*
- *Solve the basic tasks of descriptive geometry,*
- povezano sa klasom *Communicology*:
 - *Define function and effects of mass media,*
 - *Recognize the types of interlocutors,*
- povezano sa klasom
Methodical practicum in computer graphics:
 - *Recognize the latest developments in this scientific teaching area,*
 - *Teach students how to use specific software,*
- povezano sa klasom
Methodical practicum in programming basics:
 - *Conduct applicative research in the field of PB,*
 - *Explain the place of programming field in engineering and teaching,*
- povezano sa klasom *Methodical practicum in Robotics*:
 - *Recognize the teaching content in robotics toward high school program,*
 - *Transfer of modern achievements in the field of robotics in classroom lessons.*

Analogno, u modelu referentnog nastavničkog kurikuluma primeri neuparenih *Skills* potklasa:

- povezanih sa klasom *Artificial intelligence* su:
 - *Assess possible applications and limitations of the Artificial Intelligence,*
 - *Understand possibilities of application of machine learning,*
- povezanih sa klasom *Computer ethics* su:
 - *Analyze the practice of social and professional responsible informaticians,*
 - *Discuss intellectual property,*
- povezanih sa klasom *Informatics basics* su:
 - *Describe informatics theories,*
 - *Discuss limits of computing,*
- povezanih sa klasom *Privacy and security* poput:
 - *Discuss privacy issues,*
 - *Discuss security issues.*

Takođe, pojedine *Skills* potklase su ostale neuparene iako su povezane sa uparenim klasnim strukturama. Primeri takvih klasa su

- u modelu odabranog nastavničkog kurikuluma:

- *Understand the concepts the model and the benefits of e-business,*
- *Implement e-business projects and consultancy services,*
- *Recognize network OS to manage local networks,*
- *Recognize the resources of IS and CASE development tools,*
- *Develop information systems and manage information systems development projects,*
- *Diagnose and fix faults in the network.*
- u referentnom modelu:
 - *Discuss the strengths and weaknesses of two different programming paradigms in the context of a given problem,*
 - *Assess whether a computation is algorithmically solvable problem at all or whether it is a solvable but difficult algorithmic problem,*
 - *List problem solving phases,*
 - *Evaluate problem solution,*
 - *Analyze algorithms using complexity efficiency aesthetics and correctness,*
 - *Evaluate novel interaction scenarios in design,*
 - *Assess the suitability of languages for specific purpose*
 - *Evaluate relevancy and authenticity of information on the Internet,*
 - *Design programs in languages from two different programming paradigms in a manner appropriate to each paradigm.*

5.6.5. Primena 1:N algoritma

Karakteristični parovi klasa u 1:N relaciji prikazani su na slici 48. Slika sadrži primere sva tri moguća tipa relacija. S obzirom da su matematička znanja (predmeti) konkretnog nastavničkog kurikuluma opisana velikim brojem tema i podtema, ona su reprezentovana u modelu značajnim brojem klasa i potklasa. Matematička znanja referentnog modela reprezentovana su, s druge strane, manjim brojem klasa i nižom struktuiranošću. Stoga je između ekvivalentnih klasa matematičkog domena modela nastavničkog kurikuluma i referentnog modela moguća „Subclass“ relacija. Tako klase modela nastavničkog kurikuluma prikazane u redovima 1, 2, 4 i 5 predstavljaju potklasu klase *Hypothesis testing* modela referentnog nastavničkog kurikuluma. Slično, znanja vezana za softverske licence (preuzimanje softvera i distribuciju) su reprezentovana sa većim brojem klasa i većom struktuiranošću u modelu referentnog nastavničkog

kurikuluma, te je za te parove klasa dobijena „Superclass“ relacija (redovi 6, 7, 8, 9, 11).

...	Source class	Target class	Type of r...	Simila...
1	Hypotheses_about_the_different_parameters	Hypothesis_testing	Subclass	80.0%
2	Hypotheses_about_the_parameter_value	Hypothesis_testing	Subclass	80.0%
3	Parametric_hypothesis_tests	Hypothesis_testing	Equivalence	80.0%
4	T-test	Hypothesis_testing	Subclass	80.0%
5	Test_of_equality_of_variances	Hypothesis_testing	Subclass	80.0%
6	Software_licenses	Commercial_software	Superclass	100.0%
7	Software_licenses	Freeware	Superclass	100.0%
8	Software_licenses	Open_source_software	Superclass	100.0%
9	Software_licenses	Shareware	Superclass	100.0%
10	Software_licenses	Software_distribution_and_download	Equivalence	100.0%
11	Software_licenses	Trial_software	Superclass	100.0%

Slika 48. Primeri parova klasa nastavničkih modela u 1:N relaciji

Konačan procenat uparenih *Knowledge* potklasa modela nastavničkog kurikuluma, nakon primene svih uparivača, je 45.65%, dok je uparenost *Knowledge* potklasa modela referentnog nastavničkog kurikuluma 52.22%.

I ovde neuparenost klasa analiziranih modela predstavlja posledicu jednog od dva moguća uzroka: nepostojanja odgovarajućih klasa u poređenom modelu ili drugačije struktuiranosti srodnih klasa. Prvi slučaj ukazuje na to da znanja koja su reprezentovana neuparenim klasama nisu sadržana u poređenom kurikulumu, te je potrebno razmotriti unapređenje kurikuluma tako da sadrži nedostajuća znanja. U drugom slučaju moguće je da postoje odgovarajuća znanja u poređenim kurikulumima ali da sistem nije upario odgovarajuće klase. Tada je potrebno razmotriti sinhronizaciju hijerarhijskih struktura analiziranih kurikuluma tako da slične teme pripadaju srodnim tematskim oblastima (kursevima) ili je potrebno razmotriti unapređenje algoritma tako da pronalazi i ove, neuparene, srodne klase, uz očuvanje mere preciznosti.

Primeri neuparenih klasa modela odabranog nastavničkog kurikuluma za koje ne postoje odgovarajuće klase u modelu referentnog nastavničkog kurikuluma su:

- potklase klase *Descriptive_geometry*:
 - *Projection_body*,
 - *Projection_of_lines*,
 - *The_importance_and_the_basic_principles_of_descriptive_geometry, itd.*,
- potklase klase *Discrete_Mathematics*:
 - *Traveling_Salesman_Problem*,
 - *Hamilton_Cycle*,
 - *Planarity*,
 - *Proof_Techniques itd.*
- veliki broj potklasa uparenih klasa *Mathematics_1, Mathematics_2, Probability_and_Statistics* poput:
 - *Ostrogradsky_method*,
 - *L_Hôpital_s_rule*,

- *Chebyshev_Inequality, itd.*
- potklase klase *Communicology*:
 - *Business_Conversation,*
 - *Communication_Competence,*
 - *Obstacles_and_conflicts_in_communication, itd.*
- potklase klase *Docimology*:
 - *Computer_adaptive_tests,*
 - *E-tests,*
 - *Docimological_research,*
 - *Evaluation_on_Portfolios, itd.*
- neke potklase uparene klase *Educational_technology*:
 - *Ways_of_Communication_in_systems_supported_by_ICT,*
 - *Preparation_of_materials_and_its_representation_with_the_help_of_ICT,*
- klase koje odgovaraju tehničkom domenu (potklase klase *Introduction_to_technical_systems*):
 - *Robotics,*
 - *Programmable_machines,*
 - *Medical_program_equipment, itd.*
- deo potklasa informatičkog domena poput:
 - *IT_in_Biomedicine,*
 - *IT_in_industry,*
 - *fetch_decode_and_execute_instructions,*
 - *Environment_to_work_with_Oracle_databases,*
 - *The_physical_Business_process_models,*
 - *Maintenance_of_information_systems,*
 - *Customer_relationship_management_CRM,*
 - *E-business_models,*
 - *Software_development_and_system_documentation,*
 - *The_syntax_of_the_graphic_language_IDEF1X,*
 - *The_semantics_of_graphical_language_IDEF1X,*
 - *Shannon_definition_of_information,*
 - *Protection_of_multimedia_systems,*
- potklase neuparenih klasa *Methodical_practicum_in_Computer_Architecture_and_Organization,* *Methodical_practicum_in_computer_graphics,* *Methodical_practicum_in_programming_basics,* *Methodical_practicum_in_Robotics* poput:
 - *Competencies_of_teachers_in_AOR,*
 - *Innovation_of_teacher_knowledge_in_CG,*
 - *The_development_of_a_scientific_field_PB,*
 - *Evaluation_of_Teaching_in_PB,*

- *Specifics_of_teaching_robotics,*
- *Practical_application_of_robots_in_teaching_robotics_in_high_school,*
- deo potklasa uparene roditeljske klase *Methods_of_Teaching_Techniques_and_Informatics:*
 - *Exemplary_teaching,*
 - *Programmed_teaching,*
- značajan deo potklasa uparene roditeljske klase *Vocational_school_practice* (s obzirom da su ovde mapirana znanja iz detaljno opisana tri predmeta: Stručna školska praksa 1, 2 i 3) poput:
 - *Preparation_and_implementation_of_lessons_in_inclusive_classrooms,*
 - *The_integration_of_assistive_technology_in_inclusive_teaching, itd.*

Primeri neuparenih klasa modela referentnog nastavničkog kurikuluma za koje ne postoje odgovarajuće klase u modelu odabranog nastavničkog kurikuluma su:

- deo potklasa klase *General_educational_and_pedagogical_knowledge* poput:
 - *Active_learning_and_teaching,*
 - *Motivational_factors,*
 - *Possibilities_of_using_social_software_in_education,*
 - *Advantages_and_disadvantages_of_using_social_software_in_education,*
- deo potklasa klase *Informatics_domain_knowledge* poput:
 - *Problem_solving_phases,*
 - *Efficient_algorithms_for_basic_problems,*
 - *Reverse_engineering,*
 - *Formal_specification,*
 - *Characteristics_of_digital_integrated_circuits,*
 - *Analog_to_digital_conversion,*
 - *The_relevancy_and_authenticity_of_information_on_the_Internet,*
 - *Risks_and_security_on_the_Internet,*
 - *Device_drivers,*
 - *Distributed_computing,*
 - *Simulation_language,*
 - *M_learning,*
 - *Educational_computer_games,*
 - *Algorithm_efficiency,*

- *Performance_optimization_of_computers (The_impact_of_the_computer_components_on_the_performance, Techniques_that_speed_up_the_processing),*
- *Sequential_circuits* potklase poput:
 - *Flip_flop,*
 - *Sequential_logic,*
 - *Synchronous_and_asynchronous_sequential_circuits,*
- *Artificial_intelligence* klasna struktura:
 - *Artificial_neural_networks,*
 - *Definition_of_artificial_intelligence,*
 - *Ethical_issues_in_Artificial_Intelligence,*
 - *Machine_learning,*
- deo *Graphics* potklasa poput:
 - *Methods_of_presenting_static_images_in_computer,*
 - *File_Formats_in_computer_graphics,*
 - *Graphics_hardware,*
- *Human_Computer_Interaction* potklase kao što su:
 - *Measures_for_evaluation_in_HCI,*
 - *The_most_common_interface_design_mistakes,*
 - *User-centered_design_process,*
- *Algorithm_design_paradigms* potklase poput:
 - *Divide_and_conquer,*
 - *Branch_and_bound,*
 - *Greedy_algorithms,*
- *Limits_of_computing* potklase:
 - *Computationally_hard_problems,*
 - *Computationally_intensive_problems,*
 - *Computationally_unsolvable_problem,*
- *Computer_ethics* potklase
 - *Legislation_in_computing,*
 - *Protection_of_intellectual_property_rights,*
 - *The_difference_between_the_right_to_access_and_distribution_rights,*
- *Informatics_basics* potklase poput:
 - *Informatics_theories,*
 - *The_difference_between_IT_literacy_and_Informatics,*
- *Privacy_and_security* potklase kao što su:
 - *Identity_theft,*
 - *Impact_of_legislation_on_privacy_and_security,*
 - *Software_crime,*
- *Methods_of_teaching_programming* potklase poput:

- *Software_visualization (Algorithm_animation, Program_visualization, Visual_programming)*
- *Suitable_tasks_to_illustrate_the_programming_concepts,*
- *Programming_languages_used_in_schools,*
- *Research* potklase:
 - *Qualitative_methods,*
 - *Quantitative_methods_for_data_collection_and_analysis,*
 - *Plagiarism,*
 - *Searching_literature.*

Karakteristične primere neuparenosti klasa nastale kao posledica različitih struktuiranosti poređenih modela predstavljaju klase koje pripadaju *Graphics* strukturi modela referentnog nastavničkog kurikuluma. Klase *Rotation, Affine_transformations, Viewing_transformations, Three_dimensional_modeling* su ostale neuparene iako, na primer, klasa *Descriptive_geometry* modela nastavničkog kurikuluma sadrži potklasu *Transformation_and_Rotation*, dok klasa *CAD_CAM_technologies* reprezentuje kurs koji obrađuje teme 3D modela. Takođe, kompjuterska grafika se u nastavničkom kurikulumu posebno navodi samo kao jedna tema u sadržaju predmeta 'Informacione tehnologije'.

Takođe, deo potklasa klase *Discrete_mathematics* modela odabranog nastavničkog kurikuluma ostaju neuparene poput: *Binary_decision_diagrams, Minimization_of_Boolean_functions, Predicate_logic, Propositional_Logic, Automata, Turing_Machines* iako se u strukturi informatičkih (domenskih) znanja modela referentnog nastavničkog kurikuluma mogu pronaći srodne klase.

5.6.6. Analiza rezultata

Evaluacija sistema je pokazala da su vrednosti tačnosti (0.70), odziva (0.82) i F-mere (0.76), viši pri poređenju modela nastavničkih kurikuluma, nego pri poređenju ostalih kombinacija ontoloških modela. Iako se to može posmatrati kao očekivana posledica poređenja kurikuluma istog nivoa obrazovanja, rezultati su zadovoljavajući, posebno uzevši u obzir da je ovde upoređivan najveći broj klasa (ontološki modeli nastavničkih kurikuluma pojedinačno sadrže značajno više klasa od modela srednjoškolskog kurikuluma). S druge strane, procenti uparenosti *Knowledge* i *Skills* potklasa nisu visoki. Neuparenosti *Knowledge* i *Skills* potklasa, u zavisnosti od posmatranog modela nastavničkog kurikuluma, se mogu tumačiti dvojako.

1. Dobijeni rezultati neuparenosti klasa modela odabranog nastavničkog kurikuluma iziskuju razmatranje proširivanja referentnog modela tako da sadrži klase koje reprezentuju znanja i ishode kurseva dokimologije, komunikologije, diskretne matematike, nacrtna geometrije, metodičke

praktikume iz različitih oblasti informatičkog domena. Budući da analizirani nastavnički kurikulum predviđa izučavanje (realizovanje) stručne školske prakse kroz tri različita kursa, brojne klase koje reprezentuju ova znanja su ostale neuparene. Posebno je uočljivo da se oblast informacionih sistema u odabranom nastavničkom kurikulumu detaljno izučava u okviru dva kursa, dok u referentnom modelu odgovarajuća znanja nisu uopšte mapirana u posebnu klasnu strukturu.

S druge strane, klase koje odgovaraju sadržajima definisanim u kursu 'Informacione tehnologije' odabranog nastavničkog kurikuluma su u referentnom modelu strukturirane u okviru različitih klasnih struktura (*Computer_basics*, *Informatics_basics*). Ipak, pretraživanjem među klasama neuparenih i ne-disjunktnih roditelja omogućeno je da pojedine klase koje reprezentuju sadržaje predmeta 'Informacione tehnologije' budu uparene.

2. Neuparenost klasa referentnog modela ukazuje na kompetencije (znanje i veštine) kojima je potrebno proširiti odabrani nastavnički kurikulum iz Republike Srbije da bi bio usaglašen sa referentnim nastavničkim kurikulumom. Iz dobijenih rezultata se vidi da nastavnički kurikulum ne predviđa izučavanje principa (znanja i povezanih ishoda) veštačke inteligencije, interakcije čovek računar, kompjuterske etike, privatnosti i bezbednosti, koji su zastupljeni u većini analiziranih svetskih kurikuluma. Takođe, u poređenju sa referentnim modelom, odabranim nastavničkim kurikulumom nisu dovoljno obuhvaćeni ni neki od važnih koncepata koji se odnose na rešavanje problema, algoritme, ograničenja računara, veštinu programiranja i rad sa grafikom.

Deo *Knowledge* potklasa (pre svega referentnog modela) je ostao neuparen i kao posledica toga što su različitim nivoom detaljnosti opisana ista znanja u poređenim modelima nastavničkih kurikuluma. Primer za to su klasne strukture koje reprezentuju kurseve iz oblasti opštih znanja (pre svega matematike) i opštih obrazovnih i pedagoških znanja nastavničkog kurikuluma. Budući da su ovi kursevi u odabranom nastavničkom kurikulumu veoma detaljno opisani, ove klasne strukture sadrže veći broj klasa od odgovarajućih klasnih struktura referentnog modela, odnosno opšta znanja reprezentovana sa jednom ili više (N) klasa referentnog modela kurikuluma su reprezentovana sa više (M) klasa modela odabranog nastavničkog kurikuluma ($M > N$). Ovi rezultati ukazuju na potrebu uniformnijeg definisanja opisa kurseva i/ili razmatranje uvođenje dodatnih slučajeva za 1:N algoritam, ili novog M:N algoritma.

Takođe, *Skills* potklase je potrebno usaglasiti u slučajevima gde one reprezentuju ishode različitih nivoa Blumove taksonomije, što se posebno odnosi na veštine programiranja.

Iz rezultata usaglašavanja se takođe vidi da se nazivi klasa sadržanih u modelu referentnog nastavničkog kurikuluma odnose na opšte principe, uglavnom bez navođenja konkretne tehnologije,

specifičnog programskog jezika određene programske paradigme ili softverske platforme vezane za specifičnog proizvođača. S druge strane, odabrani nastavnički kurikulum predviđa sadržaje (klase) poput: okruženje za rad sa Oracle bazama podataka, JSP/servlet tehnologija, JavaScript, PHP, Java, programski jezik C, Joomla E-Commerce itd. Tako je, na primer, klasa modela nastavničkog kurikuluma koja se odnosi na okruženje za rad sa Oracle bazom podataka ostala neuparena. Ipak, ukoliko neke od navedenih informacija postoje u referentnom modelu (poput, na primer, konkretnih programskih jezika za dinamično generisanje web sadržaja), one su mapirane na labele klasa. Budući da terminološki uparivač poredi međusobno i lokalno ime jedne klase sa labelom druge klase, ove klase mogu biti uparene. Stoga jedno razrešenje opisane nekonzistentnosti ontoloških modela bi moglo biti usvajanje jedinstvenog pristupa tako što bi ta vrsta informacija bila mapirana ili na labele ili na posebne potklase. Takođe bi se moglo razmotriti i mapiranje ovih informacija na instance klasa, što bi dalje podrazumevalo primenu ekstenzionih algoritama.

Uočljivo je da su rezultati poređenja modela nastavničkih kurikuluma saglasni sa rezultatima primene sistema na ostale moguće parove ontoloških reprezentacija kurikuluma. Na primer, sistem je, pri poređenju odabranog nastavničkog kurikuluma i ACM K12 modela, pokazao da se koncepti veštačke inteligencije ne izučavaju u nastavničkom kurikulumu u Republici Srbiji. S druge strane, pri poređenju ACM K12 modela i referentnog modela nastavničkog kurikuluma, klasne strukture odgovarajuće ovim konceptima su međusobno uparene. Iz ovih rezultata se može pretpostaviti da bi ispravan rezultat poređenja modela nastavničkih kurikuluma bio neuparenost klase referentnog modela koja reprezentuje principe veštačke inteligencije, što je i dobijeno kao rezultat primene sistema.

5.7. Diskusija i budući rad

Razvijena softverska platforma, zasnovana na tehnologijama Semantičkog Weba, omogućava utvrđivanje usaglašenosti kompetencija (znanja i veština) srednjoškolskog informatičkog kurikuluma i nastavničkih kurikuluma, kao i nastavničkih informatičkih kurikuluma međusobno.

Zbog mogućnosti heterogenog načina određivanja naziva i strukture kompetencija kurikuluma, sistem je kreiran kao polu-automatski. Preliminarna evaluacija sistema, realizovana upoređivanjem dobijenih rezultata sa referentnim usaglašavanjem definisanim od strane eksperata, upućuje na zaključak da je primenom platforme moguće korektno utvrditi usaglašenost poređenih modela. Ipak, analiza dobijenih rezultata ukazuje da neki od parova klasa dobijeni u određenim fazama ne odslikavaju realnu sličnost između ekvivalentnih pojmova u kurikulumima, što opravdava **uvodenje manualnih intervencija** da oni ne bi negativno uticali na rezultate sledećih faza uparivanja.

Analiza izlaza sistema i svih kombinacija ulaznih kurikuluma, opisana u prethodnim sekcijama (5.4. - 5.6.), je pokazala sledeće.

- Model referentnog nastavničkog kurikuluma je zadovoljavajuće usaglašen sa modelom zasnovanim na ACM K12 standardu.
- Model referentnog nastavničkog kurikuluma sadrži *Skills* potklase koje reprezentuju opštiji opis ishoda nego odgovarajuće potklase srednjoškolskog modela. To je posledica zasnovanosti *Skills* potklasa referentnog nastavničkog modela pre svega na NCATE/ISTE preporukama (East et al., 2011; ISTE; 2011) koje, za određene oblasti informatičkog domena, sadrže manje ishoda (i opštije ishode) nego ACM K12 predlog standarda.
- Analizirani nastavnički kurikulum iz Republike Srbije ne predviđa izučavanje značajnih tematskih oblasti i ishoda srednjoškolskog nivoa ACM K12 standarda. Takođe, nije u potpunosti usaglašen sa reprezentativnim svetskim kurikulumima (predstavljenim referentnim ontološkim modelom).
- Deo klasa srednjoškolskog modela je ostao neuparen kao posledica različite taksonomijske strukture odgovarajućih klasa srednjoškolskog i nastavničkih modela. To se posebno odnosi na klase koje predstavljaju sadržaje teme 'Veze između matematike i računarskih nauka', koje su neretko u modelima nastavničkih kurikuluma mapirane na klase koje odgovaraju nedomenskim znanjima.
- Deo *Skills* potklasa nastavničkih modela ne predstavlja odgovarajući nivo revidirane Blumove taksonomije. Tako su u izabranom nastavničkom kurikulumu ishodi koji se odnose na veštinu programiranja, pisanja programskog koda i sl. formulisani tako da predstavljaju niži nivo kognitivne dimenzije revidirane Blumove taksonomije od odgovarajućih ishoda ACM K12 modela.
- Vrednosti sličnosti, kao i adekvatnost uparivanja su niži kod klasa u koje su modelovani ishodi/veštine tematskih oblasti ili kurseva, što je posledica slabe standardizacije ishoda (opisi veština u kurikulumima su, po pravilu, u formi obimnog slobodnog teksta).

Na osnovu rezultata prikazanih u ovom poglavlju, po autoru ovog rada, moguće je dati sledeće preporuke:

- Model referentnog nastavničkog kurikuluma je potrebno unaprediti tako da reprezentuje sva znanja predviđena ACM K12 standardom poput sledećih nedostajućih znanja: faze mašinskog ciklusa, tehnike dokumentovanja (softvera), robotika, elementi korisniku prilagođenog web sajta, principi pretraživanja Interneta i ocenjivanja web sajta.
- Potrebno je poboljšati model referentnog nastavničkog kurikuluma tako da podrazumeva detaljnost opisa veština

odgovarajući detaljnosti opisa ciljeva učenja opisanih u temama ACM K12 modela.

- Izabrani nastavnički kurikulum je neophodno unaprediti kreiranjem novih kurseva i/ili unapređenjem postojećih tako da sadrži znanja i ishode iz oblasti ograničenja računara, principa veštačke inteligencije, kompjuterske etike i bezbednosti, rešavanja problema, principa kreiranja korisniku prilagođenog interfejsa, kao i da predviđa detaljnije izučavanje sadržaja iz oblasti algoritama.
- Zbog neusaglašenosti taksonomijskih struktura teme koje se odnose na vezu između matematike i računarskih nauka je potrebno mapirati na odgovarajuću potklasu informatičkih domenskih znanja. U slučaju izabranog nastavničkog kurikulumu to znači da te teme treba da se izučavaju u okviru kursa 'Matematika informatike'.
- Neophodno je redefinisati ishode nastavničkih kurikulumu koji predstavljaju niži nivo revidirane Blumove taksonomije od odgovarajućih srednjoškolskih ishoda. To se prvenstveno odnosi na izabrani nastavnički kurikulum koji je potrebno unaprediti tako da u oblasti programiranja predviđa ishode koji odgovaraju najvišem nivou kognitivne dimenzije revidirane Blumove taksonomije.
- Preciznijem uparivanju klasa koje reprezentuju veštine kurikulumu bi doprinela i veća struktuiranost ovog dela ontologije kao i standardizacija definisanja ishoda.
- Veoma je važno da nadležno ministarstvo, tela za akreditaciju i eksperti iz oblasti definišu standardizovane preporuke o principima formulisanja sadržaja i ishoda čime bi bio omogućen razvoj softvera (ili komponente softvera u okviru predstavljene platforme u ovom radu) za precizno (polu)automatsko mapiranje sadržaja kurikulumu u ontološki model.
- Poželjno je reprezentovanje aktuelnih srednjoškolskih informatičkih i nastavničkih kurikulumu primenom ontologija i principa Semantičkog Weba. Na taj način bi bilo obezbeđeno brže i jednostavnije menjanje kurikulumu i praćenje novonastalih promena. Pri tome, u cilju preciznijeg utvrđivanja usaglašenosti kurikulumu poželjno je usvajanje zajedničkog modela viših klasa i analogne relacione strukture, uz uvažavanje aktuelnih međunarodnih standarda poput korišćenja zajedničke klasifikacije pri kategorisanju tematskih oblasti (na primer, ACM Computing klasifikacije (ACM, 2012)).
- Potrebno je da kreatori kurikulumu vrše periodične provere usaglašenosti i inoviranje kurikulumu u skladu sa aktuelnim standardima, srednjoškolskim informatičkim kompetencijama i reprezentativnim svetskim kurikulumima.
- Poželjno je periodično proveravati kompetencije aktuelnih nastavnika i istražiti potrebu za doobukom. To bi se, u postojećem sistemu, moglo realizovati mapiranjem

kompetencija nastavnika na ontološki model (uz uvažavanje viših klasa koje reprezentuju pet generalnih oblasti kurikuluma, nivoe revidirane Blumove taksonomije itd.) i utvrđivanjem usaglašenosti kreiranog modela sa modelom aktuelnog informatičkog standarda ili sa referentnim modelom nastavničkog kurikuluma. Dobijene, nedostajuće kompetencije bi predstavljale predmet razmatranja za dodatnu obuku. Pri tome, utvrđivanje kompetencija nastavnika koja bi bila reprezentovana ontologijom bi se mogla utvrditi primenom sledećih postupaka: prikupljanjem podataka o njihovom poznavanju tema i oblasti primenom upitnika (kao što je realizovano u trećem poglavlju ove disertacije); mapiranjem sadržaja i ishoda studijskog programa koji su završili na ontološki model (uz uvažavanje dosadašnjeg dodatnog obrazovanja); samostalnim kreiranjem ontologije od strane nastavnika, itd.

Takođe, potrebno je dalje istražiti mogućnosti unapređenja razvijene softverske platforme. Stoga su u nastavku opisana ograničenja platforme i mogući pravci daljeg istraživanja.

Opisani terminološki algoritam poredi reči (tokene) samo pojedinačno ne uzimajući u obzir širi kontekst (poziciju u odnosu na druge reči, značenje drugih reči koje pripadaju istom stringu itd.). Takođe, WordNet rečnik predstavlja bazu podataka reči iz različitih oblasti (ne samo računarstva), te se, ne uzimajući u obzir pripadnost domenu, programiranje (engl. *programming*) definiše na dva načina: kao „kreiranje niza instrukcija koje omogućavaju računaru da uradi nešto“ i kao „podešavanje redosleda i vremena planiranih događaja“ (izvor WordNet Search 3.1.). Potonje tumačenje bi moglo dovesti do netačnih rezultata. Stoga bi **jedan od pravaca budućih istraživanja mogao biti unapređenje terminoloških algoritama tako da koriste „specifičan računarski rečnik“ i razmatraju sintaksu jezika**. Jedno rešenje za primenu „specifičnog računarskog rečnika“ bi moglo biti uzimanje u obzir semantičkih domena WordNet rečnika (Bentivogli et al., 2004) pri poređenju tokena. Takođe je moguće razmotriti i korišćenje drugih eksternih baza znanja poput Wikipedia-je što je primenjeno u sistemu opisanom u (Hertling & Paulheim, 2012) ili upotrebu eksterne ontologije za deo domenskih znanja poput računarske ontologije prikazane u (Cassel et al., 2008). Kao eksterna ontologija za domenska znanja može biti korišćena i ACM Computing klasifikacija (ACM, 2012) uz primenu Wu & Palmer metode za utvrđivanje sličnosti (Wu & Palmer, 1994).

Kreirani ontološki modeli sadrže termine na engleskom jeziku, dok je pri utvrđivanju lingvističke sličnosti korišćena engleska verzija WordNet rečnika. Ovo opredeljenje u disertaciji proisteklo je, pre svega, iz toga što su srednjoškolski standardi (ACM K12 na osnovu kog je kreiran srednjoškolski ontološki model) i međunarodne preporuke o nastavničkim kurikulumima (CSTA, NCATE/ISTE) napisane na engleskom jeziku. Međutim, **važan pravac daljeg rada**

predstavlja (polu)automatsko mapiranje kurikuluma napisanih na srpskom jeziku na ontološke modele i istraživanje mogućnosti njihovog poređenja sa međunarodnim standardima (implementiranjem "Inter-Lingual Index" komponente srpskog WordNet-a za sinhronizovan rad dva WordNet-a), kao i poređenja sa drugim ontološkim modelima napisanim na srpskom jeziku (implementiranjem posebnog srpskog WordNet rečnika – SrpWN, opisanog u radovima (Krstev et al., 2004, 2008)).

Jedno od budućih pravaca istraživanja bi moglo biti uvođenje novih algoritama poput ekstenzionih (u slučaju da modeli sadrže instance), M:N algoritama („više ka više“), kao i implementiranje algoritama za semantičku verifikaciju i to primenom dela metoda opisanih u (Jean-Mary et al., 2009; Ngo & Bellahsene, 2012) ili upotrebom eksternih resursa kao u (Groß et al, 2012; Lambrix & Tan, 2006). Potrebno je i istražiti prednosti i nedostatke realizovanja sekvenci algoritama u više iteracija, uz manuelnu intervenciju korisnika na kraju svake iteracije.

Još jedan pravac daljeg istraživanja je poboljšanje performansi sistema primenom nekog od postupaka za rano eliminisanje kandidata (Ehrig & Staab, 2004; Ehrig, 2007; Rahm, 2011). U (Faria et al., 2013) se navodi da je većina vrednosti u matrici sličnosti veoma niska ili jednaka nuli, te je čuvanje ovih podataka nepotrebno. Stoga se predlaže čuvanje sličnosti u strukturi podataka koja skladišti i najbolje parove klasa pri čemu se odbacuju sličnosti koje su ispod zadatog praga. Kreiranje lokalnih leksikona nakon inicijalizacije ontologija (Faria et al., 2013) umesto čestog pristupa WordNet bazi bi, takođe, moglo dovesti do unapređenja performansi sistema.

Takođe, mogući pravac daljeg istraživanja je i unapređenje evaluacije sistema poređenjem većeg skupa kurikuluma i analizom rezultata uz manuelne intervencije korisnika (eksperata) nakon svake faze.

6. Zaključak

U disertaciji je predložena softverska platforma za sinhronizaciju modela informatičkih kurikuluma zasnovana na tehnologijama Semantičkog Weba i principima ontološkog uparivanja. Softverska platforma omogućava usaglašavanje kurikuluma za nastavnike informatike međusobno, kao i kurikuluma za nastavnika informatike i srednjoškolskog informatičkog kurikuluma.

Nakon uvodnog poglavlja u kome je opisan predmet istraživanja disertacije, u drugom poglavlju je dat prikaz aktuelnog stanja u oblasti obrazovanja nastavnika informatike. Naučni doprinos u ovom delu predstavlja analiza aktuelnih standarda, preporuka međunarodnih organizacija i relevantne literature, i, posebno, analiza reprezentativnih svetskih kurikuluma i kurikuluma iz Republike Srbije. Poređenjem sadržaja kurikuluma i ACM K12 modela standarda (jednostavnim prebrojavanjem tema koje se preklapaju i izračunavanjem procentualne zastupljenosti) određeno je u kojoj meri prikazani kurikulumi predviđaju informatičke kompetencije usklađene sa srednjoškolskim nivoom ACM K12 standarda. Takođe je utvrđena zastupljenost neinformatičkih znanja (opštih obrazovnih i pedagoških znanja, domensko metodičko-didaktičkih znanja, znanja nastavne prakse i opštih znanja) u kurikulumima za nastavnike informatike. Na osnovu sprovedene analize izvedeni su zaključci o generalnim oblastima koje svaki kurikulum za nastavnike informatike treba da sadrži, o poželjnim modalitetima obrazovanja nastavnika informatike i strukturi informatičkog domena u kurikulumima. Takođe, analiza je pokazala da je većinu kurikuluma potrebno unaprediti da bi bili usklađeni sa ACM-ovim modelom. Ograničenje analize, predstavljene u ovom poglavlju, predstavlja tehnika upoređivanja kurikuluma. Stoga je za tu svrhu, dalje, u disertaciji primenjeno poređenje mašinski razumljivih reprezentacija kurikuluma primenom poluautomatskog softvera.

Da bi se utvrdilo aktuelno stanje u oblasti nastave informatike u srednjim školama i dodatno sagledalo da li nastavnički kurikulumi studentima obezbeđuju adekvatne kompetencije za držanje nastave informatike u srednjoj školi, ispitanici su stavovi zaposlenih nastavnika informatike u srednjim školama. Naučni doprinos predstavlja istraživanje na reprezentativnom uzorku nastavnika informatike u Vojvodini o njihovom formalnom i dodatnom obrazovanju, stavovima o poznavanju predloženih tema nastavničkog kurikuluma, značaju tema za realizovanje nastave i zastupljenosti u srednjoškolskim informatičkim kurikulumima. Rezultati ukazuju na neujednačenost u akademskom obrazovanju nastavnika, ocenu nastavnika da nedovoljno poznaju kompleksnije i savremenije aspekte informatike, da im nedostaju znanja iz pedagoško-metodičkog korpusa i da su sve tematske oblasti informatičkog kurikuluma predložene upitnikom značajne za realizovanje nastave, ali da mnoge od njih nisu adekvatno zastupljene u nastavnim planovima i programima srednjih škola. Istraživanje je takođe ukazalo na posebne integrisane studije kao preferirani modalitet obrazovanja nastavnika informatike. Nastavnici

su većinom istakli potrebu uvida i uticaja na kurikulume svih nivoa obrazovanja primenom softvera koji bi omogućio i utvrđivanje usaglašenosti kompetencija.

U četvrtom poglavlju predložen je ontološki model srednjoškolskog informatičkog kurikuluma, zasnovan na ACM K12 standardu, referentni ontološki model za nastavnike informatike, konstruisan na bazi reprezentativnih nastavničkih kurikuluma, i ontološki model konkretnog nastavničkog kurikuluma kreiran na osnovu odabranog studijskog programa iz Republike Srbije. Originalni ontološki modeli sva tri kurikuluma imaju isti - najviši nivo modela kompetencija (zasnovan na IEEE RCD standardu) i modelovani su, u skladu sa brojnim preporukama iz aktuelne literature, istim klasama (klase *Knowledge* i *Skills*) i istom pripadajućom relacionom strukturom. Cilj kreiranja modela bio je definisanje mašinski čitljivih reprezentacija kurikuluma, tako da potencijalnim korisnicima bude obezbeđen uvid u kurikulume, izmene kurikuluma, praćenje promena kurikuluma, deljenje kurikuluma i provera semantičke konzistentnosti. Takođe, važno je bilo predstaviti kurikulum tako da bude omogućeno njegovo importovanje u softversku platformu koja bi obezbedila utvrđivanje usaglašenosti kurikuluma. Osnovni naučni doprinos ovog poglavlja je ontološka reprezentacija standardizovanog srednjoškolskog kurikuluma informatike. Prednosti su standardizovanost kurikuluma (ACM K12) koji obezbeđuje dobru osnovu za evoluciju ka standardizovanom nastavničkom kurikulumu i mašinski čitljiva reprezentacija koja doprinosi lakšem praćenju promena u oblasti i omogućava da različite ciljne grupe (na primer, akademska zajednica, nadležna ministarstva, studenti, sadašnji zaposleni u informatičkom sektoru) budu aktivno uključene u održavanje i unapređivanje kurikuluma. Nedostatom se može smatrati pojednostavljenost modela. Ona se ogleda u predstavljanju ciljeva u modelu, izostanak preduslova za izučavanje određene tematske oblasti i aspekata kurikuluma koji nisu usko vezani sa kompetencijama, poput instrukcionih metoda, načina ocenjivanja, objekata učenja i slično. Ovaj nedostatak se može, bez narušavanja postojećeg modela, otkloniti obogaćivanjem ontološkog modela. Ovde je od posebne važnosti tranzitivna objektna relacija "preduslov" koja bi omogućila mapiranje informacija o preduslovima za izučavanje određene tematske oblasti ili kursa. Pretpostavka za ovo proširenje je, pre svega, dopunjavanje postojećih opisa kurikuluma informacijama koje bi omogućile preciznije mapiranje u ontološki model. U postojećim kurikulumima, ova veza je najčešće implicitno definisana kroz godinu/nivo studija na kojima se određeni kurs/tematska oblast izučava, što omogućuje da se uspostave preliminarne veze koje bi kasnije manuelnim intervencijama ili metodama mašinskog učenja iz sadržaja kurseva bile rafinirane. Model se može proširiti i alternativama za reprezentaciju ciljeva, odnosno ishoda dodavanjem novih taksonomija, pored Blumove.

U poslednjem, petom, poglavlju opisan je razvijeni prototip softverskog alata oslonjen na standardne algoritme za uparivanje ontologija, koji omogućava poluautomatizovano usaglašavanje

informatičkih kurikuluma različitih nivoa obrazovanja. Predloženi postupak za utvrđivanje usaglašenosti uvažava navedeni model kompetencija i relacionu strukturu zajedničku za definisane ontološke reprezentacije kurikuluma. Platforma obezbeđuje korisnicima (ekspertima iz oblasti, univerzitetima, nadležnom ministarstvu) informacije o tome kako da unaprede kompetencije nastavnčkog kurikuluma (definišu nove sadržaje i ishode, izmene postojeće, promene strukturu kurseva) tako da budu usaglašene sa predviđenim kompetencijama srednjoškolskog informatičkog kurikuluma ili drugog nastavnčkog modela. Korišćenjem softvera verifikovan je model platforme ispitivanjem usaglašenosti kreiranih ontoloških modela kurikuluma i analizirani su dobijeni rezultati. Takođe, izvršeno je upoređivanje rezultata sa referentnim usaglašavanjem definisanim od strane eksperata. Osnovni zaključak je sledeći: primenom platforme moguće je u visokoj meri utvrditi usaglašenost poređenih kurikuluma. Dobijeni su rezultati koji se mogu direktno iskoristiti za unapređivanje nastavnčkih kurikuluma. To su rezultati koji ukazuju na neusaglašenosti između ACM K12 predloga kurikuluma i nastavnčkih kurikuluma (neke tematske oblasti ACM K12 predloga nisu obuhvaćene modelima nastavnčkih kurikuluma; nastavnčki kurikulumi ne obezbeđuju sve veštine potrebne za realizaciju nastave u skladu sa ACM K12 predlogom; pojedine veštine u nastavnčkom kurikulumu ne predstavljaju odgovarajuću kategoriju dimenzije kognitivnih procesa revidirane Blumove taksonomije) i rezultati koji ukazuju da modeli nastavnčkih kurikuluma nisu međusobno zadovoljavajuće usaglašeni u pogledu sadržaja oblasti (kurseva), njihovih ishoda, strukture i nivoa predviđenih veština. Naučni doprinos ovog poglavlja predstavlja, postupak za upoređivanje srednjoškolskog informatičkog i nastavnčkog kurikuluma i nastavnčkih kurikuluma međusobno. Ovde su prednosti zasnovanost postupka na opštim principima uparivanja ontologija i mogućnost manuelne intervencije čime je obezbeđeno integrisanje znanja eksperata u modele kurikuluma. Nedostaci su spregnutost postupka sa ontološkim modelom, performanse i nešto niže vrednosti sličnosti (i tačnosti uparivanja) pri poređenju ishoda kurikuluma. Prvi nedostatak se može otkloniti uvođenjem drugih tipova algoritama i/ili modifikacijom postojećih u skladu sa modifikacijom ontološkog modela. Jedan od načina za poboljšanje performansi je primena nekog od postupaka za rano eliminisanje kandidata. Potrebno je razmotriti i primenu srpske verzije WordNet rečnika, eksternih računarskih rečnika ili ontologija, primenu algoritama za semantičku verifikaciju i izvršavanje algoritama u više iteracija. Arhitektura sistema je takva da se postupak može jednostavno proširiti drugim tipovima algoritama u slučaju potrebe. Za dobijanje tačnijih rezultata, posebno u delu koji se odnosi na veštine, potrebna je veća standardizacija ove oblasti i bolja struktuiranost.

Literatura

- [1] Anderson, L.W., Krathwohl, D.R., Airasian, P.W., Cruikshank, K.A., Mayer, R.E., Pintrich, P.R., Raths, J. & Wittrock, M.C. (2001). *A taxonomy for Learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. New York: Addison Wesley Longman.
- [2] Anohina-Naumeca, A., Graudina, V. & Grundspenķis, J. (2012). Curricula comparison using concept maps and ontologies. *International Scientific Conference: Applied Information and Communication Technologies*, 5, Jelgava (Latvia), pp.177-183.
- [3] Armoni, M. (2011). Looking at secondary teacher preparation through the lens of computer science. *ACM Transactions on Computing Education*, 11(4), pp. 1-38.
- [4] Association for Computing Machinery (2008). *ACM Computer Science Curriculum 2008: An Interim Revision of CS 2001 CS curriculum*, IEEE Computer Society. <http://www.acm.org/education/curricula/ComputerScience2008.pdf>
- [5] Association for Computing Machinery (2012). The 2012 ACM Computing Classification System. <http://www.acm.org/about/class/2012>
- [6] Bentivogli, L. , Forner, P., Magnini, B. & Pianta, E. (2004). Revising WordNet Domains Hierarchy: Semantics, Coverage, and Balancing. *Proc. Workshop Multilingual Linguistic Resources (MLR '04)*, pp. 101-108.
- [7] Berners-Lee, T., Hall, W., Hendler, J.A., O'Hara, K., Shadbolt, N. & Weitzner, D.J. (2006). A Framework for Web Science. *Foundations and Trends in Web Science*, 1(1), pp. 1-130.
- [8] Brewster, C., O'Hara, K., Fuller, S., Wilks, Y., Franconi, E., Musen, M.A., Ellman, J. & Shum, S. (2004). Knowledge Representation with Ontologies: The Present and Future. *IEEE Intelligent Systems*, 19(1), pp. 72–81.
- [9] Cassel, L. N., Davies, G., LeBlanc, R., Snyder, L. & Topi, H. (2008). Using a Computing Ontology as a Foundation for Curriculum Development. In *SWEL 2008: Sixth International Workshop on Ontologies and Semantic Web for E-Learning in conjunction with ITS 2008: Ninth International Conference on Intelligent Tutoring Systems*, pp. 21-30.
- [10] Chi, Y.L. (2009). Ontology-based curriculum content sequencing system with semantic rules. *Expert Systems with Applications*, 36(4), pp. 7838–7847.
- [11] Churches, A. (2007). *Bloom's Digital Taxonomy* <http://www.techlearning.com/techlearning/archives/2008/04/andrewchurches.pdf>
- [12] College of Science and Technology (2013). Computer Science, BS — Grades 8-12. http://catalog.tamucc.edu/preview_program.php?catoid=5&pooid=362&returnto=159
- [13] Commission of the European Communities (2005). Proposal for a RECOMMENDATION OF THE EUROPEAN

PARLIAMENT AND OF THE COUNCIL on key competences for lifelong learning,
[http://www.europarl.europa.eu/meetdocs/2004_2009/documents/com/com_com\(2005\)0548_/com_com\(2005\)0548_en.pdf](http://www.europarl.europa.eu/meetdocs/2004_2009/documents/com/com_com(2005)0548_/com_com(2005)0548_en.pdf)

- [14] Cruz, I.F., Antonelli, F. P. & Stroe, C. (2009). Agreementmaker: Efficient matching for large real-world schemas and ontologies, *PVLDB*, 2(2), pp. 1586–1589.
- [15] Cruz, I.F., Sunna, W., Makar, N. & Bathala, S. (2007). A Visual Tool for Ontology Alignment to Enable Geospatial Interoperability. *Journal of Visual Languages and Computing*, 18(3), pp. 230-254.
- [16] Cruz, I.F. & Sunna, W. (2008). Structural Alignment Methods with Applications to Geospatial Ontologies. *Transactions in GIS* 12(6), pp. 683–711.
- [17] CSTA Standards Task Force (2011). K–12 Computer Science Standards, 2011, http://csta.acm.org/Curriculum/sub/CurrFiles/CSTA_K-12_CSS.pdf
- [18] Daconta, M.C., Obrst, L.J. & Smith, K.T. (2003). *The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management*. Indianapolis: John Willey & Sons.
- [19] Davies, J., Fensel, D. & van Harmelen, F. (2002). *Towards the Semantic Web: Ontology-driven Knowledge Management*. Chichester: John Wiley and Sons.
- [20] De Coi, J., Herder, E., Koesling, A., Lofi, C., Olmedilla, D., Papatreou, O. & Siberski, W. (2007). A Model for Competence Gap Analysis. *Proceedings of 3rd International Conference on Web Information Systems and Technologies (WEBIST)*, Barcelona, Spain.
- [21] Demartini, G., Enchev, I., Gapany, J. & Cudré-Mauroux, P. (2013). The Bowlogna Ontology: Fostering Open Curricula and Agile Knowledge Bases for Europe’s Higher Education Landscape. *Semantic Web - Interoperability, Usability, Applicability*, 4(1), pp. 53-63.
- [22] Dexter, H. & Davies, I. (2009). An ontology-based curriculum knowledgebase for managing complexity and change. *Ninth IEEE International Conference on Advanced Learning Technologies*, ICALT, pp.136-140.
- [23] Doan, B.-L. & Bourda, Y. (2006). An Educational System Based on Several Ontologies. *Proceedings of the Sixth International Conference on Advanced Learning Technologies 2006 (ICALT’06)*, pp. 176-183.
- [24] Donaldson, G. (2011). Teaching Scotland’s Future – Report of a review of teacher education in Scotland. <http://scotland.gov.uk/Publications/2011/01/13092132/0>
- [25] Dreyfus, P. (1962). *L’informatique*. Gestion, Paris, pp. 240-241.
- [26] Državni Univerzitet Novi Pazar (2013). *Studijski program Informatika i matematika*. <http://www.np.ac.rs/index.php/ynu/stud-prog-mat/inf-mat-oas>

- [27] East, P., Bentley, C., Kmoch, J., Rainwater, S. & Stephenson, C. (2011). NCATE standards for preparation of secondary computer science teachers. *Proceedings of the 42nd ACM technical symposium on Computer science education*, pp. 243-244.
- [28] Ehrig, M. & Staab, S. (2004). QOM - Quick ontology mapping. In *Proc. 3rd International Semantic Web Conference (ISWC04)*, pp. 683-697.
- [29] Ehrig, M. (2007). *Ontology alignment: bridging the semantic gap*. Springer, New-York (NY US), 2007.
- [30] Elsayed, A. (2009). Interaction with Content through the Curriculum Lifecycle. *Advanced Learning Technologies, ICAIT 2009*, pp. 730 – 731.
- [31] Ericson, B., Armoni, M., Gal-Ezer, J., Seehorn, D., Stephenson, C. & Trees, F. (2008). Ensuring Exemplary Teaching in an Essential Discipline: Addressing the Crisis in Computer Science Teacher Certification, *Final Report of the CSTA Teacher Certification TaskForce*. <http://www.csta.acm.org/Communications/sub/DocsPresentationFiles/CertificationFinal.pdf>
- [32] Eurydice (2004). *Key Data on Information and Communication Technology in Schools in Europe*. The information network on education in Europe, Belgium http://eacea.ec.europa.eu/education/eurydice/documents/key_data_series/048EN.pdf
- [33] Euzenat, J. (2007). Semantic precision and recall for ontology alignment evaluation. In: *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, Hyderabad, IN, pp. 348–353
- [34] Euzenat, J. & Shvaiko, P. (2007). *Ontology Matching*. Springer-Verlag, Berlin-Heidelberg, 333 p.
- [35] Fakultet tehničkih nauka (2013). *Integrirane akademske studije tehnika i informatika*. http://www.ftn.kg.ac.rs/akreditacijaFTN/index.php?sp=IS_TI
- [36] Fakultät für Informatik (2013). Studienpläne. <http://www.cs.tu-dortmund.de/nps/de/Studium/Studienplaene/PDFs/index.html>
- [37] Fakultät für Elektrotechnik, Informatik und Mathematik (2013). BA/MA Studiengänge Lehramt Informatik. <http://ddi.uni-paderborn.de/studium.html>
- [38] Fakultät für Wirtschaftswissenschaften (2012). Verkündungsblatt. https://www.uni-due.de/verwaltung/verkuendungsblatt/jahrgang10_2012.php.
- [39] Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I.F. & Couto, F.M. (2013). The agreementmakerlight ontology matching system. In: Meersman, R., Panetto, H., Dillon, T., Eder, J., Bellahsene, Z., Ritter, N., De Leenheer, P., Dou, D. (eds.) *ODBASE 2013. LNCS*, vol. 8185, pp. 527–541. Springer, Heidelberg.

- [40] Fellbaum, C. (2006). WordNet and wordnets. In: Brown, Keith et al. (eds.), *Encyclopedia of Language and Linguistics*, 13, pp. 665–670.
- [41] Fernández-Breis, J.T., Castellanos-Nieves, D., Hernández-Franco, J., Soler-Segovia, C., del Carmen Robles-Redondo, M., González-Martínez, R. & Prendes-Espinosa, M. P. (2012). A semantic platform for the management of the educative curriculum. *Expert Systems with Applications*, 39(5), pp. 6011-6019.
- [42] Fleishman, E. A., Wetrogan, L. I., Uhlman, C. E. & Marshall-Mies, J. C. (1995). Abilities In Peterson, N. G., Mumford, M. D., Borman, W. C., Jeanneret, P. R. & Fleishman, E. A. (Eds.), *Development of prototype occupational information network content model*, Salt Lake City, UT: Utah Department of Employment Security, vol. 1, p. 1086.
- [43] Frost, D., Verno, A., Buckhart, D., Hutton, M., & North, K. (2009). A Model Curriculum for K-12 Computer Science: Level I Objectives and Outlines. <http://esta.acm.org/Curriculum/sub/CurrFiles/L1-Objectives-and-Outlines.pdf>.
- [44] Fuller, U., Johnson, C. G., Ahoniemi, T., Cukierman, D., Hern, I., Hernán-Losada, I., et al. (2007). Developing a computer science-specific learning taxonomy. *ACM SIGCSE Bulletin*, 39(4), pp. 152-170.
- [45] Gal-Ezer, J. (1995). Computer science teachers' certification program. *Comput. Educ.* 25(3), pp. 163–168.
- [46] Gal-Ezer, J. & Stephenson, C. (2010). Computer science teacher preparation is critical. *ACM Inroads*, 1(1), pp. 61-66.
- [47] Gal-Ezer, J. & Zur, E. (2007) Reaching out to CS teachers: Certification via distance learning. *Math.Comput. Educ.* 41(3), pp. 250–265.
- [48] Gluga, R., Kay, J. & Lever, T. (2013). Foundations for modeling university curricula in terms of multiple learning goal sets. *IEEE Transactions on Learning Technologies*, 6(1), pp. 25-37.
- [49] Goethe-Universität (2013). Informatik, Lehramt an Haupt- und Realschulen (L2). <http://www2.uni-frankfurt.de/35791365>
- [50] Grau, B.C., Dragisic, Z., Eckert, K., Euzenat, J., et al. (2013). Results of the ontology alignment evaluation initiative 2013. In: *Proc. 8th ISWC workshop on ontology matching (OM)*, pp. 61–100.
- [51] Grgurina, N. (2008). Computer Science Teacher Training at the University of Groningen. *Lecture Notes in Computer Science*, 5090, ISSEP 2008, pp. 272 - 281.
- [52] Groß, A., Hartung, M., Kirsten, T. & Rahm, E. (2012). GOMMA results for OAEI 2012. In: *Ontology Matching Workshop, International Semantic Web Conference 2012*, pp. 133-140.
- [53] Grgurina, N. & Tolboom, J. (2008). The first decade of informatics in Dutch high schools. *Informatics in Education*, 7(1), pp. 55-74.

- [54] Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human and Computer studies*, 43(5/6), pp. 907-928.
- [55] Guarino, N. ed. (1998). Formal Ontology in Information Systems. *Proceedings of the First International Conference (FOIS '98)*, pp. 3-15.
- [56] Harchay, A., Cheniti-Belcadhi, L. & Braham, R. (2012). A Model Driven Infrastructure for Context-Awareness Mobile Assessment Personalization. *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on*, pp. 1676 - 1683
- [57] Hazzan, O., Gal-Ezer, J. & Blum, L. (2008). A model for high school computer science education: the four key elements that make it! In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education Symposium (ITiCSE'08)*. *SIGCSE Bull.* 40(1), pp. 281–285.
- [58] Hazzan, O., Gal-Ezer, J. & Ragonis, N. (2010). How to establish a Computer Science teacher preparation program at your university? – The ECSTPP Workshop. *ACM Inroads magazine*, 1(1), pp. 35-39
- [59] Hazzan, O. & Lapidot, T. (2004). The practicum in computer science education: bridging gaps between theoretical knowledge and actual performance, *ACM SIGCSE Bulletin*, 36(4), pp. 47-51.
- [60] Hazzan, O. & Lapidot, T. (2006). Social issues of Computer Science in the "Methods of Teaching Computer Science in the High School" course, *ACM SIGCSE Bulletin*, 38(2), pp. 72-75.
- [61] Heer, R (2012). *A Model of Learning Objectives–based on A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Center for Excellence in Learning and Teaching, Iowa State University, 2012. <http://www.celt.iastate.edu/teaching-resources/effective-practice/revise-blooms-taxonomy/>
- [62] Hertling, S. & Paulheim, H. (2012). WikiMatch: Using wikipedia for ontology matching. In *Proceedings of the 7th International Workshop on Ontology Matching*, pp. 37–48.
- [63] Hsu, I.-C. (2012). Intelligent Discovery for Learning Objects Using Semantic Web Technologies. *Educational Technology & Society*, 15(1), pp. 298–312.
- [64] Hu, W., Qu, Y. & Cheng, G. (2008). Matching large ontologies: A divide-and-conquer approach. *Data and Knowledge Engineering*, 67(1), pp. 140–160.
- [65] Huber, J., Sztyler, T., Noessner, J. & Meilicke, C. (2011). CODI: Combinatorial Optimization for Data Integration : results for OAEI 2011. In Proc. 6th ISWC workshop on Ontology Matching (OM), pp.134–141.
- [66] Hubwiser, P. (2009). *Didaktik der Informatik*. Berlin Heidelberg New York: Springer
- [67] International Society for Technology in Education (ISTE) (2002). *Educational Computing and Technology Standards for*

Secondary Computer Science Education Initial Endorsement Program

http://www.iste.org/docs/pdfs/ncate_iste_csed_2002.pdf?sfvrsn=2

- [68] International Society for Technology in Education (ISTE) (2011). ISTE Standards Computer Science Educators-
http://www.iste.org/docs/pdfs/20-14_ISTE_Standards-CSE_PDF.pdf
- [69] IMS Global Learning Consortium (2002). IMS Reusable Definition of Competency or Educational Objective Specification, <http://www.imsglobal.org/competencies/>
- [70] Isaac, A., Wang, S., Zinn, C., Matthezing, H., van der Meij, L. & Schlobach, S. (2009). Evaluating thesaurus alignments for semantic interoperability in the library domain, *IEEE Intelligent Systems*, 24(2), pp. 76–86.
- [71] Jaro, M. (1976). UNIMATCH: A record linkage system: User's manual. Technical report, U.S. Bureau of the Census, Washington (DC US).
- [72] Jaro, M. (1989). Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida, *Journal of the American Statistical Association*, 84(406), pp. 414–420.
- [73] Jean-Mary, Y. R., Shironoshita, E. P. & Kabuka, M. R. (2009). Ontology matching with semantic verification. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3), pp. 235–251.
- [74] Jiang, J.J. & Conrath, D.W. (1998). Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings of International Conference on Research in Computational Linguistics. ROCLING X, Taiwan, pp. 19-33.
- [75] Jiménez-Ruiz, E. & Cuenca Grau, B. (2011). LogMap: Logic-based and Scalable Ontology Matching. In: Int'l Sem. Web Conf. (ISWC). pp. 273–288
- [76] Kizilkaya, G., Torun, E.D. & Askar, P. (2007). Restructuring E-learning With Ontologies, *Fifth International Conference on Computational Science and Applications, ICCSA 2007*, Kuala Lumpur, pp. 161 – 164
- [77] Knight, C., Gašević, D., & Richards, G. (2006). An Ontology-Based Framework for Bridging Learning Design and Learning Content. *Educational Technology & Society*, 9(1), pp. 23-37.
- [78] Krathwohl, D. R. (2002). A revision of bloom's taxonomy: An overview. *Theory into Practice*, 41(4), pp. 212-218.
- [79] Krstev, C., Pavlović-Lažetić, G., Vitas, D. & Obradović, I. (2004). Using Textual and Lexical Resources in Developing Serbian Wordnet. *Romanian Journal of Information Science and Technology*, 7(1-2), pp. 147-161.
- [80] Krstev, C., Djorević, B., Antonić, S., Ivković-Berček, N., Zorica, Z., Crnogorac, V. & Macura, Lj. (2008). Cooperative Work in Further Development of Serbian Wordnet. *INFOTHECA: Journal of Informatics and Librarianship*, IX(1-2), pp. 59-78.
- [81] Lambrix P. & Tan H. (2006). SAMBO - A System for Aligning and Merging Biomedical Ontologies. *Journal of Web Semantics: Science,*

- Services and Agents on the World Wide Web, 4(3), pp. 196-206.
- [82] Lapidot, T. & Hazzan, O. (2003). Methods of teaching a computer science course for prospective teachers, *ACM SIGCSE Bulletin*, 35(4), pp. 29 – 34.
- [83] Lapidot, T. & Hazzan, O. (2005). Song debugging: merging content and pedagogy in computer science education, *ACM SIGCSE Bulletin*, 37(4), pp. 79-83.
- [84] Leacock, C. & Chodorow, M. (1998). Combining local context and wordnet similarity for word sense identification. In Fellbaum, C., ed.: *WordNet: An electronic lexical database*. MIT Press, pp. 265–283.
- [85] Learning Technology Standards Committee – Institute of Electrical Engineering (LTSC-IEEE) (2002). *Draft Standard for Learning Object Metadata*. IEEE Standard 1484.12.1 (PDF), New York: Institute of Electrical and Electronics Engineers http://ieeeltsc.org/wg12LOM/1484.12.3/Public/IEEE_1484_12_03_d8.pdf
- [86] Learning Technology Standards Committee of the IEEE Computer Society (2006). Proposed Draft Standard for Learning Technology—Simple Reusable Competency Map, <http://www.ostyn.com/standardswork/competency/ReusableCompMapProp.pdf>
- [87] Learning Technology Standards Committee of the IEEE Computer Society (2008). IEEE Standard for Learning Technology—Data Model for Reusable Competency Definitions, <http://www.doleta.gov/usworkforce/pdf/2007-ieeecomp.pdf>
- [88] Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Doklady akademii nauk SSSR*, 163(4), pp. 845–848.
- [89] Li, J., Tang, J., Li, Y. & Luo, Q. (2009). RiMOM: A Dynamic Multistrategy Ontology Alignment Framework. *IEEE Transactions on Knowledge and Data Engineering*, 21(8), pp.1218-1232.
- [90] Libbrecht, P., Desmoulins, C., Mercat, C., Laborde, C., Dietrich, M. & Hendriks, M. (2008). Cross-curriculum search for intergeo. In: Autexier, S., Campbell, J., Rubio, J., Sorge, V., Suzuki, M. & Wiedijk, F. (Eds.), *Intelligent computer mathematics*, *Lecture Notes in Computer Science*, Vol. 5144/2008, pp. 520–535.
- [91] Lin, D. (1998). An information-theoretic definition of similarity. *Proceedings of the 15th International Conf. on Machine Learning*, pp. 296–304.
- [92] Lin, F. & Sandkuhl, K. (2008). A survey of exploiting wordnet in ontology matching. in *IFIP International Federation for Information Processing*, 276, Artificial Intelligence and Practice II, pp. 341–350.
- [93] Lopez, V., Fernández, M., Motta, E. & Stieler, N. (2012). PowerAqua: Supporting users in querying and exploring the Semantic Web. *Semantic Web*, 3(3), pp. 249–265

- [94] Lundqvist, K., Baker, K. & Williams, S. (2011). Ontology Supported Competency System. *International Journal of Knowledge and Learning*, 7(3-4), pp. 197-219.
- [95] Madden, B., Verno, A., Carter, D., Cooper, S., Cortina, T., Cudworth, R., Ericson, B. & Parys, E. (2007). A Model Curriculum for K-12 Computer Science: Level III Objectives and Outlines. <http://csta.acm.org/Curriculum/sub/CurrFiles/L3-Objectives-and-Outlines.pdf>
- [96] Maedche, A. & Staab, S. (2002). Measuring similarity between ontologies. *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, pp. 251–263.
- [97] Mandić, M., Konjović, Z. & Ivanović, M. (2015). Ontological Model of the Standardized Secondary School Curriculum in Informatics. *Proceedings of the 5th International Conference on Information Society and Technology*, pp. 363-367.
- [98] Mandić, M., Konjović, Z. & Viđikant, P. (2014). The profile of high school informatics teachers in the autonomous province of Vojvodina. *Croatian Journal of Education*, 16 (3), pp. 779 – 814.
- [99] Mandić, M., Segedinac, M., Savić, G. & Konjović, Z. (2013). IEEE RCD standard based ontological modeling of Computer Science curriculum. *Proceedings of the 3th International Conference on Information Society and Technology*, pp. 189-285.
- [100] Matematički fakultet (2013). *Profesor matematike i računarstva*. <http://www.matf.bg.ac.rs/cp/14/izborni-modul-profesor-matematike-i-racunarstva-2ml/>
- [101] Mayer, R. E. (2002). Rote versus meaningful learning. *Theory into practice*, 41(4), pp. 226-233.
- [102] Melnik, S., Garcia-Molina, H. & Rahm, E. (2002). Similarity Flooding: A Versatile Graph Matching Algorithm. In *Proceedings of the 18th International Conference on Data Engineering*. (ICDE), pp. 117-128.
- [103] Memon, QA. & Khoja, S. (2009). Academic Program Administration via Semantic Web – A Case Study. *Proceedings of International Conference on Electrical, Computer, and Systems Science and Engineering*, pp. 695-698.
- [104] Micheuz, P. (2008). Some findings on informatics education in Austrian academic secondary schools. *Informatics in Education*, 7(2), pp. 221-236.
- [105] Middle East Technical University (2013). Computer Education & Instructional Technology. <http://www.ceit.metu.edu.tr/undergraduate-curriculum>.
- [106] Mirabile, R. (1997). Everything you wanted to know about competency modeling. *Training and Development*, 51(8), pp. 73–77.
- [107] Mustaro, P. N. & Silveira, I. F. (2006). Learning objects: Adaptive retrieval through learning styles. *Interdisciplinary Journal of Knowledge and Learning Objects*, 2, pp. 35-46.
- [108] Naturwissenschaftlich-Technische Fakultät (2012). Akkreditierungsantrag der Fächer Informatik, Elektrotechnik,

Elektrotechnik und Technische Informatik,
 Maschinenbautechnik, Maschinenbautechnik und
 Fertigungstechnik. <http://www.uni-siegen.de/zlb/studium/bama/downloads/mhb/bk/mhb-mb.pdf>

- [109] Ngo, D., Bellahsene, Z. & Coletta, R. (2011). YAM++ - Results for OAEI 2011. In *Proceedings of the 6th International Workshop on Ontology Matching (OM-2011)*, Volume 814, pp. 228–235
- [110] Ngo, D. & Bellahsene, Z. (2012). Yam++: A multi-strategy based approach for ontology matching task. In: ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d'Acquin, M., Nikolov, A., Aussenac-Gilles, N., Hernandez, N. (eds.) *EKAUW 2012*. LNCS, vol. 7603, pp. 421–425. Springer, Heidelberg
- [111] Niemi, H. & Jakku-Sihvonen, R. (2009). Teacher Education Curriculum of Secondary School Teachers. http://www.ince.mec.es/revistaeducacion/re350/re350_08ing.pdf
- [112] Nikitchenko, M. (2011) Gnoseology-based Approach to Foundations of Informatics. In: ERMOLAYEV, V., MAYR, H. C., NIKITCHENKO, M., SPIVAKOVSKY, A., ZAVILEYSKY, M. AND ZHOLTKEVYCH, G. *ICT in Education, Research and Industrial Applications: Integration, Harmonization and Knowledge Transfer - Proceedings of the 7-th International Conference ICTERI 2011*, Kherson, May 2011. Aachen: Technical University of Aachen, pp. 27 – 40.
- [113] Noy, N. F., Ferguson, R. W. & Musen, M.A. (2000). The knowledge model of Protege-2000: Combining interoperability and flexibility. *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management (EKAW'00)*, pp. 17-32.
- [114] O'Neill, G. & Murphy, F. (2010). Guide to Taxonomies of Learning, *UCD Teaching and Learning/ Resources*. <http://www.ucd.ie/t4cms/ucdtla0034.pdf>
- [115] Paquette, G. (2007). An Ontology and a Software Framework for Competency Modeling and Management. *Educational Technology & Society*, 10(3), pp. 1-21.
- [116] Pathmeswaran, R. & Ahmed, V. (2009). SWmLOR: Technologies for Developing Semantic Web based Mobile Learning Object Repository. *The Built & Human Environment Review*, 2(1). <http://www.tbher.org/index.php/tbher/article/view/16/17>.
- [117] Pietig, J. (1997). Foundations and teacher education: Do we need a new metaphor? *J. Teach. Educ.*, 48(3), pp. 177–184.
- [118] Poirot, J., Luerhmann, A., Norris, C., Taylor, H. & Taylor, R. (1985). Proposed curriculum for programs leading to teacher certification in computer science. *Comm. ACM*, 28(3), pp. 275–279.
- [119] Poirot, J. L., Taylor, H. G. & Norris, C. A. (1988). Retraining teachers to teach high school computer science. *Comm. ACM*, 31(7), pp. 912--917.

- [120] Princeton University (2010). *About WordNet*. WordNet. Princeton University. <http://wordnet.princeton.edu/wordnet/>
- [121] Prirodno-matematički fakultet Kragujevac (2012). *Studijski program osnovnih akademskih studija informatike*. <http://www.pmf.kg.ac.rs/images/doc/akreditacija2013/prilozi/standard1/osnovnestudijeinformatike2012.pdf>
- [122] Prirodno matematički fakultet Niš (2007). *Profesor matematike i računarstva*. <http://www.pmf.ni.ac.rs/pmf/studije/matematika/matematika.php>
- [123] Prirodno-matematički fakultet Novi Sad (2012). *Akreditacija*. <http://www.pmf.uns.ac.rs/studije/akreditacija2012>
- [124] Rahm, E. (2011). Towards large-scale schema and ontology matching. *Schema matching and mapping*, pp. 3–27
- [125] Ragonis, N., Hazzan, O. & Gal-Ezer, J. (2010). A survey of Computer Science teacher preparation programs in Israel tells us: Computer Science deserves a designated high school teacher preparation!. *Proceedings of SIGCSE 2010 - The 41st ACM Technical Symposium on Computer Science Education*, pp. 401-405.
- [126] Ragonis, N., Hazzan, O. & Gal-Ezer, J. (2011). A Study on Attitudes and Emphases in Computer Science Teacher Preparation. *Proceedings of SIGCSE 2011*, pp. 559-564.
- [127] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 448–453.
- [128] Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11, pp. 95–130.
- [129] Robertson, S. & Jones, K.S. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3), pp. 129–146.
- [130] Romero, L., Gutierrez, M. & Caliusco, L. (2014). Towards Semantically Enriched E-learning Assessment: Ontology-Based Description of Learning Objects. *Advanced Learning Technologies (ICALT)*, 2014 IEEE 14th International Conference on, pp. 336-338.
- [131] Sanchez-Alonso, S. & Vovides, Y. (2007). Integration of metacognitive skills in the design of learning objects. *Computers in Human Behavior*, 23(6), pp. 2585-2595.
- [132] Schoon, K. J. & Sandoval, P. A. (1997) The seamless field experience model for secondary teacher preparation. *J. Sci. Teach. Educ.* 8(2), pp. 127–140.
- [133] Shippmann, J., Ash, R., Battista, M., Carr, L., Eyde, L., Hesketh, B., Kehoe, J., Pearlman, K., Prien, E. & Sanchez, J. (2000). The practice of competency modeling, *Personnel Psychology*, 53(3), pp. 703-740.
- [134] Shulman, L. S. (1986). Those who understand: Knowledge growth in teaching. *Educ. Teach.* 15(2), pp. 4–14.

- [135] Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform. *Harvard Educ. Rev.*, 57(1), pp. 1-22.
- [136] Shulman, L. S. (1990). Reconnecting foundations to the substance of teacher education. *Teach. Coll. Rec.* 91(3), pp. 300–310.
- [137] Shvaiko, P. & Euzenat, J. (2011). Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25 (1), pp.158 – 176.
- [138] Sirin, E., Parsia, B., Cuenca, B., Grau, , Kalyanpur, A. & Katz, Y. (2007). Pellet: a practical OWL-DL reasoner. *J. Web Semantics*, 5(2), pp. 51-53.
- [139] Stephenson, C., Gal-Ezer, J., Haberman, B. & Verno, A. (2005). The new educational imperative: Improving high school computer science education. Final report of the CSTA Curriculum Improvement Task Force. http://csta.acm.org/Communications/sub/DocsPresentationFiles/White_Paper07_06.pdf.
- [140] Stoilos, G., Stamou, G. , & Kollias, S. (2005). A String Metric for Ontology Alignment. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) *ISWC 2005. LNCS,3729*, pp. 623–637.
- [141] Tallin University (2013). Teacher of Computer Sciences, School ICT Manager. https://ois.tlu.ee/portal/page?_pageid=35,454968&_dad=portal&_schema=PORTAL&p_action=new_lang&p_fk_str_yksus_id=56&p_kava_version_id=4710&p_net=internet&p_lang=EN&p_ezhiim=0&p_mode=1
- [142] Teaching Agency (2012). Subject knowledge requirements for entry into computer science teacher training. http://academy.bcs.org/sites/academy.bcs.org/files/subject_knowledge_requirements_for_entry_into_cs_teacher_training.pdf
- [143] Technische Universität Graz (2013). Informatik und Informatikmanagement. https://online.tugraz.at/tug_online/semesterplaene.uebersicht?cor_g_nr=&csr_nr=141&pStpStpNr=529&csj_nr=1034&csprache_nr=1
- [144] Technische Universität Wien (2012). Informatikdidaktik. <http://www.informatik.tuwien.ac.at/studium/angebot/studienplaene/informatikdidaktik>
- [145] Tedre, M. (2007). Know your discipline: Teaching the philosophy of computer science. *The Journal of Information Technology Education*, 6(1), pp. 105-122. <http://jite.org/documents/Vol6/JITEv6p105-122Tedre266.pdf>
- [146] Tehnički fakultet “Mihajlo Pupin” (2009). *Informacione tehnologije*. <http://www.tfzr.uns.ac.rs/Content/akreditacija/Informacione%20tehnologije%20osnovne%20studije.pdf>
- [147] Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C. & Verno, A. (2003). A Model Curriculum for K-12 Computer Science. The ACM K-12 Education Task Force

<http://www.csta.acm.org/Curriculum/sub/ACMK12CSModel.html>

- [148] Tucker, A., Deek, F., Jones, J., McCowan, D., Stephenson, C. & Verno, A. (2006). A Model Curriculum for K-12 Computer Science: Final Report of the ACM K-12 Task Force Curriculum Committee. 2nd Edition, Association for Computing Machinery (ACM), New York.
- [149] UNESCO (2000). *INFORMATION AND COMMUNICATION TECHNOLOGY IN SECONDARY EDUCATION, A Curriculum for Schools*.
<http://www.edu.ge.ch/cptic/prospective/projets/unesco/en/curriculum2000.pdf>
- [150] Universität Potsdam (2012). Bachelor- und Masterstudium im Fach Informatik für das Lehramt. <http://www.uni-potsdam.de/ambek/ambek2012/3/Seite2.pdf>
- [151] Universität Rostock (2013). Studiengangsspezifische Prüfungs- und Studienordnung für den Studiengang Lehramt an Gymnasien. http://www.uni-rostock.de/fileadmin/UniHome/Gremien/Bekanntmachungen/N_R_8_2013.pdf
- [152] Universität Salzburg (2013). Curriculum für das Bachelorstudium. https://online.unisalzburg.at/plus_online/wbMitteilungsblaetter.display?pNr=241797
- [153] University of Stirling (2013). Study Secondary Education and graduate with a university degree. <http://www.stir.ac.uk/undergraduate-study/course-information/courses-a-to-z/school-of-education/education-secondary/>
- [154] Vaccari, L., Shvaiko, P. & Marchese, M. (2009). A geo-service semantic integration in spatial data infrastructures. *Journal of Spatial Data Infrastructures Research*, vol. 4, pp. 24–51.
- [155] Van Diepen, N., Perrenet, J. & Zwaneveld, B. (2011). Which Way with Informatics in high schools in the Netherlands? The Dutch dilemma. *Informatics in Education*, 10(1), pp. 123–148.
- [156] Verno, A., Carter, D., Cutler, R., Hutton, M. & Pitt, L. (2006). A Model Curriculum for K-12 Computer Science: Level II Objectives and Outlines. <http://csta.acm.org/Curriculum/sub/CurrFiles/L2-Objectives-and-Outlines.pdf>
- [157] Vesin B., Ivanović M., Klačnja-Milićević A. & Budimac Z. (2012). Protus 2.0: ontology-based semantic recommendation in programming tutoring system. *Experts systems with application*, 39(15), pp. 12229–12246.
- [158] Xiao, H. & Cruz, I. (2006). Ontology-based Query Rewriting in Peer-to-Peer Networks. In *Proc. Of the 2nd International Conference on Knowledge Engineering and Decision Support*. <http://www.cs.uic.edu/~advis/publications/dataint/ickeds06.pdf>

- [159]Xin-juan Z., Xian-Feng, L. & Wei, G. (2007). Ontology Based Sharing and Services in E-Learning Repository. In *International Conference on Network and Parallel Computing*, pp. 18-21.
- [160]Yaghmaie, M. & Bahreininejad, A. (2011). A context-aware adaptive learning system using agents. *Expert Systems with Applications*, 38(4), pp. 3280-3286
- [161]Wang, S. (2008). Ontology of learning objects repository for pedagogical knowledge sharing. *Interdisciplinary Journal of E-Learning and Learning Objects*, 4, pp. 1-12
- [162]Wilson, C., Sudol, L., Stephenson, C. & Stehlik, M. (2010). Running on empty: the failure to teach K/12 Computer Science in the digital age. *Tech. rep. ACM*, New York, <http://www.acm.org/runningonempty/>.
- [163]Winkler, W. (1999). *The state of record linkage and current research problems*, tech. report 99/04, Statistics of Income Division, Internal Revenue Service Publication.
- [164]Wu, W., Yu, C., Doan, A. & Meng, W. (2004). An Interactive Clustering-based Approach to Integrating Source Query interfaces on the Deep Web. *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, pp. 95-106.
- [165]Wu, Z. & Palmer, M. (1994). Verb semantics and lexical selection. In: 32nd. Annual Meeting of the Association for Computational Linguistics, pp. 133 – 138. New Mexico State University, Las Cruces, New Mexico
- [166]W3C Recommendation (2004a). OWL, Web Ontology Language, <http://www.w3.org/TR/owl-features/>
- [167]W3C Recommendation (2004b). SWRL, Semantic Web Rule Language, <http://www.w3.org/Submission/SWRL/>
- [168]Zablith, F. (2009) Evolva: A comprehensive approach to ontology evolution, *In Proc 6th European Semantic Web Conference (ESWC) PhD Symposium*, pp. 944-948.
- [169]Zeidler, D. L. (2002). Dancing with maggots and saints: Visions for subject matter knowledge, pedagogical knowledge and pedagogical content knowledge in science teacher education reform. *Journal of Science Teacher Education*, 13(1), pp. 27-42.

Prilozi

Prilog 1. Skraćene oznake fakulteta/univerziteta

Fakultet/Univerzitet	Oznaka
Fakultät für Informatik, Technische Universität Dortmund, Nemačka	GER1
Fakultät für Wirtschaftswissenschaften, Universität Duisburg-Essen, Nemačka	GER2
Naturwissenschaftlich-Technische Fakultät, Universität Siegen, Nemačka	GER3
Fakultät für Elektrotechnik, Informatik und Mathematik, Universität Paderborn, Nemačka	GER4
Goethe-Universität, Nemačka	GER5
Universität Rostock, Nemačka	GER6
Universität Potsdam, Nemačka	GER7
Universität Salzburg, Austrija	AUS1
Technische Universität Graz, Austrija	AUS2
Technische Universität Wien, Austrija	AUS3
Tallin University, Estonija	EST1
Open University, Izrael	ISR1
College of Science and Technology, Texas, SAD	USA1
Stirling University, Škotska	SCO1
Middle East Technical University, Turska	TUR1
Tehnički fakultet "Mihajlo Pupin", Zrenjanin	SRB1
Tehnički fakultet Čačak, Srbija	SRB2
Prirodno-matematički fakultet Novi Sad, Srbija	SRB3
Matematički fakultet, Beograd, Srbija	SRB4
Prirodno-matematički fakultet Kragujevac, Srbija	SRB5
Državni Univerzitet u Novom Pazaru	SRB6
Prirodno-matematički fakultet Niš, Srbija	SRB7

Prilog 2

UPITNIK

1. Formalno obrazovanje

Nivo studija	Naziv fakulteta	Naziv studijskog programa	Godina upisa	Godina završetka	Prosečna ocena
Osnovne bečelor (3 godine)					
Osnovne akademske (4 godine)					
Master studije (1 ili 2 godine)					
Magistarske studije (2 godine)					
Doktorat (po starom)					
Doktorske studije (3 godine)					

2. Nastavničko iskustvo u godinama

- a. U osnovnoj školi _____
- b. U srednjoj školi _____
- c. U visokoškolskoj instituciji _____

3. Rad u nastavi

Navedite podatke o Vašim trenutnim nastavnim aktivnostima:

Obrazovni profili na kojima držite nastavu	Predmeti koje predajete (po profilima)

4. Struktura formalnog obrazovanja

- a. Navedite nazive predmeta iz oblasti **Informatičkih znanja**¹ koje ste položili u okviru studija

Redni broj	Naziv predmeta
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	

- b. Navedite nazive predmeta iz oblasti **Opštih pedagoških znanja**² koje ste položili u okviru studija

Redni broj	Naziv predmeta
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	

-
- ¹ **Informatička (Domenska) znanja** se odnose na užu domenu informatike/računarskih nauka. Domenskim znanjima pripadaju predmeti poput: programskih jezika, baza podataka, računarskih mreža, kompjuterske etike, hardvera, istorije računarstva i sl.
- ² **Opšta pedagoška znanja** čine *pedagoško – psihološki i opšti didaktički predmeti* i odnose se na pedagoške teorije, strategije učenja (nastavne metode), instrukcije u nastavi (koncepte za personalizovano učenje, kolaborativnu nastavu, tehnike valorizacije učeničkog postignuća i evaluacije nastavnog procesa (merenja i ocenjivanja) i sl.)

- c. Navedite nazive predmeta iz oblasti **Metodike nastave informatike**³ koje ste položili tokom studija

Redni broj	Naziv predmeta
1	
2	
3	
4	
5	

- d. Navedite nazive predmeta iz oblasti **Opštih znanja i veština**⁴ koje ste položili tokom studija

Redni broj	Naziv predmeta
1	
2	
3	
4	
5	

- e. Označite sa X znanja i veštine iz **oblasti nastavne prakse** koje ste imali prilike da steknete tokom studija

Redni broj	Naziv znanja/veštine	
1	Organizacija rada škole	<input type="checkbox"/>
2	Upravljanje školom	<input type="checkbox"/>
3	Delatnost odeljenskog starešine	<input type="checkbox"/>
4	Metodička praksa u realizaciji nastave informatičkih sadržaja (hospitovanje)	<input type="checkbox"/>

-
- 3 Znanja metodike nastave informatike - Domenska metodička znanja** obuhvataju: odnos saznavanja u nauci i nastavi, specifičnosti saznavanja u nastavi, logičko-gnoseološka struktura znanja i razvoj pojmova u informatici; jezik informatike; nastavne strategije i instrukcije u nastavi informatike; oblici rada u nastavi informatike; tok nastavnog procesa u nastavi informatike; planiranje nastave i kurikulum informatike, nastavnikovo pripremanje i samoevaluacija u nastavi informatike; valorizacija učeničkog postignuća u nastavi informatike.
- 4 Opšta znanja i veštine** obuhvataju predmete iz oblasti: matematike, fizike, hemije, jezika, menadžmenta.

5. Sticanje znanja nakon formalnog obrazovanja

i. Navedite način na koji ste stekli dodatna znanja iz oblasti

Informatičkih znanja

1. nisam sticao/la dodatna znanja
2. učešće u radionicama/seminarima
3. učešće na naučnim i stručnim konferencijama
4. proučavanje literature
5. pohađanje posebnih kurseva (unesite podatke o najvažnijim pohađanim kursevima u sledeću tabelu):

Redni broj	Naziv kursa i naziv organizatora kursa	Kratak sadržaj kursa	Trajanje kursa (u danima)
1			
2			
3			
4			
5			

6. Neki drugi način –
navesti koji _____

ii. Navedite način na koji ste stekli dodatna znanja iz oblasti

Opštih pedagoških znanja

1. nisam sticao/la dodatna znanja
2. učešće u radionicama/seminarima
3. učešće na naučnim i stručnim konferencijama
4. proučavanje literature
5. pohađanje posebnih kurseva (unesite podatke o najvažnijim pohađanim kursevima u sledeću tabelu):

Redni broj	Naziv kursa i naziv organizatora kursa	Kratak sadržaj kursa	Trajanje kursa (u danima)
1			
2			
3			
4			
5			

6. Neki drugi način –
navesti koji _____

iii. Navedite način na koji ste stekli dodatna znanja iz oblasti
Metodike nastave informatike

1. nisam sticao/la dodatna znanja
2. učešće u radionicama/seminarima
3. učešće na naučnim i stručnim konferencijama
4. proučavanje literature
5. pohađanje posebnih kurseva (unesite podatke o najvažnijim pohađanim kursevima u sledeću tabelu):

Redni broj	Naziv kursa i naziv organizatora kursa	Kratak sadržaj kursa	Trajanje kursa (u danima)
1			
2			
3			
4			
5			

6. Neki drugi način –
navesti koji _____

iv. Navedite način na koji ste stekli dodatna znanja iz oblasti

Znanja nastavne prakse

1. nisam sticao/la dodatna znanja
2. učešće u radionicama/seminarima
3. učešće na naučnim i stručnim konferencijama
4. proučavanje literature
5. pohađanje posebnih kurseva (unesite podatke o najvažnijim pohađanim kursevima u sledeću tabelu):

Redni broj	Naziv kursa i naziv organizatora kursa	Kratak sadržaj kursa	Trajanje kursa (u danima)
1			
2			
3			
4			
5			

6. Neki drugi način –
navesti koji _____

6. Standardizacija nastavnog plana i programa (kurikuluma)

i. Da li su Vam poznate faze i učesnici procesa standardizacije kurikuluma?

- a. Uopšte mi nije poznat proces
- b. Poznajem minimalno proces
- c. Znam osnovnu strukturu procesa i mogu nabrojati većinu učesnika
- d. Poznajem ceo proces

ii. Da li Vam je poznat ACM K12 CS standard?

- a. Nisam upoznat/a
- b. Znam ukupan planirani cilj
- c. Poznajem strukturu i ciljeve po nivoima
- d. Detaljno poznajem standard

iii. Ocenite potrebu usaglašavanja informatičkog obrazovanja u R. Srbiji sa ACM K12 CS standardom (ukoliko ga poznajete):

- a. Nema potrebe
- b. Potrebno je minimalno usaglašavanje
- c. Potrebno je srednje usaglašavanje
- d. Standard treba usvojiti

iv. Da li Vam je poznat ECDL plan i program?

- a. Nisam upoznat/a
- b. Znam ukupan planirani cilj
- c. Poznajem strukturu i nazive svih modula
- d. Poznajem sadržaje svih modula

v. Ocenite potrebu usaglašavanja informatičkog obrazovanja na srednjoškolskom nivou sa ECDL sertifikacijom (ukoliko je poznajete):

- a. Nema potrebe
- b. Potrebno je minimalno usaglašavanje
- c. Potrebno je srednje usaglašavanje
- d. Principe ECDL-a treba usvojiti u našim srednjim školama

7. Samoevaluacija

i. Poznavanje nastavnih planova i programa (kurikuluma)

- a. Ocenite ocenom 1 – 5⁵ Vaše poznavanje informatičkih predmeta u nastavnom planu osnovne škole
- b. Ocenite ocenom 1 – 5⁶ Vaše poznavanje informatičkih predmeta koji postoje u nastavnim planovima srednjoškolskog nivoa obrazovanja (profili za koje Vi ne držite nastavu)
- c. Ocenite ocenom 1 – 5⁷ Vaše poznavanje informatičkih predmeta u informatički usmerenim studijskim programima visokoškolskog nivoa obrazovanja
- d. Ocenite ocenom 1 – 5⁸ Vaše poznavanje informatičkih predmeta u studijskim programima visokoškolskog nivoa obrazovanja koji nisu informatički usmereni
- e. **Domen informatike:** Ocenite ocenom 1 – 5⁹ Vaše poznavanje sledećih tematskih oblasti informatičkog (domenskog) aspekta:

Tematske oblasti	Ocena
Osnove informatike	
<ul style="list-style-type: none">• Informacija i informatika.• Predstavljanje raznih tipova informacija (tekstualne, grafičke i zvučne).• Kodiranje karaktera, kodne sheme.• Jedinice za merenje količine informacija.	

- 5 Značenje ocene je: 1 – ne znam koji su to predmeti; 2 – znam samo koji su to predmeti; 3 – znam planirane ciljeve predmeta; 4 – znam planirane ishode predmeta; 5 – znam nastavne sadržaje predmeta.
- 6 Značenje ocene je: 1 – ne znam koji su to predmeti; 2 – znam samo koji su to predmeti; 3 – znam planirane ciljeve predmeta; 4 – znam planirane ishode predmeta; 5 – znam nastavne sadržaje predmeta.
- 7 Značenje ocene je: 1 – ne znam koji su to predmeti; 2 – znam samo koji su to predmeti; 3 – znam planirane ciljeve predmeta; 4 – znam planirane ishode predmeta; 5 – znam nastavne sadržaje predmeta.
- 8 Značenje ocene je: 1 – ne znam koji su to predmeti; 2 – znam samo koji su to predmeti; 3 – znam planirane ciljeve predmeta; 4 – znam planirane ishode predmeta; 5 – znam nastavne sadržaje predmeta.
- 9 Značenje ocene je: 1 – ne znam o oblasti; 2 – znam da postoji ali ništa ne znam o oblasti; 3 – znam koji su osnovni koncepti oblasti ali ne razumem njihovu suštinu; 4 – poznajem i razumem sve osnovne koncepte; 5 – posedujem praktično iskustvo (za praktične teme to znači da ste programirali/radili sa alatom; za istoriju računarstva i slično to znači da ste to nekom predavali).

<ul style="list-style-type: none"> • Istorija računarstva. • Značaj i primena računara. • Interdisciplinarna priroda računarstva i veza sa drugim disciplinama. • Informatika u perspektivi (profesije/karijere u računarstvu). 	
Računarski hardver	
<ul style="list-style-type: none"> • Organizacija i princip rada računara. • Vrste memorije računara. • Procesor. • Matična ploča. • Magistrala. • Ulazno-izlazni uređaji. • Uticaj komponenti na performanse računara. 	
Osnove rada u operativnom sistemu sa grafičkim interfejsom	
<ul style="list-style-type: none"> • Osnovni elementi grafičkog interfejsa operativnog sistema (radna površina, prozor, ikona, dugme, panel, meni, katalog). Pokretanje programa. • Osnovna podešavanja operativnog sistema: radne površine (pozadine, čuvara ekrana, rezolucije ekrana), promena korisničkih naloga i sl. • Datoteka (atributi datoteke, putanja datoteke, grupno ime datoteka) i osnovne operacije nad datotekom. Direktorijumi. • Arhiviranje datoteka i sredstva za arhiviranje datoteka. • Sredstva i metode zaštite računara i informacija. 	
Programska podrška računara	
<ul style="list-style-type: none"> • Sistemski i aplikativni softver. • Distribucija i preuzimanje programskih proizvoda (komercijalna, deljena (engl. shareware), besplatna (engl. freeware),), probna (engl. trial), open source). • Instaliranje korisničkih programa. Uklanjanje programa. Instaliranje upravljačkih programa perifernih uređaja. 	
Softverske aplikacije za rad sa tekstem	
Softverske aplikacije za rad sa prezentacijama	
Softverske aplikacije za rad sa tabelama	
Matematičke osnove računara	
<ul style="list-style-type: none"> • Binarni brojni sistem. • Logička kola. • Automati. 	
Hardverski dizajn	
<ul style="list-style-type: none"> • Dizajn hardvera. • Instaliranje (sastavljanje) i podešavanje računarskog sistema. • Kriterijumi za „naručivanje“ i unapređivanje računarskog hardvera. 	
Operativni sistemi	
<ul style="list-style-type: none"> • Funkcije OS-a. • Tipovi operativnih sistema. • Komponente OS-a. • Fajl sistem. • Instaliranje OS-a i podešavanje BIOS-a. • Sistemski podešavanja komercijalnih i besplatnih OS-a (Windows, Linux, itd). • Naredbe iz komandne linije (Command prompt) • Napredna podešavanje zaštite OS-a. • Programiranje operativnog sistema (osnove asemblerskog programiranja) • Virtuelne mašine. 	
Osnove računarskih mreža	

<ul style="list-style-type: none"> • Pojednostavljeni model komunikacije. • Tehnologije prenosa (žično, bežično). • Definicija računarske mreže. • Komponente računarske mreže. Aktivne komponente. Pasivne komponente. • Topologija računarske mreže. • Adresiranje. • Mrežni hardver (Mrežna karta , Switch , Router). • Mrežni protokoli. ISO/OSI referentni model. • Softverska arhitektura mreže (peer-to-peer i klijent server). • Performanse mreže (propusni opseg, kašnjenje). 	
<p>Osnove Interneta i korišćenje Interneta</p> <ul style="list-style-type: none"> • Osnove Interneta <ul style="list-style-type: none"> • Organizacija Interneta. • Pristup Internetu. • Arhitektura Interneta i osnovni Internet protokoli. • Standardni Internet servisi (www, ftp, e-mail). • Korišćenje Interneta <ul style="list-style-type: none"> • Pretraživači, pretraživanje i korišćenje informacija sa Interneta. • Alati za komunikaciju (razmena poruka, fajlova, deljenje ekrana, audio i video pozivi). • Elektronska trgovina, elektronsko poslovanje i bankarstvo. • Elektronski podržano učenje. • Osnove Web 2.0 i društvene mreže. • Rizici i bezbednost na Internetu <ul style="list-style-type: none"> • Pouzdanosti i autentičnost informacija na Internetu. • Privatnost. • Internet kriminal. 	
<p>Standardi za elektronski podržano učenje</p> <ul style="list-style-type: none"> • Standardi za opis metapodataka (IEEE LOM). • Standardi za definisanje obrazovnih paketa (SCORM). • Specifikacija podataka o učenicima (IMS Learner Information Package). • Specifikacija ispitnih pitanja (IMS Question and Test Interoperability). • Specifikacija načina predavanja (IMS Learning Design). 	
<p>Kompjuterska etika, pitanja legalnosti, bezbednosti i zaštite podataka</p> <ul style="list-style-type: none"> • Aspekti društveno i etički ispravnog korišćenja računara i konsekvence neadekvatne primene. • Bezbednost, uslovi korišćenja i zaštite intelektualne svojine u računarstvu. • Društvene i ekonomske implikacije hakerisanja i softverske piraterije. • Razlike između prava pristupa informacija i prava distribucije. • Uticaj računara na zdravlje. • Uticaj pravne regulative na privatnost i bezbednost. • Identifikovanje i razlikovanje prikladnog i neodgovarajućeg ponašanja na društvenim mrežama. • Etička pitanja vezana za računarske mreže (bezbednost, privatnost, open source razvoj, deljenje informacija, javni domeni i sl.). 	
<p>Računarska grafika</p> <ul style="list-style-type: none"> • Načini predavljanja statičkih slika u računaru – vektorska i rasterska grafika. 	

<ul style="list-style-type: none"> • Karakteristike računarske grafike – rezolucija i broj boja. • Klasifikacija programa za rad sa računarskom grafikom. • Formati datoteka. • Ulazne i izlazne grafičke jedinice. • Primena programa za kreiranje i obradu rasterske grafike. • Primena programa za kreiranje i obradu vektorske grafike. 	
Multimedija	
<ul style="list-style-type: none"> • Predstavljanja zvuka u računaru. • Osnovni formati zapisa zvuka (WAV, MP3, MIDI,...). • Programi za reprodukciju zvučnih zapisa. • Programi za snimanje zvuka. • Načini predstavljanja video-zapisa u računaru. • Osnovni formati video-zapisa. • Programi za reprodukciju video zapisa. • Uvoz video zapisa sa digitalne kamere. • Programi za montažu video zapisa (kombinacija slike, videa i zvuka). • Postavljanje video zapisa na veb. 	
Tipovi i strukture podataka	
<ul style="list-style-type: none"> • Celobrojni tip. Opseg celobrojnog tipa. • Realni tip. Opseg realnog tipa. • Logički tip. • Znakovni tip. • Nizovni tip. • String tip i osnovne funkcije za rad sa stringovima. • Liste. • Stabla. • Stek. • Klasa (atributi i metode klase – osnovni pojmovi). 	
Algoritmi	
<ul style="list-style-type: none"> • Definicija algoritma. • Tipovi (linijski, razgranati, ciklični i strukture sa podprogramima). • Jezici za reprezentaciju algoritama (blok-dijagram, pseudo kod). 	
Rešavanje problema pomoću računara	
<ul style="list-style-type: none"> • Transformacija problema na oblik pogodan za rešavanje na računaru. • Klasifikovanje problema kao: prilagodljiv za računarsko procesuiranje ili računarski nerešiv. • Ocenjivanje algoritama po njihovoj efikasnosti, tačnosti i jasnoći. • Tipovi programskih jezika i programske paradigme. 	
Imperativno programiranje – naredbe dodele i izrazi	
<ul style="list-style-type: none"> • Aritmetički izrazi. Prioritet operacija. Logički izrazi. • Naredba dodele. • Konverzija tipova podataka. • Unošenje i prikazivanje podataka. • Algoritam razmene vrednosti dve promenljive. 	
Proceduralni programski jezici	
<ul style="list-style-type: none"> • Sintaksa i semantika. • Top-down pristup rešavanja problema u programiranju. • Procedure i parametri. • Sekvence, uslovni izrazi, rekurzije i iteracije. 	
Objektno-orijentisana paradigma	
<ul style="list-style-type: none"> • Klasa. • Metode. Formalni parametri metode. Telo metode. Sintaksa 	

<p>poziva metode. Stvarni parametri metode. Lokalne promenljive metode.</p> <ul style="list-style-type: none"> • Objekat. • Razmena poruka između objekata. • Konstruktori. • Nasleđivanje. • Polimorfizam. • Kompozicija i generalizacija. • Rukovanje izuzecima. • Biblioteke klasa. 	
UML (objedinjeni jezik za objektno modelovanje)	
Programiranje grafičkog korisničkog interfejsa - programi zasnovani na prozorima	
<ul style="list-style-type: none"> • Osnovne karakteristike programa zasnovanih na prozorima. • Elementi grafičkog korisničkog interfejsa (Graphical User Interface). • Komponente izbora: radio-dugme (RadioButton), okvir za potvrdu (CheckBox), okvir s listom (ListBox), kombinovani okvir (ComboBox). • Kontejnerske komponente: okvir za grupu (GroupBox), ploča (Panel). • Java applet-i. • Programi rukovođeni događajima (događaji, izvori događaja i obrada događaja) • GUI biblioteke klasa (AWT i Swing) 	
Integrirana razvojna okruženja	
<ul style="list-style-type: none"> • Definicija integrisanog razvojnog okruženja. • Komponente integrisanog razvojnog okruženja. • Rad i upravljanje razvojnim okruženjem. <ul style="list-style-type: none"> • Prazan projekat. Čuvanje i otvaranje projekta. • Dodavanje komponenti formi. • Svojstva komponenti i njihovo podešavanje. • Događaji komponenti i obrada događaja. • Pokretanje i dibagovanje programskog koda. 	
Baze podataka	
<ul style="list-style-type: none"> • Definicija baze podataka. • Tipovi baza podataka (relacione, objektno, hibridne). • Relacione baze podataka. <ul style="list-style-type: none"> • Polje, Slog, Tabela. • Ključ (evi). • Veze između tabela. • Funkcionalna zavisnost. • Zavisnost sadržavanja. • Referencijalni integritet. • Softver za upravljanje bazama podataka <ul style="list-style-type: none"> • Kreiranje baze podataka. • Ažuriranje baze podataka. Forme (obraci). • Pretraživanje, sortiranje i izveštavanje 	
Upiti nad bazama podataka	
<ul style="list-style-type: none"> • Standardizacija upitnog jezika. • Osnove SQL komandi (definicione SQL komande, kontrolne SQL komande, manipulacione SQL komande). • Kreiranje upita (sa i bez čarobnjaka). • Pregled rezultata upita. • Kreiranje multitabelarnih upita. • Izveštaji. • Vizuelizacija podataka iz baze. 	

Uspostavljanje lokalne računarske mreže (LAN) i povezivanje LAN na Internet	
<ul style="list-style-type: none"> • Lokalne mreže, formiranje i struktura. • Povezivanje čvorova mreže. • IP šema adresiranja. • Ruteri i rutiranje. • Organizacija domena i domenskih imena, DNS (Domain Name System). • Deljenje resursa lokalne mreže. • Navigacija kroz lokalnu mrežu. • Administriranje kućne lokalne mreže. • Povezivanje lokalne mreže sa globalnom mrežom (Internetom). • Arhitektura računarske mreže škole. 	
Napredni Internet servisi i Veb tehnologije	
<ul style="list-style-type: none"> • Računarstvo u oblaku (cloud computing). • Deljenje dokumenata na web-u. • Instaliranje i administriranje blog-a, wiki-ja. • Elektronski portfolio. • XML tehnologije. • Tehnologija Veb servisa. • Mobilne tehnologije i platforme. • Semantički Veb. 	
Programiranje statičkih veb strana (Uvod u HTML i CSS)	
<ul style="list-style-type: none"> • Osnove HTML-a. • Osnovni tagovi HTML-a. • HTML kontrole. • Stilovi u HTML-u (kaskadni stilovi - CSS). • Metode izrade HTML dokumenata i kaskadnih stilova. 	
Programiranje dinamičkih veb strana	
<ul style="list-style-type: none"> • Skripte sa klijentske strane (Java script). • Skripte sa serverske strane (JSP). • Programiranje Internet aplikacija. 	
Upravljanje gotovim veb dizajn rešenjima (CMS - Content Management System, LMS – Learning Management System)	
<ul style="list-style-type: none"> • Šta je CMS. • Osobine CMS-a. • Najčešće korišćeni CMS portali (Joomla, WordPress). • Instaliranje i podešavanje CMS portala. • Instaliranje i podešavanje LMS-a (Moodle). • Administracija CMS-a i LMS-a. 	
Osnove veštačke inteligencije (i kompjuterskih simulacija)	
<ul style="list-style-type: none"> • Osnovni pojmovi. • Osnove mašinskog učenja i veštačkih neuronskih mreža. • Primena koncepata veštačke inteligencije u obrazovanju. 	
Računarske obrazovne igre	
<ul style="list-style-type: none"> • Učenje kroz računarske igre. • Puzzle. • Obrazovne igre za jednog studenta, manje i veće grupe studenata. • Uloga nastavnika. • Potencijalna opasnost računarskih igara. 	

- f. **Opšte pedagoško znanje.** Ocenite ocenom 1 – 5¹⁰
Vaše poznavanje sledećih opšte pedagoških oblasti :

Oblast	Ocena
Obrazovna psihologija	
Sociologija obrazovanja	
Interaktivna pedagogija	
Didaktika	
Personalizovano učenje	

- g. **Domensko metodičko znanje:** Ocenite ocenom 1 – 5¹¹
Vaše poznavanje sledećih aspekata metodike nastave informatike:

Aspekt	Ocena
Odnos saznavanja u nauci u nastavi informatike	
Specifičnosti saznavanja u nastavi informatike	
Logičko-gnoseološka struktura znanja i razvoj pojmova u informatici	
Jezik informatike	
Nastavne strategije i instrukcije u nastavi informatike	
Oblici rada u nastavi informatike	
Tok nastavnog procesa u nastavi informatike	
Planiranje nastave i kurikulum informatike	
Nastavnikovo pripremanje i samoevaluacija u nastavi informatike	
Valorizacija učeničkog postignuća u nastavi informatike	

- h. **Opšta znanja i veštine:** Ocenite ocenom 1 – 5¹² nivo
Vaših opštih znanja/veština u odnosu na zahteve predmeta iz kojih držite nastavu:

Opšte znanje/veština	Ocena
Strani jezici	
Matematika	
Sistemi za upravljanje obrazovnim sadržajem/okruženja za učenje (LMS, CMS)	

-
- ¹⁰ Značenje ocene je: 1 - nisam čuo/la za oblast; 2 – znam da oblast postoji ali ne poznajem ni osnovne koncepte; 3 – znam koji su osnovni koncepti oblasti ali ne razumem njihovu suštinu; 4 – poznajem i razumem osnovne koncepte; 5 – primenjujem znanja iz oblasti u nastavi.
- ¹¹ Značenje ocene je: 1 – nisam čuo/la za oblast; 2 - znam da postoji ali ne poznajem ni osnovne koncepte; 3 - znam koji su osnovni koncepti oblasti ali ne razumem njihovu suštinu; 4 - poznajem i razumem sve osnovne koncepte; 5 – primenjujem znanja u nastavi.
- ¹² Značenje ocene je: 1- nikakvo; 2 – loše; 3 – dobro; 4 – vrlo dobro; 5 – odlično.

Kolaborativan softver/društveni servisi	
Obrazovna tehnologija	

- i. **Angažovanje u svakodnevnoj nastavnoj praksi:**
 Ocenite ocenom 1 – 5¹³ Vaše angažovanje u sledećim oblastima svakodnevne nastavne prakse:

Oblast	Ocena
Planiranje nastave(ciljevi i ishodi)	
Kreiranje novog nastavnog materijala	
Selekcija i korišćenje postojećeg nastavnog materijala	
Izbor nastavnih metoda	
Izbor metoda i oblika rada prilagođenih različitim tipovima učenika	
Razvoj, priprema i primena adekvatnih metoda ocenjivanja	
Organizovanje učenika u grupe i kolaborativan rad	
Saradnja sa drugim nastavnicima i institucijama	
Rad u računarskim laboratorijama	
Istraživanje	

8. Ocene o modalitetima obrazovanja nastavnika informatike

Označite koji koncept obrazovanja nastavnika informatike smatrate najpodesnijim?

- a. Master akademske studije usmerene na pedagoške aspekte i aspekte metodike nastave informatike koje bi pohađali svršeni studenti osnovnih akademskih studija informatike
- b. Master akademske studije usmerene na informatičke (domenske) aspekte koje bi pohađali svršeni studenti osnovnih akademskih pedagoških studija
- c. Integrisane osnovne i master akademske studije na kojima bi već od nivoa osnovnih akademskih studija studenti izučavali sadržaje predviđenim kurikulumom nastave informatike uključujući sve aspekte
- d. Neki drugi pristup, opisati princip

¹³ Značenje ocene je: 1 – nikakvo; 2 – minimalno; 3 – zadovoljavajuće; 4 – vrlo dobro; 5 – odlično.

9. Stavovi nastavnika o kurikulumima (nastavnim planovima i programima)

i. Sadržaji nastavnih planova i programa (kurikuluma)

i.1. Ocenite ocenom 1 – 5 koliki je, po Vama, **značaj** sledećih **aspekata**¹⁴ za uspešno realizovanje nastave za nivo obrazovanja na kome Vi držite nastavu:

a. Informatička (Domenska) znanja

Tematske oblasti	Značaj u
Osnove informatike	
Računarski hardver	
Osnove rada u operativnom sistemu sa grafičkim interfejsom	
Programska podrška računara	
Softverske aplikacije za rad sa tekstem	
Softverske aplikacije za rad sa prezentacijama	
Softverske aplikacije za rad sa tabelama	
Matematičke osnove računara	
Hardverski dizajn	
Operativni sistemi	
Osnove računarskih mreža	
Osnove Interneta i korišćenje Interneta	
Standardi za elektronski podržano učenje	
Kompjuterska etika, pitanja legalnosti, bezbednosti i zaštite podataka	
Računarska grafika	
Multimedija	
Tipovi i strukture podataka	
Algoritmi	
Rešavanje problema pomoću računara	
Imperativno programiranje - naredbe dodele i izrazi	
Proceduralni programski jezici	
Objektno-orijentisana paradigma	
UML (objedinjeni jezik za objektno modelovanje)	
Programiranje grafičkog korisničkog interfejsa - programi zasnovani na prozorima	
Integrisana razvojna okruženja	
Baze podataka	
Upiti nad bazama podataka	

¹⁴ Značenje ocene je: 1 – nema nikakav značaj; 2 – vrlo mali značaj; 3 – srednji značaj; 4 – veliki značaj; 5 – presudan značaj.

Tematske oblasti	Značaj u nastavi
Uspostavljanje lokalne računarske mreže (LAN) i povezivanje LAN na Internet	
Napredni Internet servisi i Veb tehnologije	
Programiranje statičkih veb strana (Uvod u HTML i CSS)	
Programiranje dinamičkih veb strana	
Upravljanje gotovim veb dizajn rešenjima (CMS - Content Management System, LMS – Learning Management System)	
Osnove veštačke inteligencije (i kompjuterskih simulacija)	
Računarske obrazovne igre	
Navedite neke druge teme i oblasti, ukoliko postoje, koje smatrate značajnim, a nisu ovde navedene:	

b. Opšti pedagoški aspekt

Oblast	Značaj u nastavi
Obrazovna psihologija	
Sociologija obrazovanja	
Interaktivna pedagogija	
Didaktika	
Personalizovano učenje	
Navedite neke druge teme i oblasti, ukoliko postoje, koje smatrate značajnim, a nisu ovde navedene:	

c. Opšta znanja i veštine

Oblast	Značaj u nastavi
Strani jezici	
Matematika	
Sistemi za upravljanje obrazovnim sadržajem/okruženja za učenje (LMS, CMS)	
Kolaborativan softver/društveni servisi	
Obrazovna tehnologija	
Navedite neke druge teme i oblasti, ukoliko postoje, koje smatrate značajnim, a nisu ovde navedene:	

d. Domenski metodički aspekt

Oblast	Značaj u nastavi
Odnos saznavanja u nauci u nastavi informatike	
Specifičnosti saznavanja u nastavi informatike	
Logičko-gnoseološka struktura znanja i razvoj pojmova u informatici	
Jezik informatike	
Nastavne strategije i instrukcije u nastavi informatike	
Oblici rada u nastavi informatike	
Tok nastavnog procesa u nastavi informatike	
Planiranje nastave i kurikulum informatike	
Nastavnikovo pripremanje i samoevaluacija u nastavi informatike	
Valorizacija učeničkog postignuća u nastavi informatike	
Navesti neke druge teme i oblasti, ukoliko postoje, koje smatrate značajnim, a nisu ovde navedene:	

e. Nastavna praksa

Oblast	Značaj u nastavi
Opšta pedagoška praksa	
Hospitovanje	

i.2. Ocenite ocenom 1 – 5 kakva je, po Vama, zastupljenost tematskih oblasti¹⁵ u nastavnom planu nivoa obrazovanja na kome Vi držite nastavu:

a. Informatička (Domenska) znanja

Tematske oblasti	Značaj u
Osnove informatike	
Računarski hardver	
Osnove rada u operativnom sistemu sa grafičkim interfejsom	
Programska podrška računara	
Softverske aplikacije za rad sa tekstom	
Softverske aplikacije za rad sa prezentacijama	
Softverske aplikacije za rad sa tabelama	
Matematičke osnove računara	
Hardverski dizajn	
Operativni sistemi	
Osnove računarskih mreža	
Osnove Interneta i korišćenje Interneta	
Standardi za elektronski podržano učenje	
Kompjuterska etika, pitanja legalnosti, bezbednosti i zaštite podataka	
Računarska grafika	
Multimedija	
Tipovi i strukture podataka	
Algoritmi	
Rešavanje problema pomoću računara	
Imperativno programiranje - naredbe dodele i izrazi	
Proceduralni programski jezici	
Objektno-orijentisana paradigma	
UML (objedinjeni jezik za objektno modelovanje)	
Programiranje grafičkog korisničkog interfejsa - programi zasnovani na prozorima	
Integrisana razvojna okruženja	
Baze podataka	
Upiti nad bazama podataka	
Uspostavljanje lokalne računarske mreže (LAN) i povezivanje LAN na Internet	
Napredni Internet servisi i Veb tehnologije	
Programiranje statičkih veb strana (Uvod u HTML i CSS)	
Programiranje dinamičkih veb strana	
Upravljanje gotovim veb dizajn rešenjima (CMS - Content Management System, LMS – Learning Management System)	

¹⁵ Značenje ocene je: 1 – ne postoji; 2 – nedovoljno; 3 – prihvatljivo; 4 – dobro je zastupljeno; 5 – adekvatno je zastupljeno.

Tematske oblasti	Značaj u
Osnove veštačke inteligencije (i kompjuterskih simulacija)	
Računarske obrazovne igre	
Navedite neke druge teme i oblasti, ukoliko postoje, koje smatrate značajnim, a nisu ovde navedene:	

ii. Dostupnost nastavnih planova i programa (kurikuluma)

- ii.1. Ocenite ocenom 1 – 3¹⁶ da li Vam je/su dostupan/ni informatički nastavni plan/ovi i program/i koji se primenjuje/ju u obrazovnim institucijama Republike Srbije za nivo obrazovanja na kome Vi držite nastavu?
- ii.2. Ocenite ocenom 1-3¹⁷ potrebu dostupnosti nastavnog plana i programa putem Interneta
- ii.3. Ocenite ocenom 1 – 3¹⁸ potrebu Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja na kome Vi držite nastavu
- ii.4. Ocenite ocenom 1 – 3¹⁹ mogućnost Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja na kome držite nastavu
- ii.5. Ocenite ocenom 1 – 3²⁰ da li smatrate da je potreban uvid u informatičke nastavne planove i programe prethodnog nivoa obrazovanja za uspešno sprovođenje nastave informatike koju Vi držite?
- ii.6. Ocenite ocenom 1 – 3²¹ potrebu Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja koji prethodi onom nivou na kome držite nastavu
- ii.7. Ocenite ocenom 1 – 3²² mogućnost Vašeg uticaja na razvoj nastavnog plana i programa za nivo obrazovanja koji prethodi onom nivou na kome držite nastavu
- ii.8. Ocenite ocenom 1 – 3²³ da li smatrate da je potreban uvid u informatičke nastavne planove i programe sledećeg nivoa obrazovanja za uspešno sprovođenje nastave informatike koju Vi držite?

16 Značenje ocene je: 1 – nikako mi nije dostupan; 2 – dostupan mi je u štampanom obliku; 3 – dostupan mi je putem Interneta.

17 Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – neophodno je.

18 Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – neophodno je.

19 Značenje ocene je: 1 – nema mogućnosti; 2 – postoji ograničena - srednja mogućnost uticaja; 3 – potpuno mi je omogućeno učešće u razvoju nastavnog plana i programa.

20 Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – neophodno je.

21 Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – neophodno je.

22 Značenje ocene je: 1 – nema mogućnosti; 2 – postoji ograničena - srednja mogućnost uticaja; 3 – potpuno mi je omogućeno učešće u razvoju nastavnog plana i programa.

23 Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – neophodno je.

- ii.9. Ocenite ocenom 1 – 3²⁴ koliko bi Vam bio koristan softverski alat koji bi Vam omogućio pretraživanje i prikazivanje nastavnog plana i programa/kurikuluma po: ishodima; nastavnim sadržajima; znanjima i vještinama koje obezbeđuju; vezama između sadržaja i ishoda (koji sadržaji obezbeđuju ishode); nastavnim metodama; primerima organizacije nastave; nastavnim materijalima i sl.
- ii.10. Ocenite ocenom 1 – 3²⁵ koliko bi Vam bio koristan softverski alat koji bi Vam omogućio elektronsko prilaganje predloga za razvoj novih i modifikaciju postojećih nastavnih planova i programa

24 Značenje ocene je: 1– nije koristan; 2 – srednje koristan; 3 – veoma koristan.

25 Značenje ocene je: 1– nije koristan; 2 – srednje koristan; 3 – veoma koristan.

iii. Prilagođenost i usaglašenost nastavnih planova i programa (kurikuluma)

- iii.1. Označite sa X stepen strukturiranosti i sadržajnosti nastavnog plana i programa po kome Vi držite nastavu:
- a. Nastavni plan i program je nepregledan i na neadekvatan način prikazuje sadržaje
 - b. Nastavni plan i program je dobro strukturiran ali nedovoljno (ili previše) detaljno opisuje sadržaje
 - c. Nastavni plan i program nije dobro strukturiran iako na dobrom nivou detalja prikazuje sve sadržaje
 - d. Nastavni plan i program je adekvatno strukturiran i sadržaji tematskih oblasti su dovoljno detaljno prikazani
- iii.2. Ocenite ocenom 1-5²⁶ koliko Vaš trenutni nastavni plan i program odgovara aktuelnom stanju u oblasti računarstva (da li odlikava stanje u oblasti odgovarajuće nivou obrazovanja)?
- iii.3. Ocenite ocenom 1 – 5²⁷ dinamiku promena nastavnog plana i programa u odnosu na dinamiku promena u oblasti računarstva (da li se nastavni plan i program menja po dinamici koja prati promene u računarstvu?)
- iii.4. Ocenite ocenom 1 – 5²⁸ koliko je, po Vama, aktuelni nastavni plan i program prilagođen potrebama primene tehnološki podržanog učenja (da li nastavnici nauče da primenjuju tehnološki podržano učenje, da li učenici nauče da koriste alate tehnološki podržanog učenja)?
- iii.5. Ocenite ocenom 1 – 5²⁹ koliko je, po Vama, aktuelni nastavni plan i program prilagođen upotrebi novih tehnologija i elektronskih servisa (elektronske kartice, uzimanje potvrde o državljanstvu, zakazivanje posete lekaru, elektronski dnevnik, servisi koje nudi škola i sl.)?

²⁶ Značenje ocene je: 1 – uopšte ne odgovara; 2 – minimalno odgovara; 3 – srednje odgovara; 4 – odgovara stanju u oblasti; 5 – odlično je prilagođen.

²⁷ Značenje ocene je: 1 – nikako ne prati dinamiku promena u oblasti; 2 – prati dinamiku promena u oblasti sa neprihvatljivim kašnjenjem; 3 – prati dinamiku promena u oblasti sa velikim kašnjenjem; 4 – prati dinamiku promena u oblasti sa prihvatljivim kašnjenjem; 5 - prati dinamiku promena u oblasti bez kašnjenja.

²⁸ Značenje ocene je: 1 – nije nikako prilagođen; 2 – minimalno je prilagođen; 3 – zadovoljavajuće je prilagođen; 4 – dobro je prilagođen; 5 – odlično je prilagođen.

²⁹ Značenje ocene je: 1 – nije nikako prilagođen; 2 – minimalno je prilagođen; 3 – zadovoljavajuće je prilagođen; 4 – dobro je prilagođen; 5 – odlično je prilagođen.

- iii.6. Ocenite ocenom 1 – 5³⁰ koliki je *stepen usaglašenosti* nastavnih planova i programa:
- a. Osnovna – srednja škola (da li je ono što se nauči u osnovnoj školi dovoljno za nastavak obrazovanja u srednjoj školi)
 - b. Fakultet – srednja škola (da li je ono što se nauči na fakultetu dovoljno da bi se predavalo u srednjoj školi)
- iii.7. Ocenite ocenom 1 – 3³¹ koliko bi Vam bio koristan softverski alat koji bi Vam omogućio utvrđivanje usaglašenosti nastavnih planova i programa različitih nivoa obrazovanja
- iii.8. Na koji način proveravate usaglašenost realizovanog nastavnog sadržaja u odnosu na propisani nastavni plan i program?
- 1 – ne proveravam
 - 2 – upoređivanjem sa pisanim dokumentom
 - 3 – primenom računarskih alata (poput Excel-a)
 - 4 – neki drugi način, napisati koji
- iii.9. Ocenite ocenom 1 – 3³² koliko bi Vam bio koristan softverski alat koji bi Vam omogućio utvrđivanje usaglašenosti realizovanog i propisanog nastavnog plana i programa/kurikuluma po: (1) ishodima predmeta (2) nastavnim metodama (3) obrazovnim materijalima (4) trajanju kurseva i nastavnih jedinica i sl.

³⁰ Značenje ocene je: 1 – nije nikako usaglašeno; 2 – minimalno je usaglašeno; 3 – zadovoljavajuće je usaglašeno; 4 – dobro je usaglašeno; 5 – odlično je usaglašeno.

³¹ Značenje ocene je: 1 – nikako; 2 – srednje; 3 – veoma.

³² Značenje ocene je: 1 – nije koristan; 2 – srednje; 3 – veoma.

10. Dodatno/Permanentno obrazovanje nastavnika informatike

- i. Oceniti ocenom 1-3³³ koliko je, po Vama, potrebno permanentno obrazovanje za svaki od aspekata:

Aspekt	Ocena
Informatički (domenski) aspekt	
Opšti pedagoški aspekt	
Aspekt metodike nastave informatike	
Aspekt nastavne prakse	

- ii. Odgovoriti ocenom 1-4³⁴ koliko je često potrebno unapređivati znanja za svaki od aspekata:

Aspekt	Ocena
Informatički (domenski) aspekt	
Opšti pedagoški aspekt	
Aspekt metodike nastave informatike	
Aspekt nastavne prakse	

- iii. Rangirajte Vaše motive rangovima (1-5)³⁵ za permanentnim obrazovanjem:

Motivi	Rang
Pružanje učenicima najnovijih znanja iz oblasti	
Držanje nastave u skladu sa savremenom pedagoškom praksom	
Lični profesionalni napredak i sticanje viših kompetencija	
Zadržavanje posla/radnog mesta	
Potreba nametnuta od obrazovne institucije u kojoj radite i drugih nadležnih institucija	

³³ Značenje ocene je: 1 – nije potrebno; 2 – potrebno je; 3 – veoma je potrebno.

³⁴ Značenje ocene je: 1 – bar jednom u 10 godina; 2 – bar jednom u 5 godina; 3 – bar jednom u 2 godine; 4 – svake godine.

³⁵ Značenje rangova: rang 1 odgovara najmanjoj motivisanosti, rang 5 – najvećoj motivisanosti

11. Korisnost softverskih alata u nastavi

- i. Rangirajte rangom 1 do 10³⁶ (ne mogu postojati 2 jednaka ranga) korisnost sledećih softverskih alata za izvođenje nastave informatike (koji bi Vam bili najkorisniji, bez obzira da li ih imate na raspolaganju):

Softverski alati	Rang
Prevodioci za različite programske jezike	
Alati za dizajn i specifikaciju softvera	
Simulacioni alati i simulatori fenomena i uređaja	
Digitalni rečnici, enciklopedije, atlas, itd.	
Obrazovne računarske igre	
Alati za upravljanje nastavnim sadržajima (kreiranje i skladištenje digitalnih (multimedijalnih) nastavnih sadržaja; pretraživanje i preuzimanje digitalnih nastavnih sadržaja; kreiranje elektronskih kurseva)	
Alati za kolaborativno učenje (blog, facebook, itd)	
Alati za komunikaciju (nastavnik-učenik, učenik-učenik)	
Alati za proveru postignuća učenika (testiranje)	
Alati za podršku individualizovanom učenju	

- ii. Za koje softverske tehnologije smatrate da bi Vam dodatni kursevi bili najkorisniji u realizovanju nastave?

1.	
2.	
3.	
4.	
5.	

³⁶ Značenje rangova: rang 1 odgovara najnižoj oceni o korisnosti softverskog alata, dok rang 10 predstavlja najvišu ocenu za korisnost softverskog alata

Kratka biografija



- Milinko Mandić je rođen 12.06.1979. u Somboru.
- Osnovnu školu i Gimnaziju je završio u Somboru.
- Školske godine 1998/99 upisao je Fakultet tehničkih nauka u Novom Sadu; osnovne studije: Elektrotehnika i računarstvo; odsek: Energetika, elektronika i telekomunikacije; smer: telekomunikacije. Studije je završio 2004. godine sa prosekom ocena 8,89 i stekao zvanje diplomirani inženjer elektrotehnike i računarstva – master.
- Doktorske akademske studije, na Prirodno-matematičkom fakultetu u Novom Sadu, Metodika nastave prirodnih nauka, matematike i računarstva, smer: metodika nastave informatike je upisao 2009. godine. Prosek ocena na doktorskim studija je 9,33.
- Napisao je rad koji pripada problematici doktorske disertacije koji je objavljen u časopisu sa SSCI liste. Pored toga, autor je ili jedan od koautora više od 10 radova saopštenih na međunarodnim i domaćim skupovima.
- Od 2005. do 2006. godine radio je kao informatičar u Školskoj upravi Sombor. Od 2. Oktobra 2006. do danas radi na Pedagoškom fakultetu u Somboru i to najpre kao saradnik u nastavi, a od 1. decembra 2008. kao asistent. Drži časove vežbi iz sledećih predmeta: Uvod u programiranje, Praktikum 1 (programski jezik Java), Baze podataka, Modelovanje i simulacije u obrazovanju, Web portali i učenje na daljinu.

Novi Sad, 2015.

Milinko Mandić

UNIVERZITET U NOVOM SADU
PRIRODNO - MATEMATIČKI FAKULTET
KLJUČNA DOKUMENTACIJSKA INFORMACIJA

Redni broj:

RBR

Identifikacioni broj:

IBR

Tip dokumentacije: Monografska dokumentacija

TD

Tip zapisa: Tekstualni štampani materijal

TZ

Vrsta rada: Doktorska disertacija

VR

Autor: Milinko Mandić

AU

Mentor: prof. dr Mirjana Ivanović, prof. dr Zora Konjović

MN

Naslov rada: Razvoj platforme za standardizaciju obrazovanja nastavnika informatike

NR

Jezik publikacije: Srpski (latinica)

JP

Jezik izvoda: Srpski/Engleski

JI

Zemlja publikovanja: Republika Srbija

ZP

Uže geografsko područje: Vojvodina, Novi Sad

UGP

Godina: 2015

GO

Izdavač: Autorski reprint

IZ

Mesto i adresa: Novi Sad, Trg Dositeja Obradovića 4

MA

Fizički opis rada: (6/223/169/39/48/0/2)

(broj poglavlja/strana/lit. citata/tabela/slika/grafika/priloga)

FO

Naučna oblast: Metodika nastave prirodnih nauka, matematike i računarstva

NO

Naučna disciplina: Metodika nastave informatike

ND

Ključne reči: informatika, kompetencije, kurikulum, ACM K12, ontologije, usaglašavanje ontologija

PO

UDK:

Čuva se:

ČU

Važna napomena: nema

VN

Izvod: Doktorska disertacija se bavi istraživanjima koja imaju za cilj da unaprede obrazovanje nastavničkog kadra iz oblasti informatike. Predmet istraživanja disertacije je razvoj platforme koja će olakšati sinhronizaciju razvoja nastavničkog kurikuluma iz informatičke oblasti sa aktuelnim srednjoškolskim informatičkim standardima i savremenim kurikulumima za nastavnike informatike. Prvi pravac istraživanja je identifikovanje stanja u oblasti obrazovanja nastavnika informatike (modaliteti obrazovanja nastavnika i kompetencije koje se tim modalitetima stižu). U disertaciji se stanje identifikuje na osnovu analize savremene literature, aktuelnih standarda, reprezentativnih kurikuluma i preliminarnog istraživanja nad populacijom sadašnjih nastavnika informatike na teritoriji Vojvodine. Drugi pravac istraživanja je obezbeđivanje (tehničke) pretpostavke za objektivnije i efikasnije kreiranje, upoređivanje i razmenu kurikuluma. Taj pravac se odnosi na razvoj mašinski čitljive ontologije kurikuluma bazirane na već postojećim standardima u domenu obrazovanja i softverske podrške za upoređivanje takvih reprezentacija kurikuluma bazirane na tehnologijama Semantičkog Weba.

Ciljevi istraživanja u disertaciji su specifikacija mašinski čitljivog modela informatičkog kurikuluma za obrazovanje nastavnika informatike, mašinski čitljivog modela informatičkog kurikuluma za srednjoškolski nivo obrazovanja i razvoj prototipa softverske aplikacije za verifikaciju predloženih modela.

IZ

Datum prihvatanja teme od strane NN veća: 18.07.2014.

DP

Datum odbrane:

DO

Članovi komisije:

KO

Predsednik: dr Zoran Budimac, redovni profesor, Prirodno-matematički fakultet, Univerzitet u Novom Sadu

Mentor: dr Mirjana Ivanović, redovni profesor, Prirodno-matematički fakultet, Univerzitet u Novom Sadu

Mentor: dr Zora Konjović, redovni profesor, Fakultet tehničkih nauka, Univerzitet u Novom Sadu

Član: dr Ivan Milentijević, redovni profesor, Elektronski fakultet, Univerzitet u Nišu

Član: dr Katerina Zdravkova, redovni profesor, Fakultet za informatičke tehnologije i kompjutersko inženjerstvo, Univerzitet sveti Kiril i Metodije u Skoplju

UNIVERSITY OF NOVI SAD
FACULTY OF SCIENCE
KEY WORDS DOCUMENTATION

Accession number:

ANO

Identification number:

INO

Document type: Monograph type

DT

Type of record: Printed text

TR

Contents Code: Doctoral dissertation

CC

Author: Milinko Mandić

AU

Mentor: Ph.D. Mirjana Ivanović, Ph.D. Zora Konjović

MN

Title: Development of a platform for standardizing education of informatics teachers

XI

Language of text: Serbian (Latin alphabet)

LT

Language of abstract: Serbian/English

LA

Country of publication: Republic of Serbia

CP

Locality of publication: Vojvodina, Novi Sad

LP

Publication year: 2015

PY

Publisher: Author's reprint

PU

Publ. place: Novi Sad, Trg Dositeja Obradovića 4

PP

Physical description: (6/223/169/39/48/0/2)

(no. of chapters/pages/bib. refs/tables/figures/graphs/appendices)

PD

Scientific field: Methods of teaching science, mathematics and computing

SF

Scientific discipline: Methods of teaching informatics

SD

Key words: informatics, competence, curriculum, ACM K12, ontology, ontology alignment

UC:

Holding data:

HD Note:

Abstract: The doctoral thesis deals with researches that aim to improve the education of teaching staff in the field of informatics. The subject of the dissertation research is to develop a platform that will facilitate the synchronization of the development of teaching curricula in the informatics field with the current secondary informatics standards and modern curricula for informatics teachers of informatics. The first research direction is to identify the state in the informatics teacher education (modalities of teacher education and competences acquired through these modalities). In the dissertation, the state is identified from the review of literature, current standards, representative curricula and a preliminary research on population of current informatics teachers in Vojvodina. Another research direction is to provide (technical) prerequisites for more objective and more efficient creation, comparison and exchange of curricula. That direction is related to the development of a machine-readable ontological representation of a curriculum based on existing standards in the field of education and to developing software support for comparing such representations of curricula based on the Semantic Web technologies.

The goals of the research in the dissertation are to specify a machine readable model of the informatics teacher education curriculum, a machine-readable model of the informatics curriculum for secondary level education and the development of a software application prototype in order to verify the proposed models.

AB

Accepted by the Scientific Board on: 18.07.2014.

Defended:

Thesis defend board:

President: Ph.D. Zoran Budimac, full professor, Faculty of Science, University of Novi Sad

Mentor: Ph.D Mirjana Ivanović, full professor, Faculty of Science, University of Novi Sad

Mentor: Ph.D Zora Konjović, full professor, Faculty of Technical Sciences, University of Novi Sad

Member: Ph.D Ivan Milentijević, full professor, Faculty of Electronic Engineering, University of Niš

Member: Ph.D Katerina Zdravkova, full professor, Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje