PRILAGODLJIVI TUTORSKI SUSTAV ZA E-UČENJE PODRŽAN DUBINSKOM ANALIZOM PODATAKA

Jugo, Igor

Doctoral thesis / Disertacija

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:195:040148

Rights / Prava: Attribution-NonCommercial-NoDerivs 3.0 Unported/Imenovanje-Nekomercijalno-Bez prerada 3.0

Download date / Datum preuzimanja: **2024-03-28**

Sveučilište u Rijeci
Fakultet informatike i digitalnih tehnologija

Repository / Repozitorij:

Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository

DIGITALNA KNJIŽNICA

Image not found or type unknown

SVEUČILIŠTE U RIJECI
ODJEL ZA INFORMATIKU

Igor Jugo

PRILAGODLJIVI TUTORSKI SUSTAV ZA E-UČENJE PODRŽAN DUBINSKOM ANALIZOM PODATAKA

DOKTORSKI RAD

Rijeka, 2016.

SVEUČILIŠTE U RIJECI
ODJEL ZA INFORMATIKU

Igor Jugo

PRILAGODLJIVI TUTORSKI SUSTAV ZA E-UČENJE PODRŽAN DUBINSKOM ANALIZOM PODATAKA

DOKTORSKI RAD

Mentor: doc. dr. sc. Božidar Kovačić

Rijeka, 2016.

UNIVERSITY OF RIJEKA
DEPARTMENT OF INFORMATICS

Igor Jugo

ADAPTIVE TUTORING E-LEARNING SYSTEM SUPPORTED BY DATA MINING

DOCTORAL THESIS

Rijeka, 2016.

Mentor rada: doc. dr. sc. Božidar Kovačić

Doktorski rad obranjen je dana 07. lipnja 2016. na Odjelu za informatiku Sveučilišta u Rijeci, pred povjerenstvom u sastavu:

1. Prof. dr. sc. Nataša Hoić-Božić,
2. Prof. dr. sc. Mile Pavlić,
3. Doc. dr. sc. Marina Ivašić-Kos,
4. Prof. dr. sc. Mario Radovan,
5. Prof. dr. sc. Dragan Čišić.

Najljepše se zahvaljujem mom mentoru, doc. dr. sc. Božidaru Kovačiću, na svesrdnoj stručnoj i osobnoj podršci tijekom izrade ovog doktorskog rada.

Također se iskreno zahvaljujem svim kolegama s Odjela za informatiku Sveučilišta u Rijeci koji su svojim savjetima i podrškom pridonijeli kvaliteti ovog rada.

Zahvaljujem se i svim članovima moje obitelji koji su mi sve ove godine pružali bezuvjetnu potporu.

Sažetak:

Inteligentni tutorski sustavi su računalni sustavi za učenje, razvijeni s ciljem da omoguće tutorski model učenja, koji svoju visoku efikasnost temelji na potpunoj prilagodbi procesa učenja jednom učeniku. Inteligencija, odnosno prilagodljivost sustava promjenjivim potrebama i razini znanja učenika ostvaruje se različitim pristupima, primjerice implementacijom algoritama strojnog učenja ili dubinskom analizom podataka. Inteligentni tutorski sustavi većinom se razvijaju za dobro definirane domene znanja, kao što su matematika, fizika i druge. Međutim, postoje brojna područja koja nemaju dobro definirane strukture znanja, a potrebno ih je poučavati. Takva područja nazivaju se slabo definiranim domenama znanja. Pripremu procesa učenja u slabo definiranim domenama obavlja učitelj koji ga oblikuje prema vlastitom znanju i iskustvu, a realizira se putem implementirane inteligencije tutorskog sustava. Implementacijom metoda za dubinsku analizu podataka možemo u podacima o interakcijama korisnika sa sustavom otkriti korisne informacije, te ih pridodati postojećim mogućnostima sustava kako bi poboljšali njegovu učinkovitost.

U ovom radu predložen je novi model postojećeg inteligentnog tutorskog sustava koji, primjenom metoda dubinske analize podataka, sugerira korisniku korake koji mu slijede u procesu učenja, s ciljem kreiranja učinkovitijeg puta kroz domenu znanja. Ključni dijelovi unaprijeđenog tutorskog sustava su podsustavi za komunikaciju s alatima za dubinsku analizu podataka, grupiranje korisnika i otkrivanje čestih i učinkovitih puteva kroz domenu znanja. Predloženi tutorski model sustava prikazuje sugestije korisniku na početku i na kraju procesa učenja u obliku poveznica prema pojmovima u domeni znanja koje je najbolje učiti prije, odnosno nakon odabranog pojma.

Opisani model sustava implementiran je web aplikacijom nazvanom DITUS (*Department of Informatics TUtoring System*). Djelotvornost predloženog modela vrednovana je analizom rezultata kontrolne i eksperimentalne skupine.

Ključne riječi: inteligentni tutorski sustavi, e-učenje, dubinska analiza podataka, prilagodljivost sustava za e-učenje, otkrivanje uzoraka u sekvencama, razredovanje, sugestije.

Abstract

Inteligent tutoring systems are e-learning systems, developed with the goal of emulating the tutoring teaching model, a highly efficient learning environment based on complete adaptation of the learning process to the needs of one student. The goal of such systems is that the computer system behaves in an intelligent way, e.g. that it adapts to the current needs and knowledge level of the student. This is achieved in various ways such as the application of artificial intelligence or data mining methods. Intelligent tutoring system are developed mostly for teaching in well defined knowledge domains such as mathematics, physics, etc., but there are many other not so well defined areas that also need to be taught. These areas are called ill-defined domains, where the teacher (expert) prepares the learning process based on his/hers knowledge and experience as well as the current „intelligence“ implemented in the system. By applying data mining methods to data about students' interactions with the system we can discover useful information about their learning processes and combine this information with the current system functionality in order to improve it's efficiency.

This thesis proposes a new model of an existing intelligent tutoring system which, through the implementation of aforementioned methods, offers the student suggestions on which steps to learn next in order to streamline his/hers path through the knowledge domain. The key components of the presented system are: a communication layer for communication with data mining tools, a clustering model discovery and selection module and a high-utility frequent sequential patterns discovery module. The new tutoring model of the intelligent tutoring system then offers suggestions to the user in the form of hyperlinks to other knowledge units that are best learned before or after the unit the student has selected.

The described system is implemented as a web application called DITUS (Department of Informatics TUtoring System). The validitiy of the proposed model was verified through an experiment wich determined its viability and efficiency.

Keywords: Intelligent tutoring systems, e-learning, data mining, e-learning system adaptivity, sequential pattern mining, clustering, learning suggestions

Sadržaj

1 Uvod 1

1.1 Motivacija 2

1.2 Ciljevi, hipoteze i znanstveni doprinosi 3

1.3 Struktura rada 4

2 Dubinska analiza podataka u obrazovanju 7

2.1 Pojam dubinske analize podataka 7

2.2 Ciljevi dubinske analize podataka u obrazovanju 9

2.3 Povezivanje sustava za učenje i alata za DAP 12

2.4 Primjena metoda DAP u sustavima za učenje 14

2.4.1 Klasteriranje 14

2.4.2 Otkrivanje čestih uzoraka 15

2.5 Inteligentni tutorski sustavi 17

2.5.1 Sustav učenja na daljinu zasnovan na dijalogu 18

2.5.2 Predložena proširenja arhitekture za poboljšanje učinkovitosti 21

3 Modeliranje sustava 25

3.1 Podsustav za komunikaciju s alatima za DAP 26

3.1.1 Alati za dubinsku analizu podataka 27

3.1.2 Model podsustava 27

3.2 Podsustav za klasteriranje korisnika 28

3.2.1 Kreiranje značajki za provedbu klasteriranja korisnika 30

3.2.2 Izrada skupa modela klasteriranja 35

3.2.3 Postupak procjene kvalitete i odabira modela 36

3.2.4 Postupak procjene kvalitete procesa učenja odabranog modela 38

3.3 Podsustav za otkrivanje čestih i učinkovitih putanja korisnika 43

3.3.1 Model procjene učinkovitosti putanja korisnika 44

3.3.2 Oblikovanje skupa putanja korisnika 47
3.3.3 Postupak otkrivanja čestih i učinkovitih putanja 49
3.4 Unaprijeđeni tutorski model sustava 52
4 Implementacija sustava 55
4.1 Baza podataka 57
4.1.1 Upravljanje korisnicima 57
4.1.2 Upravljanje sadržajima u domenama znanja 58
4.1.3 Pohrana podataka o provjerama znanja 58
4.1.4 Pohrana podataka o aktivnostima korisnika 60
4.1.5 Pohrana podataka o DAP i sugestijama 62
4.2 Arhitektura sustava 64
4.2.1 Implementacija podsustava za komunikaciju s alatima za DAP 65
4.2.2 Implementacija podsustava za klasteriranje korisnika 67
4.2.3 Implementacija podsustava za analizu čestih putanja 69
4.2.4 Implementacija poboljšanog tutorskog modela 71
4.3 Sučelja sustava 75
4.3.1 Sučelja administratora sustava 76
4.3.2 Sučelja za popunjavanje domene znanja 77
4.3.3 Sučelja za provedbu procesa učenja 82
4.3.4 Sučelja za analizu podataka o učenju 88
5 Verifikacija rada sustava 95
5.1 Verifikacija rada podsustava za klasteriranje studenata 96
5.1.1 Verifikacija postupka kreiranja skupa modela grupiranja 98
5.1.2 Verifikacija postupka odabira modela 99
5.1.3 Verifikacija postupka interpretacije kvalitete procesa učenja 101
5.2 Verifikacija rada podsustava za okrivanje čestih i učinkovitih putanja 103

5.3 Verifikacija rada unaprijeđenog tutorskog modula 106
6 Opis istraživanja 109
6.1 Sudionici 109
6.1.1 Podjela sudionika u grupe 109
6.2 Domene znanja 111
6.2.1 Domena „Osnovne web tehnologije“ 111
6.2.2 Domena „Uzorci dizajna“ 114
6.3 Provjera znanja 116
6.4 Provjera zadovoljstva korisnika 116
6.5 Analiza učinkovitosti sustava analizom prikupljenih podataka 117
7 Vrednovanje sustava 119
7.1 Vrednovanje djelotvornosti sustava provjerama znanja 119
7.1.1 Domena „Osnovne web tehnologije“ 119
7.1.2 Domena „Uzorci dizajna“ 121
7.2 Vrednovanje sustava ispitivanjem zadovoljstva korisnika 124
7.2.1 Domena „Osnovne web tehnologije“ 124
7.2.2 Domena „Uzorci dizajna“ 127
7.3 Vrednovanje učinkovitosti sustava 129
7.3.1 Domena „Osnovne web tehnologije“ 129
7.3.2 Domena „Uzorci dizajna“ 133
7.3.3 Korelacija broja praćenih sugestija i aktivnosti učenja 138
8 Zaključak 143
Literatura 147
Popis slika 154
Popis tablica 157
Popis privitaka 159

Privitak 1. Test znanja za domenu „Osnovne web tehnologije“ 159
Privitak 2. Test znanja za domenu „Uzorci dizajna“..... 165
Privitak 3. Anketa za provjeru zadovoljstva korisnika sustavom DITUS..... 172
Privitak 4. Rezultati ankete za domenu „Osnovne web tehnologije“..... 175
Privitak 5. Rezultati ankete za domenu „Uzorci dizajna“ 180
Životopis..... 185

1 Uvod

U svijetu u kojem je napredak u svim sferama ljudskog djelovanja toliko brz te se govori o tome kako se cjelokupno ljudsko znanje udvostručuje svake godine, cjeloživotno obrazovanje postaje nužnost koju prihvaća sve veći broj ljudi. Istovremeno se, uslijed snažnog širenja i interdisciplinarnosti područja znanja pojavljuje problem nedovoljnog broja stručnjaka u području cjeloživotnog obrazovanja koji mogu i žele poučavati druge. World Wide Web (dalje: Web) je omogućio globalizaciju poučavanja, omogućivši stručnjacima da učine svoje materijale, komentare, zapise i zadatke dostupne svima, odnosno da pomoću sustava za e-učenje, poučavaju veliki broj fizički udaljenih korisnika takvih sustava. Posljednjih godina razvijen je velik broj platformi za e-učenje (Rosenberg, 2001) koji ima za cilj omogućiti što većem broju korisnika stjecanje novih znanja. Međutim, kod većine tih platformi pojavio se problem jednoobraznosti, odnosno nemogućnosti prilagodbe razinama znanja, iskustvima, motivaciji i navikama individualnih korisnika. Svaka osoba uči na neki svoj, specifičan način te pristupa takvim sustavima iz jedinstvene perspektive svojih prethodnih znanja i iskustava te se njezin način i vrijeme savladavanja prikazanih sadržaja, ali i cjelokupne domene znanja, razlikuje od pristupa svih drugih korisnika. Kroz povijest, najefikasnija metoda prijenosa znanja bila je individualno poučavanje, karakterizirana odnosom u kojem učitelj (tutor) kroz svakodnevnu direktnu interakciju pomaže svom učeniku pri usvajanju sve složenijih znanja i/ili vještina. Takav je proces potpuno prilagođen jednom korisniku – učeniku. Taj je ideal nemoguće postići u modernom svijetu u kojem Webu svakodnevno pristupa nekoliko milijardi ljudi, a pojedine tečajeve na poznatim otvorenim (eng. *Massive Open Online Courses* – MOOCs) (Fini, 2009) platformama upisuju stotine tisuća ljudi. Jedno od rješenja tog problema, koje se proučava i razvija već dugi niz godina, i prije pojave Weba, je personalizacija, odnosno razvoj sustava za poučavanje koji imaju mogućnosti da neki(e) aspekt(e) svoje funkcionalnosti prilagode korisniku. Jedna takva vrsta sustava su inteligentni tutorski sustavi (Corbett, 1997).

Inteligenti tutorski sustavi imaju za cilj, primjenom informacijsko kumunikacijske tehnologije te metoda umjetne inteligencije i strojnog učenja, implementirati tutorski model poučavanja. Takvi sustavi pokazali su se vrlo efikasnima (Bloom, 1984; Cohen, 1982). S razvojem globalne Internet infrastrukture te Web standarda i tehnologija, inteligentni tutorski sustavi počinju se pojavljivati i na Webu.

1.1 Motivacija

Pored složenosti razvoja prilagodljivih sustava za e-učenje i velike količine vremena potrebne za pripremu materijala za poučavanje kojima će manipulirati sustav, kod razvoja inteligentnih tutorskih sustava (dalje: ITS) za Web pojavljuje se i problem poboljšanja interakcije, koja je donedavno bila ograničena na unos podataka u za to predviđena polja (obrasce). Nadalje, pojavljuje se i problem praćenja interakcije korisnika sa sustavom, te kontrole njegovih akcija, koji je prouzročen temeljem Web usluge – *HyperText Transfer Protokolom* (HTTP, 2016) (dalje: HTTP). Sustav temeljen na HTTP-u koristi klijent-poslužitelj model komunikacije i ne čuva stanje o nizu interakcija, već je razvijen s ciljem da primljeni zahtjev pošalje na obradu poslužitelju i vrati dobivene rezultate korisniku. Iz navedenih razloga broj ITS sustava razvijen za korištenje putem Weba je puno manji od „klasičnih“ (Brusilowsky & Peylo, 2003).

Inteligentni tutorski sustavi se najčešće razvijaju za domene znanja definirane strogim pravilima, kao što su matematika, fizika, kemija ili, u području računarskih znanosti, SQL jezik (Mitrović, 2003). Pored takvih domena, postoji potreba za razvojem sustava za e-učenje u domenama koje nemaju strogo definiranu strukturu niti redoslijed usvajanja znanja. Takve domene nazivaju se slabo definiranima (engl. *ill-defined domains*) (Aleven, et al., 2006; Lynch, 2006; Fournier-Viger, et al., 2010). Redoslijed savladavanja pojmova te najbolji način organizacije sadržaja i pojmova je kod takvih domena nepoznat i temelji se na pretpostavkama, znanju i iskustvu eksperta koji izrađuje domenu. U cilju poboljšanja takvog sustava sve se češće primjenjuje dubinska analiza podataka prikupljenih tijekom interakcije korisnika sa sustavom. Dubinskom analizom podataka moguće je otkriti pravila ili uzorke ponašanja koji se mogu koristiti za poboljšanje rada sustava. Pri tome se, kao jedan od ključnih problema, ističe odvojenost samog sustava za e-učenje i alata za dubinsku analizu podataka, te nemogućnost korištenja takvih sustava osobama bez ekspertnog znanja. Ističe se potreba integracije takvih alata i sustava za e-učenje kako bi se dobivene informacije mogle iskoristiti za poboljšanje sustava (Romero & Ventura, 2007). Kao još jedan važan nedostatak može se istaknuti i vremenski odmak između perioda u kojem korisnici koriste sustava (kreiraju podatke) i perioda u kojem se, nakon provedbe dubinske analize podataka, dobiveni rezultati (modeli) implementiraju u sustav. Taj vremenski odmak često traje nekoliko mjeseci ili cijelu akademsku godinu. U ovom doktorskom radu dan je prijedlog rješenja spomenutih problema razvojem vlastitog modela i prototipa sustava za e-učenje podržanog dubinskom

analizom podataka, u kojem se dobivene informacije koriste za poboljšanje prilagodljivosti sustava korisniku, odnosno poboljšanje učinkovitosti procesa učenja.

1.2 Ciljevi, hipoteze i znanstveni doprinosi

Cilj ovog rada je povećanje učinkovitosti inteligentnog tutorskog sustava unaprjeđenjem njegove prilagodljivosti korisniku. Poboljšanje prilagodljivosti omogućeno je razvojem novog tutorskog modela podržanog dubinskom analizom podataka, koji pomaže korisniku kreirati učinkovitije putanja učenja, odnosno poboljšava učinkovitost procesa učenja. Razvijeni su postupci za procjenu utjecaja pronađenih čestih uzoraka putanja studenata na napredak studenata u domeni znanja, na temelju čega je unaprijeđen tutorski model sustava.

Pored navedenog, razvijena je i metoda za automatsko otkrivanje i interpretaciju klastera (engl. *cluster*) korisnika sustava s ciljem povećanja osjetljivosti sustava za otkrivanje čestih uzoraka. Ova metoda omogućuje da se obogati model studenta, rangiraju otkriveni klasteri prema procjenjenoj kvaliteti procesa učenja, te proces otkrivanja čestih uzoraka provede po klasterima, što omogućava da sugestije ne budu jednake za sve korisnike, već prilagođene pojedinom klasteru.

U radu su postavljene dvije istraživačke hipoteze:

H1: Dubinska analiza putanja tutorskog sustava može se koristiti za povećanje prilagodljivosti tutorskog sustava.

Hipoteza će se smatrati potvrđenom ukoliko se, prema razvijenom modelu, izradi prototip sustava koji na funkcionalan i korisniku koristan način primjenjuje metode dubinske analize putanja korisnika kreiranih prilikom savladavanja domene znanja, korištenjem tutorskog sustava. Potvrđenost ove hipoteze opisana je u poglavlju 5.3.

H2: Povećanje prilagodljivosti tutorskog modela povećava učinkovitost učenja pomoću inteligentnog tutorskog sustava.

Hipoteza će se smatrati potvrđenom ukoliko se, nakon provedbe eksperimenta, utvrdi poboljšanje učinkovitosti ključnih indikatora kvalitete procesa učenja eksperimentalne skupine u odnosu na kontrolnu skupinu. Potvrđenost ove hipoteze opisana je u poglavlju 7.3.

Prilikom izrade ovog rada postignuti su sljedeći znanstveni doprinosi:

- Definiran je skup podataka koji se koristi za pronalazak čestih putanja u tutorskom sustavu.
- Razvijen je vlastiti algoritam za evaluaciju čestih putanja uz pomoć kojeg se putanje rangiraju prema utjecaju na promjenu razine znanja studenta.

- Definiran je novi model tutorskog sustava podržan dubinskom analizom podataka.
- Razvijen je vlastiti algoritam za evaluaciju kvalitete procesa učenja svake skupine.
- Izrađen je prototip prilagodljivog tutorskog sustava podržanog dubinskom analizom podataka u kojem su implementirani predloženi modeli i algoritmi te je provedena verifikacija njegove učinkovitosti u stvarnom obrazovnom okruženju.

1.3 Struktura rada

U prvom, uvodnom poglavlju iskazana je motivacija za opisano istraživanje, naveden cilj i očekivani znanstveni doprinosi rada te kratak prikaz strukture rada po poglavljima.

Drugo poglavlje daje pregled područja istraživanja, te glavna područja primjene dubinske analize podataka u sustavima za e-učenje. Također se daje pregled trenutnog stanja i pravaca budućeg razvoja inteligentnih tutorskih sustava, analiziraju se nedostatci postojećih sustava i ističu ključne razlike u načinu provedbe postupaka dubinske analize podataka i implementacije dobivenih rezultata u sustav za učenje. Ističu se funkcionalnosti ranije razvijenog sustava za učenje. Iznosi se prijedlog nove arhitekture sustava s ciljem unaprjeđenja prilagodljivosti odnosno učinkovitosti sustava.

U trećem poglavlju prikazan je vlastiti model nove arhitekture tutorskog sustava podržanog dubinskom analizom podataka. Sustav se sastoji od nekoliko ključnih komponenti: podsustava za komunikaciju s alatima za dubinsku analizu podataka, podsustava za klasteriranje korisnika i podsustava za otkrivanje čestih i učinkovitih putanja korisnika kroz domenu znanja. Opisi podsustava uključuju odabrane algoritme dubinske analize podataka, (algoritme za) pripremu značajki, oblikovanje skupova podataka i interpretaciju odnosno obradu rezultata. Rezultat rada ovih dvaju podsustava omogućuje novom tutorskom modelu odabir i prezentaciju sugestija korisnicima u cilju poboljšanja učinkovitosti njihovog procesa učenja.

Četvrto poglavlje opisuje postupke implementacije predloženog modela sustava kao prototipa sustava pod nazivom DITUS (*Department of Informatics TUtoring System*). Uz opis modela baze podataka te arhitekture sustava, navedeni su detalji implementacije opisanih algoritama i komunikacije s alatima za dubinsku analizu podataka. Prikazana su ključna sučelja sustava za studente, profesore i administratore sustava.

U petom poglavlju provedena je verifikacija rada sustava. Analiziran je rad ključnih podsustava, te dan pregled provedenih analiza uz isticanje uočenih prednosti i nedostataka.

U šestom poglavlju daje se plan i metodologija istraživanja te opis korištenih domena znanja, s pregledom sadržaja, pitanja i strukture.

Sedmo poglavlje posvećeno je vrednovanju sustava. Sustav se vrednuje s tri aspekta – djelotvornosti sustava (rezultatima testova znanja), zadovoljstva korisnika (anketom) te analizom učinkovitosti sustava (primjenom metoda deskriptivne statistike i vizualizacije podataka).

Osmo poglavlje donosi zaključke i pregled smjerova daljnjeg razvoja svih elemenata sustava.

2 Dubinska analiza podataka u obrazovanju

U ovom poglavlju dan je uvod u dubinsku analizu podataka (eng. *data mining*) (dalje: DAP) i mogućnosti primjene DAP u obrazovanju odnosno sustavima za e-učenje. Zasebno se opisuju ciljevi istraživanja i trenutno stanje u području primjene DAP u obrazovanju odnosno sustavima za e-učenje. U posljednjem podpoglavlju opisan je sustav za učenje temeljen na dijalogu koji je teorijska osnova ovog istraživanja, odnosno sustav čija se arhitektura proširuje s ciljem poboljšanja njegove prilagodljivosti i učinkovitosti.

2.1 Pojam dubinske analize podataka

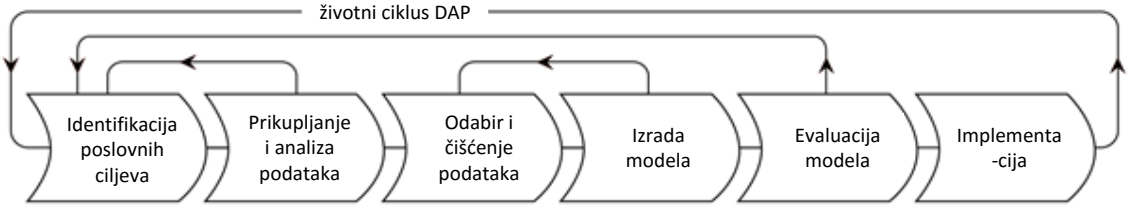
Dubinska analiza podataka je naziv za interdisciplinarno polje računarskih znanosti koje se bavi otkrivanjem uzoraka, pravila, funkcija i drugih oblika učenja, u velikim skupovima podataka primjenom metoda iz više različitih područja – strojnog učenja, umjetne inteligencije, statistike i drugih (Witten, et al., 2011). Opći cilj dubinske analize podataka je otkrivanje korisnih informacija iz skupa (ili skupova) podataka. Često se definira i kao samo jedan korak (korak analize) u složenijem procesu nazvanom „otkrivanje znanja u podacima“ (Fayyad, et al., 1996) koji obuhvaća:

- upravljanje podacima i bazama podataka,
- predprocesiranje i transformaciju podataka,
- izradu modela (analiza),
- evaluaciju i primjenu (vizualizacija, korištenje otkrivenih informacija u odlučivanju ili implementacija dobivenih modela u različitim informacijskim sustavima) rezultata.

Sam postupak analize odnosi se na pokretanje postupaka automatske ili poluautomatske analize podataka odnosno pokretanje alagoritama nad skupom podataka, pri čemu rezultat (izlaz analize) ovisi o vrsti pokrenutog algoritma i može biti otkrivanje skupina, uzoraka, pravila, funkcija (predviđanje, klasifikacija), itd. Vrlo često se rezultati jedne analize koriste kao ulazni skup (ili dio ulaznog skupa) podataka za sljedeću analizu.

Provedbu procesa dubinske analize podataka pokušalo je standardizirati više skupina istraživača i institucija, pa su tako nastali prijedlozi općenitih standarda *Cross Industry Standard Process for Data Mining – CRISP-DM* (Chapman, 2000) i *Sample, Explore, Modify, Model and Assess – SEMMA* (Azevedo, 2008), te specijalizirani standardi vezani uz pojedina razvojna okruženja kao što je *Java Data Mining – JDM* (Hornick, 2010) ili standard za opisivanje modela *Predictive Model Markup Model – PMML* (Guazzelli, 2009) temeljen na

Xtensible Markup Language (Bray, 2016) (XML) standardu. Iako niti jedan od spomenutih nije usvojen kao službeni standard od strane najvećih svjetskih standardizacijskih tijela, CRISP-DM se prema broju referenci i anketama (Piatetsky-Shapiro, 2014) ističe kao najraširenij. CRISP-DM opisuje faze procesa provedbe dubinske analize podataka. Proces je opisan kao iterativan i dijeli se u šest faza koje prikazuje Slika 1.



Slika 1. Životni ciklus dubinske analize podataka prema CRISP-DM modelu (prilagođeno prema (Chapman, 2000))

Prvi prijedlog standarda objavljen je 1996. godine, a zadnji sastanci skupine stručnjaka, istraživača i predstavnika proizvođača softvera (primarno IBM) održani su 2009. godine.

Identifikacija poslovnih ciljeva	Prikupljanje i analiza podataka	Odabir i čišćenje podataka	Izrada modela	Evaluacija modela	Implementacija
Određivanje poslovnih ciljeva *Motivacija* *Poslovni ciljevi* *Kriteriji poslovnog uspjeha* **Procjena situacije** *Popis resursa, zahtjeva, pretpostavki i ograničenja* *Terminologija* *Analiza troškova/dobiti* **Određivanje ciljeva DAP** *Ciljevi primjene DAP* *Kriteriji uspjeha DAP* **Plan projekta** *Izrađen plan* *Inicijalna procjena* alata i tehnika	**Početno prikupljanje podataka** *Izvješće o prikupljenim podacima* **Opis (skupa) podataka** *Izvješće o opisu (skupa) podataka* **Analiza podataka** *Izvješće o analizi podataka* **Verifikacija kvalitete podataka** *Izvješće o kvaliteti podataka*	**Odabir podataka** *Razlozi za odabir/odbacivanje podataka* **Čišćenje podataka** *Izvješće o čišćenju podataka* **Izrada (novih) podataka/značajki** *Kreirane značajke* *Kreirani zapisi* **Integracija podataka** *Spojeni podaci* **Oblikovanje podataka** *Preoblikovani skup podataka* *Opis skupa podataka*	**Odabir metoda DAP** *Tehnika DAP za izradu modela* *Pretpostavke izrade modela* **Dizajn testova** *Izrađen dizajn testova* **Izrada modela** *Parametri* *Modeli* *Opis modela* **Procjena modela** *Procjenjen model* *Nove postavke parametara*	**Evaluacija rezultata** *Procjena dobivenih rezultata u odnosu na kriterije poslovnog uspjeha* *Odabrani modeli* **Analiza cjelokupnog procesa** *Izvješće o analizi* **Određivanje sljedećih koraka** *Popis mogućih akcija/odlika*	**Implementacija plana** *Izrađen plan implementacije* **Praćenje i održavanje** *Izrađen plan održavanja i praćenja* **Završno izvješće** *Izrađeno završno izvješće* *Završna prezentacija* **Analiza projekta** *Iskustva*

Slika 2. Popis općenitih zadataka i rezultata prema CRISP-DM modelu (prilagođeno prema (Chapman, 2000))

Svaka faza sastoji se od skupa zadataka (eng. *tasks*) koji rezultiraju dokumentima (eng. *outputs*) kao što prikazuje Slika 2.

Proces (iteracija procesa) započinje definiranjem pitanja, problema ili cilja provedbe analize s aspekta poslovanja odnosno šireg konteksta, nakon čega se oblikuje probem na razini dubinske analize podataka. Sljedeća faza bavi se prikupljanjem ili odabirom skupa podataka, njegovom inicijalnom analizom, uočavanjem potencijalnih problema s kvalitetom podataka, itd. Treća faza obuhvaća sve potrebne postupke koji vode do konstrukcije konačnog skupa podataka odnosno popisa značajki i njihovih vrijednosti. U četvrtoj fazi se nad skupom podataka izvode različite analize, uz prilagodbu ulaznih parametara. S obzirom na specifične pretpostavke primjene različitih algoritama (npr. različitih formata zapisa podataka) moguće je ponovno izvođenje dijela postupaka iz prethodne faze. Nakon dobivanja rezultata, odnosno najkvalitetnijeg modela, potrebno je provesti njegovu evaluaciju kako bi se dokazalo da model ostvaruje željene rezultate. Često se provodi testiranje modela na novom skupu podataka, usporedba s rezultatima dobivenih iz standardnih skupova podataka ili usporedba s rezultatima drugih modela na istom skupu podataka. U posljednjoj fazi, rezultat procesa se koristi odnosno daje potvrdu ili odgovor na postavljeno pitanje, dakle može imati oblik izvještaja koji sadrži vizualizacije, ili model može biti ugrađen u neki postojeći informacijski sustav.

Postupci dubinske analize podataka su brojni, a broj algoritama i njihovih verzija stalno raste, ali se najčešće dijele u osnovnih šest skupina:

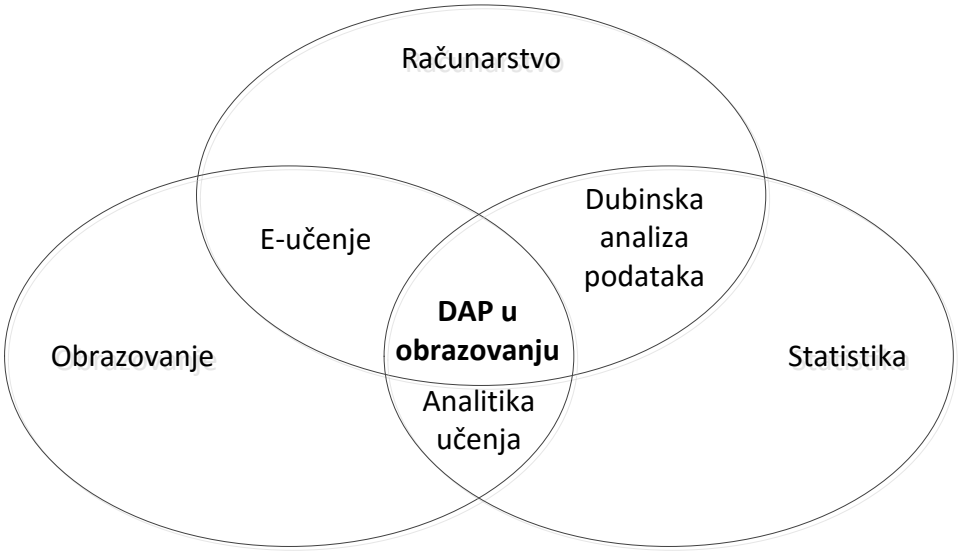
- otkrivanje iznimaka (anomalija) – eng. *outlier/anomaly detection*
- otkrivanje pravila i uzoraka – eng. *association rules, sequential pattern mining*
- klasteriranje – eng. *clustering*
- klasifikacija – eng. *classification*
- regresija – eng. *regression*
- sumarizacija

U ovom radu korišteni su postupci za klasteriranje i otkrivanje čestih uzoraka.

2.2 Ciljevi dubinske analize podataka u obrazovanju

Metode DAP brojni su istraživači počeli primjenjivati na analizu podataka (logova) iz sustava za učenje već od sredine 90-ih godina s ciljem dobivanja boljeg uvida u načine korištenja sustava od strane studenata i boljeg razumijevanja procesa učenja. Pregled većeg broja radova iz tog perioda dan je u (Koutri, 2005).

Na vodećoj znanstvenoj konferenciji o primjeni metoda umjetne inteligencije u obrazovnim sustavima (AIED, 2016)(eng. *Artificial Intelligence in Education – AIED*) 2005. godine održana je radionica pod nazivom „*Workshop on Educational Data Mining*“ čime je stvoreno novo, brzo rastuće (Romero & Ventura, 2013), interdisciplinarno područje istraživanja. DAP u obrazovanju (eng. *Educational Data Mining* – EDM) se bavi prilagodbom i primjenom postojećih metoda strojnog učenja i umjetne inteligencije na podatke iz domene poučavanja, ali i „razvojem novih metoda za istraživanje jedinstvenih tipova podataka koji proizlaze iz obrazovnog okruženja, te primjenom tih metoda za bolje razumijevanje studenata i okruženja u kojem uče“ (EDM, 2016). Na slici 3 je (na pojednostavljeni način) prikazana pozicija EDM područja u odnosu na grane istraživanja koje povezuje.



Slika 3. Venn dijagram položaja područja istraživanja EDM i LA (Romero & Ventura, 2013)

Na slici 3 prikazano je još jedno novo, interdisciplinarno područje nazvano analitika učenja (eng. *Learning Analytics – LA*). LA se u (SRI, 2012) definira kao područje koje „povezuje EDM, sociologiju, računarstvo, statistiku, psihologiju i pedagogiju, a orjentirano je na razumijevanje učenja kao cjeline, te prilagodbu obrazovanja mogućnostima i potrebama svakog učenika“. Posljednjih godina ova područja se sve više povezuju zbog sličnih općih ciljeva, te zbog kreiranja veće zajednice istraživača (Baker & Siemens, 2014).

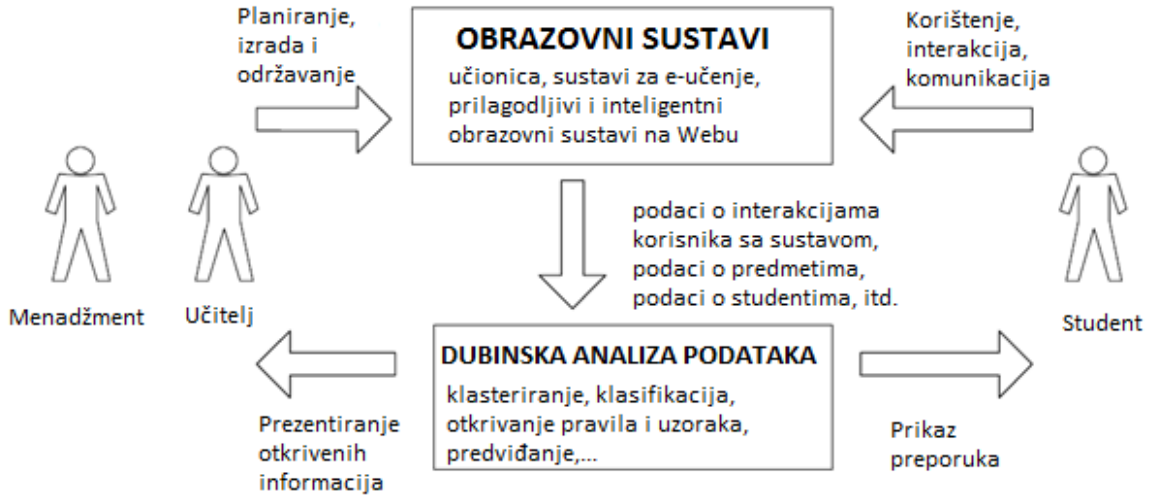
Prilagodba postojećih, kao i razvoj novih metoda u području EDM nužna je zbog:

- nove domene istraživanja (e-učenje),
- brojnosti izvora podataka o procesu učenja (manje količine podataka u odnosu na transakcijske sustave, ali više mogućih izvora – log datoteke, baze podataka, analogni izvori),
- specifičnih ciljeva (poboljšanje učenja i sustava za e-učenje).

Ključni subjekti na koje se mogu odnositi ili čijem radu mogu dopridonositi istraživanja u području EDM su:

- Studenti – dobivaju unaprijeđene sustave za e-učenje, koji se prilagođavaju potrebama studenta, sugeriraju dobre prakse, nude sugestije o načinima prolaska kroz materijale ili nude dodatne materijale
- Nastavnici – dobivaju više povratnih informacija o učenju i efikasnosti procesa učenja, mogućnosti klasifikacija učenika, pronalaženja uzoraka ponašanja i predviđanja posljedica, pronalaženja najčešćih grešaka, sugestije u prilagodbi materijala i strukture znanja, vizualizacije podataka o aktivnostima studenata
- Administratori/menadžeri u području obrazovanja – dobivaju bolji uvid u općenitu efikasnost sustava, koji se može koristiti za poboljšanje programa studija.

Kao što je prikazano na slici 4, jedan od ključnih ciljeva je poboljšanje prilagodljivosti sustava (engl. *Adaptive and Intelligent Educational Systems – AIES*) ili okruženja za učenje. Pored toga, brojni istraživači u području bavili su se analizom procesa učenja, predviđanjem uspješnosti studenata u sustavu ili na pojedinom predmetu, analizom društvenih mreža, itd. Posljednjih nekoliko godina, sve se više provode istraživanja pri kojima se podaci o aktivnostima studenata tijekom procesa učenja prikupljaju i kamerama, te sustavima osjetljivima na kretanje (akcelerometri) kako bi se izradili dinamički modeli otkrivanja emocija, dosade, motivacije ili interesa, te prema razultatima rada tih modela, omogućilo dinamičku prilagodbu procesa učenja odnosno interakcije sustava s korisnikom (Bosch, 2015).



Slika 4. Općeniti ciljevi primjene DAP u obrazovanju (Romero & Ventura, 2007)

Pregled ključnih područja primjene EDM metoda dan je u (Baker & Yacef, 2009):

- unaprijeđenje modela studenata radi unaprijeđenja učenja
- otkrivanje i unaprijeđivanje strukture domene znanja
- istraživanje pedagoške podrške (npr. suradničko učenje)
- traženje empirijskih dokaza za usavršavanje i proširivanje teorija učenja

Kao ciljeve istraživanja u području EDM autori u (Romero & Ventura, 2010) navode:

a) učiniti alate za DAP pristupačnijima nastavniku,
b) standardizacija metoda i podataka o e-učenju,
c) integracija alata za DAP i sustava za e-učenje
d) razvoj specifičnih/efikasnijih metoda DAP koje integriraju znanje iz domene obrazovanja.

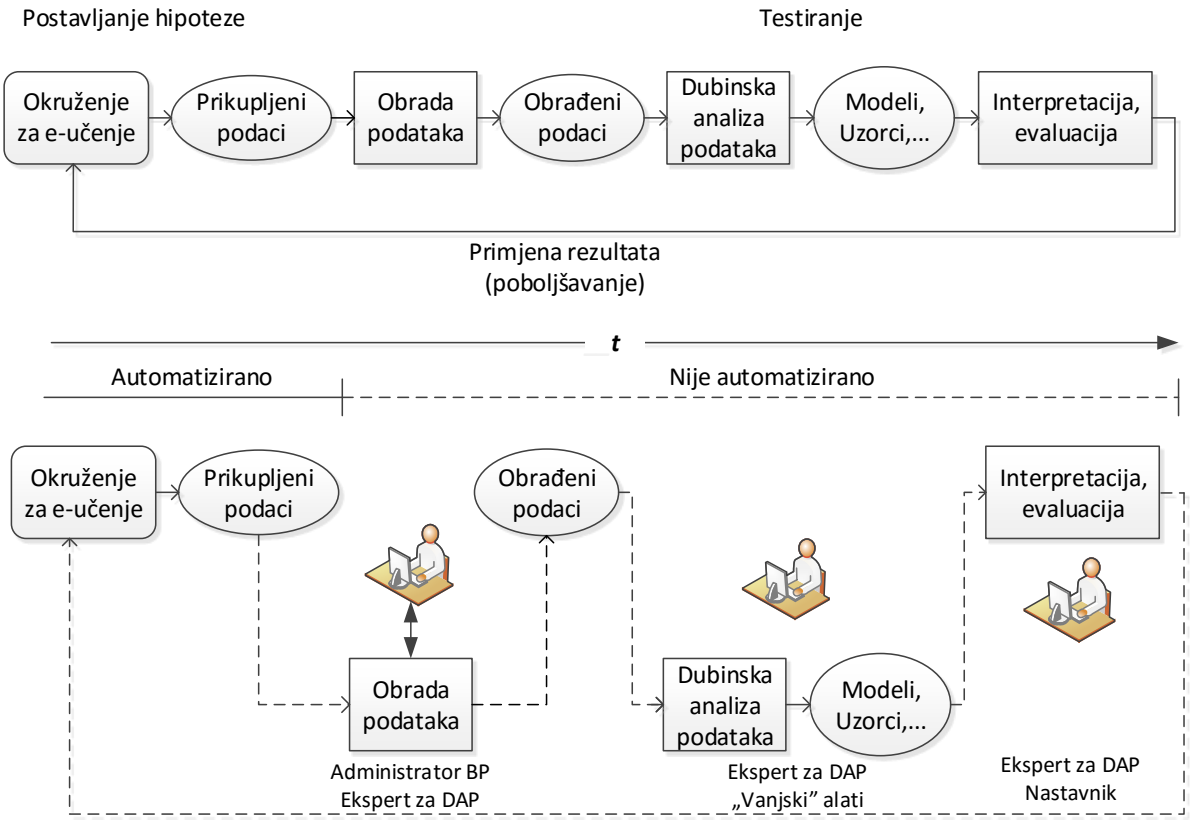
Smjerovi navedeni pod a) i c) poklapaju se s ciljevima ovog straživanja. Integracijom alata i tehnika za DAP u sustav za e-učenje, omogućava se provedba DAP periodički ili na zahtjev bilo kojeg tipa korisnika (student, nastavnik, administrator), ovisno o načinu implementacije, čime se može pozitivno djelovati na kvalitetu procesa učenja.

2.3 Povezivanje sustava za učenje i alata za DAP

Na slici 5, prikazan je životni ciklus provedbe EDM kao prilagođena verzija općeg modela prikazanog na slici 1. Iako je ciklus (na gornjem dijelu slike 5) prikazan kao neprekinuti niz koraka, on se realno odvija na način koji je prikazan u donjem dijelu slike, iz kojeg je vidljivo da je ovaj proces samo manjim dijelom automatiziran (samo faza prikupljanja osnovnih podataka o interakcijama korisnika sa sustavom/okruženjem za e-učenje). Provedba ostalih koraka zahtijeva angažman jednog ili više stručnjaka (administratora baze podataka – dohvaćanje i priprema podataka, eksperta za DAP – provedba DAP, prezentacija i interpretacija rezultata, itd.) što rezultira značajnim vremenskim odmakom između postupaka prikupljanja podataka i implementacije odnosno korištenja dobivenih rezultata u okruženje za e-učenje u svrhu njegovog poboljšanja.

Problemom integracije alata za DAP i sustava za e-učenje bavio se relativno mali broj autora. U (Zaiane & Luo, 2001) i (Zaïane, 2002) dan je teorijski model sustava koji koristi termin „integracije DAP“ u sustav za e-učenje kao primjer boljeg iskorištenja mogućnosti DAP. Umjesto analize podataka nakon procesa učenja (npr. po završetku semestra), zagovara se direktno povezivanje sustava za e-učenje i alata za DAP kako bi se analize vršile što ćešće i odmah utjecale na rezultat rada sustava za preporučivanje. U (Zorilla & Garcia-Saiz, 2013)

opisan je alat za DAP u obliku usluge za analizu podataka kojem se pristupa putem Web preglednika.



Slika 5. Općeniti model tijeka provedbe DAP u obrazovanju, prilagođeno prema (Romero & Ventura, 2013)

Alat omogućuje slanje podataka od strane korisnika, nad kojima se zatim može pokrenuti nekoliko algoritama za DAP, te generirati brojne vizualizacije. Alat se oslanja na Java/Matlab platformu te pomaže nastavnicima analizirati uzorke ponašanja studenata i navigacijske putanje prilikom rada s Moodle (Moodle, 2016) sustavom. Drugi autori bavili su se problemima povezivanja alata za DAP i vizualizaciju rezultata te njihovim prihvaćanjem od strane nastavnika. U (Ali, et al., 2012) autori su opisali razvoj alata LOCO-Analyst za vizualizaciju podataka (primjerice društvenih mreža) i njegovu primjenu u sustavu za e-učenje. Kasnije su analizirali stavove nastavnika i njihovo prihvaćanje takvog alata u praksi (Ali, et al., 2013). U (Zorrila, et al., 2010) autori opisuju arhitekturu sustava razvijenog za analizu podataka o učenju u okruženju Black Board (BlackBoard.com, 2016) koji obrađuje podatke iz sustava, pokreće DAP, te sprema dobivene razultate u bazu podataka. Nastavnici mogu koristiti spomenuti sustav kako bi analizirali aktivnosti svojih studenata. U (Romero, et al., 2013) opisana je komponenta (proširenje) za Moodle sustav koji omogućuje nastavnicima, da iz Moodle sučelja za nastavnika, pokrenu tri algoritma (klasteriranje, klasifikacija i otkrivanje pravila) nad zapisima sustava o aktivnostima studenata, nakon čega mogu preuzeti

rezultate u izvornom obliku (bez dodatnog oblikovanja ili vizualizacije). U radu nije opisano jesu li spomenuti algoritmi implementirani direktno u sustav ili je sustav povezan s alatom za DAP. Jedan od važnih dijelova modela sustava opisanog u ovom doktorskom radu, te implementiranog u obliku prototipa, čine podsustavi za komunikaciju s alatima za DAP, te provedbu EDM procesa, s minimalnim angažmanom nastavnika. Time se proces gotovo u potpunosti automatizira, a rezultati odmah koriste za poboljšavanje procesa učenja. U dijelu sučelja za nastavnika omogućena je (vizualna) analiza podataka. Na ovaj način se ranije spomenuti vremenski odmak gotovo u potpunosti uklanja.

2.4 Primjena metoda DAP u sustavima za učenje

Za ostvarenje cilja ovog istraživanja – poboljšanja učinkovitosti sustava za e-učenje primjenom metoda DAP, korišten je princip „ulančavanja modela“ dobivenih pomoću DAP (eng. *discovery with models*) (Baker R. , 2010). U ovom radu koriste se metoda klasteriranja studenata, nakon čega se nad dobivenim rezultatima primjenjuje metoda otkrivanja čestih i učinkovitih uzoraka (eng. *high-utility sequential pattern mining – SPM*) (Ahmed, 2010).

2.4.1 Klasteriranje

Klasteriranje (Kaufman, 2009) studenata se vrlo često koristi u području EDM. Postoje brojni pristupi klasteriranju: temeljeni na povezanosti (eng. *connectivity-based*), temeljeni na definiranju ili izračunu središnjih vrijednosti skupine (eng. *centroid-based*), temeljeni na distribuciji podataka (eng. *distribution-based*), temeljeni na gustoći/frekvenciji podataka (eng. *density-based*). Za svaki od spomenutih pristupa razvijeno je više različitih algoritama. Pregled problema klasteriranja, te kritičnih koraka u analizi, odnosno otkrivanju klastera dan je u (Milligan, 1989). Kod otkrivanja klastera, jedan od temeljnih problema je određivanje optimalnog broja klastera, što ima presudan utjecaj na rezultate, odnosno rezultirajući model. Određivanje broja klastera je dobro poznati problem optimizacije kojim se bavio velik broj istraživača (Xu & Wunsch, 2005). U (Gordon, 1999) dan je pregled većeg broja metoda za odabir broja klastera odnosno procjenu kvalitete kreiranog modela nakon odabira. Metode su podjeljene u dvije glavne skupine: globalne i lokalne metode. Lokalnim metodama provjerava se hipoteza da li bi dva klastera trebala biti spojena i mogu se primjenjivati samo na hijerahijski ugnježđene dijelove modela. Globalne metode, za zadani broj klastera, mjere kvalitetu modela nekim od sljedećih metoda/kriterija: metoda Calinski/Harabasz, metoda Hartigan, metoda Krzanowski/Lai, Gap metoda te metoda siluete. Učinkovitost tih metoda

analizirana je u (Tibshirani, 2001) i (Symons, 1981). Većina ovih metoda zahtjeva intervenciju u sam algoritam za klasteriranje, što nije bilo moguće u ovom istraživanju, koje ima za cilj osigurati komunikaciju s alatima za DAP bez izmjene postojećih algoritama kako bi se zadržala fleksibilnost u odabiru algoritma klasteriranja, ali i omogućila reprodukcija rezultata drugim istraživačima. Za odabir najboljeg modela klasteriranja u ovom doktorskom radu odabrana je metoda siluete (Rousseeuw, 1987). Nakon odabira najkvalitetnijeg modela, evaluacija se može provesti različitim statističkim (Bouchet, 2013) ili drugim, kompleksnijim metodama (Gordon, 1999). Kako sustav opisan u ovom radu periodički provodi postupak klasteriranja na rastućem skupu podataka, evaluacija modela na taj način nije moguća (jer ne postoji neki skup podataka i rezultata klasteriranja za koji je moguće reći da je točan odnosno referentan), već se provodi interpretacija kvalitete procesa učenja članova svakog otkrivenog klastera u odabranom modelu. Interpretacija rezultata klasteriranja uvijek ovisi o području istraživanja i prirodi analiziranih podataka. U tu svrhu razvijena je vlastita metoda kojom se provodi automatska interpretacija rezultata, odnosno procjenjuje kvaliteta procesa učenja svakog otkrivenog klastera. Na temelju procjene, utvrđuje se poredak skupina, koji se koristi u sljedećoj fazi rada sustava – otkrivanju čestih i učinkovitih putanja studenata kroz domenu znanja.

2.4.2 Otkrivanje čestih uzoraka

Otkrivanje čestih uzoraka (eng. *sequential pattern mining*), odnosno pronalaženje čestih nizova aktivnosti u skupu podataka o interakcijama korisnika sa sustavom, te povezivanje pronađenih uzoraka s klasterima korisnika bila je osnova sustava ze preporučivanje u sustavima za e-učenje većeg broja autora. U (Romero, et al., 2007) opisan je sustav koji omogućava korisniku da nad skupom podataka o aktivnostima korisnika, preuzetih iz okruženja za e-učenje, provede postupak klasteriranja, a zatim i postupak otkrivanja čestih pravila. Dobivena pravila u spomenutom sustavu pregledava nastavnik i odabire ona koja smatra korisnima za poboljšanje procesa učenja. Odabrana pravila prezentiraju se studentu kao preporučene Web stranice s materijalima za učenje. Spomenuti sustav ima nekoliko nedostataka – zahtijeva tehničko znanje korisnika potrebno za provedbu DAP, zahtjeva od nastavnika pojedinačnu analizu velikog broja otkrivenih pravila, te nije povezan sa samim sustavom za učenje. U (Klašnja-Milićević, 2011) sličan se pristup (klasteriranje se provodi po stilovima učenja), koristi kao proširenje tutorskog sustava za učenje programiranja.

Jedan od ključnih izazova kod otkrivanja čestih uzoraka/sekvenci je razlikovanje „boljih“ od „loši(ji)h“, odnosno produktivnih od neproduktivnih uzoraka (skupa aktivnosti, redoslijeda aktivnosti, putanja kroz domenu znanja, itd.) Kod većine opisanih sustava, ta faza analize podrazumijeva angažman nastavnika ili eksperta za DAP. Jedan od uzroka tome je „informacijska siromašnost“ algoritama za otkrivanje čestih uzoraka, odnosno nemogućnost prihvaćanja (kao ulaznog skupa podataka) nikakvih dodatnih informacija osim skupa sekvenci koje se sastoje od određenog broja numeričkih identifikatora objekata (primjerice učenih pojmova). Neki autori su tom izazovu pristupili složenijim postupcima predprocesiranja podataka, odnosno prilagodbom skupa ulaznih podataka u algoritam. Primjerice, u (Perera, 2009) autori su razvili više skupina oznaka za dodavanje konteksta/informacija u skup sekvenci, te na taj način dobili informativnije česte uzorke, podobnije za dobivanje uvida u njihova istraživačka pitanja (otkrivanje uzoraka aktivnosti u grupnom radu studenata na projektima razvoja softvera).

Kod prilagodljivih tutorskih sustava (ali i u ovom istraživanju) izazov predstavlja odlučivanje o odabiru slijeda aktivnosti učenja koje će omogućiti studentu da dosegne najvišu razinu kompetencija za sve koncepte ili vještine koje poučavamo ITS-om.

U do sada spomenutim, ali i brojnim drugim radovima, autori su većinom primjenjivali starije algoritme za pronalaženje čestih uzoraka kao što su AGP (Srikant, 1996) i PrefixSPAN (Pei, 2004). U posljednjih nekoliko godina, neke istraživačke grupe razvile su veći broj novih algoritama za otkrivanje čestih uzoraka ili ugradile spomenute algoritme u vlastite algoritme s ciljem da u njih uključe više informacije o kontekstu učenja. U sljedećim odlomcima ukratko su opisani njihovi doprinosi uz komentare o važnosti njihovih rezultata za ovo istraživanje.

U (Guerra, 2014) autori opisuju metodu za izgradnju kompaktnih karakteristika ponašanja studenata pri rješavanju problema na razini mikro uzoraka u inteligentnom sustavu za testiranje znanja QuizJET (Quizjet, 2016). Najčešći uzorci čine tzv. genom (stabilna karakteristika studenta koja ga razlikuje od ostalih) koji predstavlja vektor genskih frekvencija. Na temelju pronađenih genoma i klasterizacije studenata prema istome, autori žele prepoznavati i prevenirati neželjene uzorke, odnosno pomoći studentima u postizanju boljih ocjena na testovima. Ovaj smjer istraživanja je važan zbog njegove utemeljenosti na sustavu za provjeru znanja koji predstavlja osnovu za ažuriranje modela znanja studenta, odnosno jedan od ključnih elemenata arhitekture inteligentih tutorskih sustava. Razlikovanje vrste, odnosno težine pitanja i procjene utjecaja odgovora na pitanje (točnog i netočnog) na model znanja, je bitan segment ITS-a.

Za potrebe unaprjeđenja ITS-a za učenje korištenja robotske ruke koja se koristi na Internacionalnoj svemirskoj postaji, autori su u (Fournier-Viger, et al., 2008) opisali nekoliko različitih pristupa učenju čestih uzoraka uz dodavanje kontekstualnih informacija – najprije dodavanjem vremenskih intervala u bazu podataka, zatim dodavanjem vrijednosti akcija u elemente sekvenci (npr. kut zakrivljenosti zgloba ruke), te u konačnici proširivanje algoritma za višedimenzionalno učenje čestih uzoraka (Pinto, 2001) dodavanjem informacija o uspješnosti niza učinjenih akcija i razine znanja korisnika. Pronađeni uzorci koriste se za odabir akcija koje će ITS ponuditi korisniku pri zahtjevu za pomoć. Svaki niz aktivnosti nastao korištenjem ITS-a ima oznaku uspješnosti (0/1), te razinu znanja korisnika (ekspert/početnik) čime je olakšan odabir čestih uzoraka koji će se ponuditi korisniku. Spomenuti sustav nije razvijen kao Web aplikacija, već kao samostalna aplikacija koja se pokreće na računalu. Također, bitna prednost pri provedbi postupka otkrivanja čestih uzoraka je postojanje spomenutih oznaka uspješnosti i razine znanja prije početka provedbe analize čime je evaluacija rezultata znatno olakšana.

U (Kinnebrew, 2013) autori opisuju metodu za identifikaciju i usporedbu segmenata produktivnog i neproduktivnog ponašanja tijekom učenja u sustavu Betty's Brain (Brain, 2016). Metoda se temelji na novom algoritmu koji integrira postupak otkrivanja čestih uzoraka, te nad rezultatima provodi statističku analizu radi otkrivanje sekvenci produktivnih i neproduktivnih aktivnosti za svakog učenika, te omogućuje naknadnu interpretaciju dobivenih uzoraka s ciljem dobivanja boljeg uvida u načine korištenja sustava. Cilj ovog rada bilo je dobivanja boljeg uvida u način učenja, ali ne i direktna primjena dobivenih rezultata za poboljšanje rada sustava.

Rezultati istraživanja spomenutih istraživačkih grupa upućuju na nedovoljno istražene mogućnosti poboljšanja prilagodljivosti sustava zasnovane na analizi putanja učenja primjenom DAP, čime se, uz druge ciljeve, bavi ovo istraživanje.

2.5 Inteligentni tutorski sustavi

Inteligentni tutorski sustavi su računalni sustavi za podršku praćenja procesa učenja pružanjem trenutnih i prilagođenih uputa i povratnih informacija studentu, bez intervencije nastavnika. Osnovna namjena je vođenje procesa učenja na smislen i efikasan način podržan računalnim tehnologijama. Arhitekturu inteligentnog tutorskog sustava čine moduli za: sekvencioniranje kurikuluma, inteligentnu analizu testova i interaktivnu podršku rješavanju problema. Cilj sekvencioniranja kurikuluma je pružiti studentu prilagođenu sekvencu jedinica

znanja te pomoći studentu kako bi te jedinice savladao na učinkoviti način. Aktivno sekvencioniranje sadržaja temelji se na cilju učenja (podskupu koncepata u domeni koje želimo naučiti), i trenutnom znanju studenta, odnosno razlici između trenutnog znanja i postavljenog cilja. Ova tehnologija je najkorištenija u prilagodljivim sustavima za e-učenje (Beck, 2008).

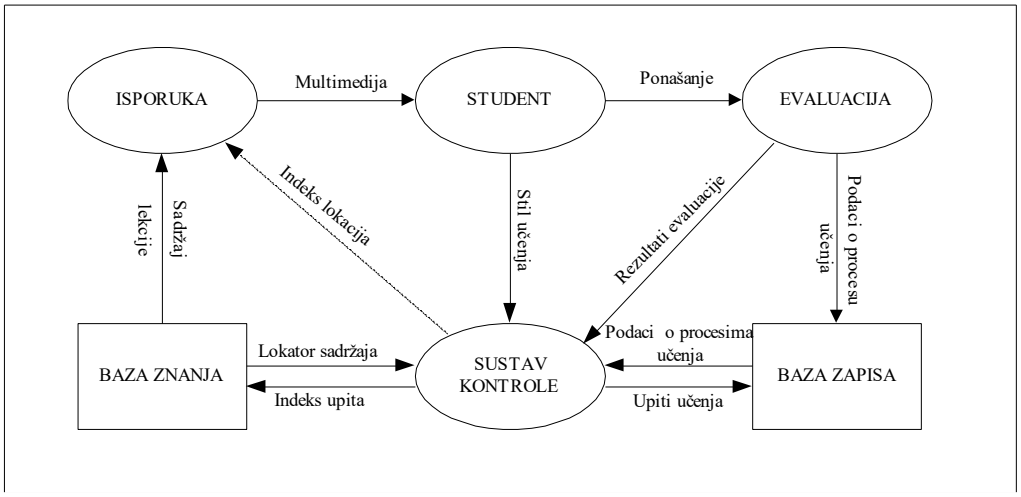
Kod slabo definiranih domena znanja, njihov sadržaj ili struktura otežava ili onemogućuje primjenu tradicionalnih pristupa koji koriste inteligentni tutorski sustavi u dobro definiranim domenama. U takvim domenama nije jednoznačno određen niti poznat redoslijed savladavanja pojedinih jedinica znanja (preduvjeti) niti je moguće razviti takve zadatke koji unaprijeđuju znanje ili vještinu vezanu za samo jedan koncept (jedinicu znanja) što su neke od ključnih pretpostavki rada inteligentnih tutorskih sustava. U ovom radu dan je prijedlog nove arhitekture prilagodljivog tutorskog sustava za poučavanje slabo definiranih domena znanja, opisanog u sljedećem poglavlju.

2.5.1 Sustav učenja na daljinu zasnovan na dijalogu

Model sustava učenja na daljinu zasnovanog na dijalogu opisan je u doktorskoj disertaciji (Kovačić, 2002). Arhitektura tog sustava se sastoji od: baze znanja, procesa za isporuku sadržaja (sučelje), studenta kao ključnog korisnika sustava, procesa evaluacija korisnikovih odgovora na postavljena pitanja, baze zapisa, te sustava kontrole. Sustav kontrole je najsloženiji element modela te ima najveću važnost za uspješnost procesa učenja. Ovaj element modela izvodi sljedeće procese:

- generira proces učenja na daljinu odabirom sadržaja učenja i prijenosom sadržaja do studenta,
- analizira učinke generiranog procesa učenja na daljinu vrednovanjem ponašanja studenta tijekom izvođenja procesa učenja,
- modificira tekući proces učenja s ciljem poboljšanja procesa učenja na daljinu, a dobivene spoznaje o procesu učenje ugrađuje u bazu zapisa,
- na osnovu unešenih podataka o tekućem i prijašnjim procesima učenja definira bitne parametre koji pridonose poboljšanju procesa učenja,
- na temelju stečenih znanja o prethodno izvedenim procesima učenja utječe na kriterije za odabir podataka iz baze znanja, što se ostvaruje postavljenjem upita na bazu znanja za traženi sadržaj.

Slika 6 prikazuje model sustava.



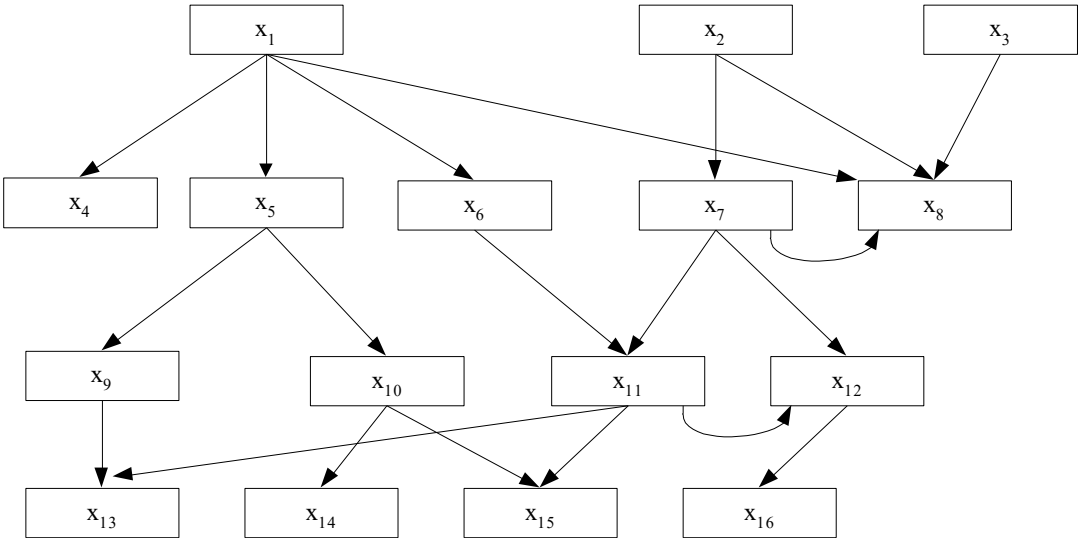
Slika 6. Model učenja na daljinu zasnovan na dijalogu (Kovačić, 2002)

Model sustava je predvidio dva ključna procesa kojima studenti vrše interakciju sa sustavom:

- UČENJE – proces koji se sastoji od dva ključna podprocesa
  - prikaza nastavnih materijala
  - provjere usvojenog znanja i znanja o povezanim pojmovima niže razine u strukturi baze znanja
- PONAVLJANJE – proces koji se sastoji od postavljanja niza pitanja za provjeru usvojenog znanja o odabranom pojmu, pri čemu duljinu niza, odnosno broj postavljenih pitanja određuje korisnik.

Domena znanja prezentirana je skupom povezanih pojmova čime se definira strukturu znanja domene. Primjer strukture domene znanja dan je na slici 7.

Nastavnik oblikuje domenu znanja u strukturu pojmova, na temelju vlastitog znanja i iskustava u korištenju ili poučavanju te domene. Dakle, struktura pojmova predstavlja pretpostavku nastavnika o „dobrom“ načinu savladavanja domene. Ne postoji unaprijed definirani najbolji put za savladavanje domene, koji bi osigurao maksimalnu učinkovitost procesa učenja u smislu broja potrebnih aktivnosti, uloženog vremena ili usvojenosti znanja. Sustav omogućuje korisnicima potpunu slobodu u odabiru pojma koji žele učiti, te ih, nakon odabira, vodi kroz procese učenja pojmova niže razine, u ovisnosti od strukture pojmova i ispravnosti odgovora na inicijalna pitanja.



Slika 7. Primjer strukture pojmova u domeni znanja (Kovačić, 2002)

Proces učenja odabranog pojma započinje odabirom i isporukom nastavnih sadržaja, analizom učinaka na korisnika provjerom usvojenog znanja odabranog pojma i pojmova niže razine odabranog pojma. Prilagodljivost sustava očituje se prilikom postupka provjere usvojenog znanja u procesu UČENJA, gdje sustav kontrole odabire pojmove niže razine prema trenutnom stanju modela studenta, odnosno ne postavlja studentu pitanja o pojmovima koje je prethodno savladao. Prilikom provjere usvojenog znanja, kod oba ključna procesa, sustav se koristi razvijenim algoritmom za odabir odgovarajućeg pitanja za ponavljanje, koji uzima u obzir trenutno stanje u modelu studenta, te prethodno prikazana pitanja, dok se kod provjere znanja o pojmovima niže razine isti algoritam koristi za odabir inicijalnih pitanja kojima se provjerava razumijevanje pojmova niže razine. Funkcija dijaloga ostvaruje se vođenjem procesa učenja u ovisnosti od ispravnosti odgovora na inicijalna pitanja. U slučaju netočnog odgovora na inicijalno pitanje, sustav će ponovno provesti cjelokupan proces učenja, ali će se kao odabrani pojam postaviti pojam kojem pripada pitanje na koji je dan netočan odgovor. Na ovaj način, proces vodi korisnika kroz strukturu pojmova baze znanja, odnosno ostvaruje dijalog s korisnikom. Sustav bilježi početni pojam procesa učenja, te se cjelokupni proces, neovisno o broju pokrenutih podprocesa učenja, završava kada su postavljena sva inicijalna pitanja odnosno izvršena provjera znanja za sve pojmove niže razine prvotno odabranog pojma.

Kod procesa PONAVLJANJA, proces učenja se pojavljuje u pojednostavljenom obliku. Proces ponavljanja u osnovi se sastoji od odabira pojma za koji student želi provjeriti znanja, nakon čega sustav prikazuje jedno po jedno pitanje na koje korisnik odgovara. Nakon toga,

korisniku se nudi mogućnost završetka procesa ili nastavka odnosno prikaza još jednog pitanja. Prilikom unosa pitanja u bazu zapisa, nastavnik može svaki od netočnih ponuđenih odogovora povezati s nekim drugim pojmom u domeni znanja, ukoliko je odabir ponuđenog odgovora indikator nedovoljnoj razumijevanja tog pojma. Ukoliko tijekom procesa ponavljanja student odabere takav odgovor, započeti će se pojednostavljeni/skraćeni proces učenja pojma povezanog s netočnim odgovorom. Sustav će studentu prezentirati nastavne materijale, nakon ćega će se studentu postaviti jedno pitanje za provjeru usvojenosti znanja. Skraćeni proces učenja ne postavlja incijalna pitanja o pojmovima niže razine, već se nakon dobivanja odgovora od strane studenta proces učenja završava, odnosno proces ponavljanja se nastavlja prezentacijom sljedećeg pitanja za ponavljanje/provjeru znanja o incijalno odabranom pojmu.

Model sustava je opisao i druge funkcionalnosti, poput postupaka odabira pitanja za provjeru, podsustava za nastavnika, koji omogućuje izradu i strukturiranje domene znanja, unos nastavnih materijala i pitanja, funkcionalnost provedbe testa znanja i druge, koji se ne opisuju detaljnije u ovom radu.

Sustav je razvijen do razine prototipa, korištenjem Java tehnologije i objektno orjentirane baze podataka.

2.5.2 Predložena proširenja arhitekture za poboljšanje učinkovitosti

U sljedećim iteracijama razvoja sustava, te provedbom kraćih eksperimenata tijekom kojih su studenti Odjela za informatiku koristili sustav za savladavanje različitih domena znanja uočeno je da sustav, pored odgovora na pitanja, ne bilježi niti koristi podatke o brojnim aspektima rada korisnika sa sustavom. Studenti mogu generirati velike količine podataka (ovisno o mogućnostima sustava da ih pohranjuje) koje je moguće iskoristiti kao izvor podataka nad kojim se mogu provesti različite analize, u cilju otkrivanja korisnih informacija. Na temelju tih informacija nastavnik može donositi odluke s ciljem poboljšanja procesa učenja ili rada samog sustava.

Kako bi se stvorili preduvjeti za korištenje podataka za poboljšanje učinkovitosti sustava, predloženo je proširenje sustava za pohranu podataka, prema primjerima dobre prakse prezentiranima u (Krüger, 2010; Mostow, 2006).

Studenti u sustavu imaju slobodu odabira pojma kojeg žele učiti i to kod svakog pokretanja bilo kojeg od dva spomenuta ključna procesa – učenja ili ponavljanja. Tijekom analize podataka, nakon provedbe niza kraćih eksperimenata, uočene su velike razlike u načinu

korištenja sustava među različitim studentima, kao i u strategijama učenja odnosno redoslijedu savladavanja pojmova u domeni znanja. Preciznije, uočene su skupine studenata koje su pristupale radu sa sustavom na različite načine, primjerice:

- detaljno proučavanje materijala, zatim odgovaranje na pitanja
- korištenje procesa učenja za vođenje kroz domenu znanja
- kombiniranje dvaju ključnih procesa za savladavanje jednog po jednog pojma,
- i dr.

Također su uočene različite strategije savladavanja domene znanja, primjerice:

- odozdo prema dolje
- odozgo prema dolje
- „grana“ po „grana“ hijerarhije pojmova (npr. X_9 i X_{13} na slici 7.)
- „nasumični“ odabir pojmova

Na temelju uočenog predložena su daljnja proširenja sustava, s ciljem korištenja prikupljenih podataka za poboljšanje učinovitosti procesa učenja:

- otkriti klastere studenata koje imaju sličan proces učenja
- rangirati sve otkrivene klastere na temelju procjene kvalitete procesa učenja studenata u klasteru
- za svaki klaster otkriti česte i učinkovite putanje kroz domenu znanja
- unaprijediti model kontrole (tutorski model) sustava na način da studentima sugerira pojmove koje bi trebali učiti kako bi poboljšali kvalitetu svog procesa učenja.

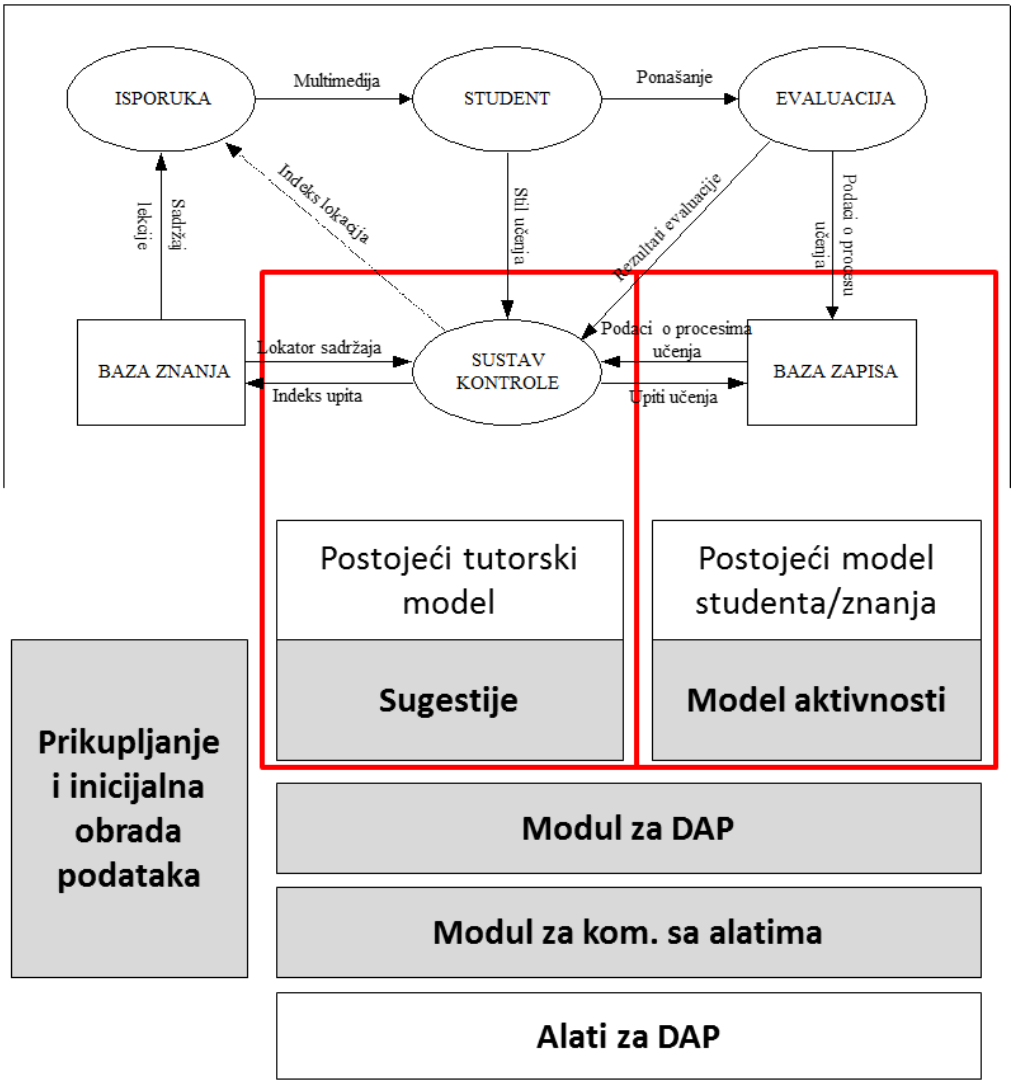
Ovaj model omogućuje da se strategije koje primjenjuju studenti bolje rangiranog klastera sugeriraju studentima lošije rangiranog klastera. Na ovaj način tutorski sustav bi bolje ostvarivao svoju pedagošku funkciju „virtualnog tutora“, predlažući studentima pojmove koje treba učiti na temelju otkrivenih putanja njihovog klastera ili klastera ocjenjenog za jednu razinu bolje (jednom ocjenom više). Time se povećava mogućnost da se sugerirani redoslijed pojmova prilagodi iskustvu, motivaciji i znanju članova svakog klastera, umjesto da sustav sugerira isti redoslijed pojmova svim studentima, iako među njima postoje značajne razlike.

Pregled novog modela arhitekture sustava s ugrađenim prijedlozima proširenja arhitekture za primjenu metoda dubinske analize podataka za poboljšanje prilagodljivosti/učinkovitosti tutorskog sustava prikazan je na slici 8.

Kao temeljna pretpostavka ostvarenja predloženih proširenja arhitekture sustava u cilju poboljšanja njegove prilagodljivosti i učinkovitosti, uočena je potreba za primjenom metoda

dubinske analize podataka. Kao moguće rješenje definirani su sljedeći novi elementi arhitekture sustava:

- modul za prikupljanje i inicijalnu obradu podataka o svim interakcijama korisnika sa sustavom,
- modul za komunikaciju s alatima za dubinsku analizu podataka koji omogućuje pozivanje odgovarajućih algoritama, slanje skupa podataka, temeljnu obradu dobivenih rezultata, itd.,
- modul za provedbu procesa DAP nad prikupljenim podacima.



Slika 8. Proširena arhitektura sustava

Nova arhitektura sustava kreira novi tutorski sustav temeljen na Web-u, koji je u mogućnosti periodično provoditi dubinske analize podataka kroz podsustav za komunikaciju sa

specijaliziranim alatima, čime se ostvaruje nova funkcija prilagodljivosti sustava s ciljem poboljšavanja učinkovitosti sustava, odnosno povećanja kvalitete procesa učenja.

3 Modeliranje sustava

U ovom poglavlju detaljno se opisuje model sustava za povećanje prilagodljivosti tutorskog modela podržanog dubinskom analizom podataka čiji je osnovni koncept opisan u poglavlju 2.5.2. Povećanje prilagodljivosti ostvaruje se proširenjem osnovne arhitekture postojećeg sustava za e-učenje temeljenog na dijalogu koja omogućuje pohranu, pripremu i dubinsku analizu velike količine podataka o interakcijama korisnika sa sustavom. Trenutne funkcionalnosti tutorskog modela sustava omogućuju prilagodbu strukture učenja u ovisnosti od stanja u modelu studenta (savladanosti domene znanja). Prošireni tutorski model sustava, na temelju rezultata provedenih dubinskih analiza podataka, dobiva mogućnost da korisniku sugerira korake u procesu savladavanja domene koji bi trebali poboljšati iskustvo učenja korisnika, kao i povećati učinkovitost samog procesa učenja.

Osnovne komponente predložene nove arhitekture prilagodljivog tutorskog sustava su:

- podsustav za komunikaciju s alatima za dubinsku analizu podataka
- podsustav za klasteriranje i automatsku interpretaciju rezultata klasteriranja
    - sadrži model aktivnosti i učinkovitosti korisnika
- podsustav za otkrivanje čestih i učinkovitih putanja korisnika
- unaprijeđeni tutorski model sustava
    - sadrži podsustav za odabir sugestija

Spomenute komponente ostvaruju osnovni cilj proširenja sustava provedbom procesa koji se sastoji od sljedećih koraka:

1. Periodička provedba DAP - u periodu u kojem korisnici vrše interakciju sa sustavom s ciljem savladavanja odabrane domene znanja, periodički provoditi analizu nad rastućim skupom podataka pri čemu razmak između provedba analize ovisi o intenzitetu aktivnosti korisnika (što veći intenzitet to je razmak manji). Analiza podataka se sastoji od sljedećih faza:
    a. otkriti klastere korisnika koje sustav koriste na sličan način, te ih rangirati vlastitim algoritmom za automatsku procjenu kvalitete procesa učenja svakog klastera
    b. za svaki klaster otkriti skup čestih i učinkovitih putanja savladavanja domene
2. Prezentacija sugestija korisnicima – novi tutorski model na temelju rezultata provedbe DAP korisnicima na početku i na kraju svake aktivnosti prikazuje sugestiju o tome koji pojam je najbolje učiti prije, odnosno nakon pojma kojeg su u tom trenutku odabrali.

Svrha otkrivanja skupina i njihovog automatskog rangiranja prema kvaliteti procesa učenja članova skupine je povećanje prilagodljivosti, odnosno ostvarivanje pedagoške funkcije tutorskog sustava na način da omogućuje provedbu otkrivanja čestih i učinkovitih putanja korisnika za svaki klaster zasebno. Time se omogućuje otkrivanje više različitih putanja za svaki klaster, što nadalje omogućuje tutorskom modelu da jednom korisniku nudi sugestije temeljene na analizi učinkovitih putanja klastera kojem (trenutno) pripada ili klastera čija je kvaliteta procesa učenja procjenjena jednu razinu više, umjesto korištenja jednog skupa sugestija za sve korisnike. Na ovaj način povećava se vjerojatnost da će sugestije biti prilagođene načinu učenja i predznanju korisnika.
U nastavku se opisuje svaka od spomenutih komponenti nove arhitekture sustava.

3.1 Podsustav za komunikaciju s alatima za DAP

Dvije spomenute faze dubinske analize podataka oslanjaju se na algoritam za klasteriranje (eng. *clustering*) korisnika, te algoritam za otkrivanje čestih i učinkovitih uzoraka/sekvenci (eng. *high-utility sequential pattern mining*).

Karakteristika koja razlikuje ovaj sustav od drugih spomenutih u poglavlju 2.3, je da sustav:

a) ne implementira spomenute algoritme u sam kôd sustava - implementacija algoritma u kôd sustava podrazumijeva gubitak fleksibilnosti (u odabiru algoritma za klasteriranje ili otkrivanje čestih uzoraka), donosi rizike ispravnosti implementacije, kao i tehnički izazov implementacije (u skriptni programski jezik kojem nedostaju složene strukture podataka potrebne za implementaciju algoritma).
b) ne zahtijeva nikakvu intervenciju korisnika (administratora, eksperta za dubinsku analizu podataka) u sustav - intervencija korisnika u proces analize unosi veliki vremenski odmak, te rizik ljudske pogreške kod prikupljanja podataka, provedbe analize, te unosa rezultata.

Svrha ovog podsustava je izbjegavanje svih spomenutih nedostataka i rizika. Podsustav omogućava provedbu analize podataka kreiranjem komunikacijskog kanala prema alatima za dubinsku analizu podataka, te daje veliku fleksibilnost omogućavanjem korištenja bilo kojeg algoritma za klasteriranje ili otkrivanje čestih uzoraka koje ti alati nude, uz minimalne intervencije u sustavu.

3.1.1 Alati za dubinsku analizu podataka

Podsustav omogućuje komunikaciju sa skupom od dva alata *AT = {Weka, SPMF}*. Svaki od alata definira i vlastiti format zapisa podataka. Spomenuti alati zadovoljavaju sljedeće kriterije:

- dug period razvoja (>10 godina)
- prepoznati u znanstveno-istraživačkoj zajednici (visok broj citata)
- besplatni za korištenje / otvorenog kôda
- omogućena komunikacija putem komandne linije
- omogućeno pokretanje na poslužitelju (bez sučelja)

Weka (Weka3, 2016) je skup alata za dubinsku analizu podataka, koji implementira veći broj algoritama strojnog učenja i umjetne inteligencije (primarno algoritama klasifikacije i klasteriranja (Pentaho, 2016), te pruža brojne funkcionalnosti pripreme, analize i transformacije ulaznih skupova podataka, provedbe eksperimenata, te vizualizacije rezultata (Witten, et al., 2011).
Karakteristika ovog alata je nužnost oblikovanja ulaznih skupova podataka u vlastiti tekstualni format nazvan *Attribute-Relation File Format* – ARFF (ARFF Data format, 2016), što zahtijeva uvođenje funkcionalnosti za oblikovanje skupa podataka o interakcijama korisnika sa sustavom prema pravilima spomenutog formata podataka.

SPMF (Fournier-Viger, et al., 2014) (*Sequential Pattern Mining Framework*) je alat koji nudi 115 algoritama (primarno algoritama za otkrivanje čestih uzoraka i pravila) te omogućuje vizualizaciju rezultata nekih analiza. Alat ne nudi funkcionalnosti pripreme, analize i transformacije podataka. Karakteristika ovog alata je korištenje vlastitog, jednostavnog tekstualnog formata, što također zahtijeva uvođenje funkcionalnosti za oblikovanje skupa podataka o putanjama korisnika sa sustavom prema pravilima spomenutog formata podataka.

3.1.2 Model podsustava

Podsustav za komunikaciju se sastoji od tri dijela – pripreme API poziva, izvršavanja poziva i obrade rezultata. Kako bi se ostvarila fleksibilnost sustava u pogledu korištenja raznih algoritama potrebno je kreirati model preslikavanja parametara u ispravnu strukturu API poziva. Svaki od alata iz skupa *AT* ima konačan skup algoritama $AK = \{ak_1, ak_2, .., ak_n\}$. Svaki

od tih algoritama ima konačan skup parametara $pr_{akj} = \{pr_1, pr_2, .., pr_m\}$, $m \in \mathbb{N}$, pri čemu se svaki parametar pr_i opisuje uređenim parom *(oznaka, vrijednost)*.
Iako svaki algoritam može imati različiti broj parametara, kod klasteriranja korisnika sustav definira $pr_{min} \subset pr$, koji se sastoji od parametara pr_{min} = {broj_klastera, ulazna_datoteka, izlazna_datoteka} kojim se osigurava postojanje minimalnog skupa nužnih podataka potrebnih za izvođenje algoritma. Ukoliko je treći parametar izostavljen, podrazumijeva se dohvaćanje rezultata sa standardnog izlaza procesa putem kojeg je upućen API poziv. Kod otkrivanja čestih i učinovitih sekvenci sustav definira minimalni podskup parametara pr_{min} = {support, ulazna_datoteka, izlazna_datoteka} pri čemu *support* označava prag (minimalnu frekvenciju pojave nekog uzorka u odnosu na veličinu ulaznog skupa podataka, primjerice za *support*=30%, uzorak se mora pojaviti u barem 30% redaka ulaznog skupa podataka) nakon kojeg se uzorak dodaje u izlazni skup čestih uzoraka. Svi algoritmi za otkrivanje čestih uzoraka primaju ovaj parametar.

Opis ulaznog skupa podataka, skup podataka, te opis izlaznog skupa podataka podsustav dobiva iz podsustava za klasteriranje korisnika (poglavlje 3.2) ili iz podsustava za otkrivanje čestih i učinkovitih sekvenci (poglavlje 3.3).

Na temelju odabranog alata i skupa ulaznih podataka kreira se ulazna datoteka na disku poslužitelja, nakon čega se izvodi API poziv. Svaki API poziv se definira kao uređena n-torka (at_i, ak_j, pr_{min}, pr_{akj}), pri čemu vrijednost at_i određuje alat u kojem se želi izvršiti algoritam, vrijednost ak_j određuje koji se algoritam želi izvršiti, pr_{min} sadrži ranije objašnjeni minimalni skup parametara potrebnih za ispravno pokretanje algoritma, dok pr_{akj} može sadržavati druge vrijednosti iz konačnog skupa ulaznih parametara definiranih za odabrani algoritam (i implementiranih u željenom alatu za DAP – neki alati ne implementiraju sve parametre koje algoritam podržava). Nakon izvršavanja poziva obavlja se inicijalna obrada dobivenih rezultata prema opisu izlaznog skupa podataka. Implementacija podsustava opisana je u poglavlju 4.2.

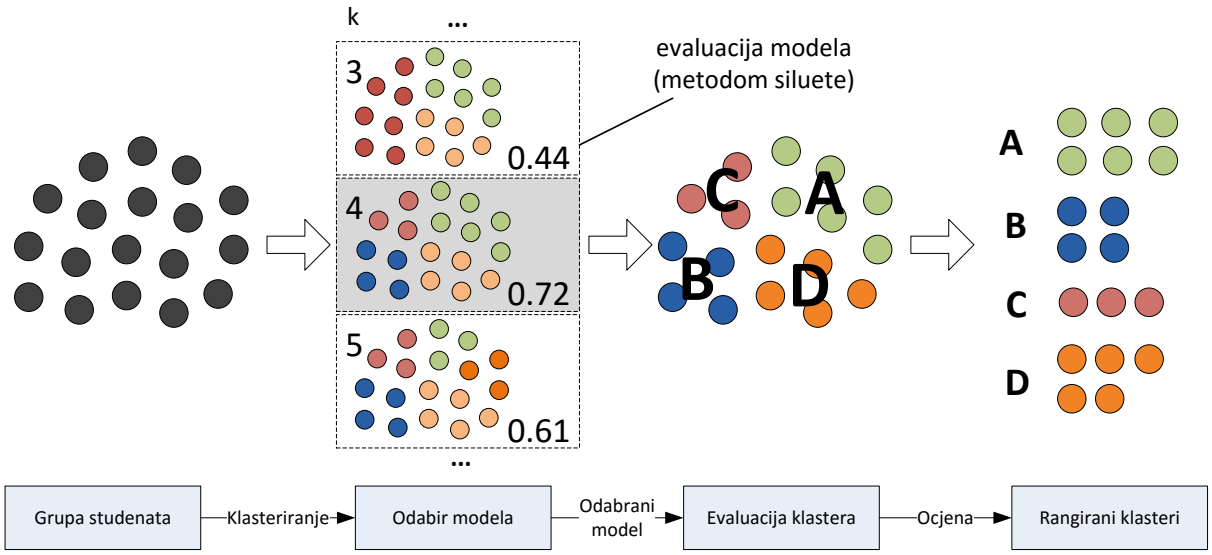
3.2 Podsustav za klasteriranje korisnika

Ovaj podsustav predstavlja najsloženiju komponentu nove arhitekture sustava. Složenost proizlazi iz zahtjeva za potpunom automatizacijom procesa koji se sastoji od više koraka:

a) dohvaćanja i transformacije podataka, te kreiranja značajki odnosno skupa podataka za klasteriranje korisnika (opisano u poglavlju 3.2.1.)
b) kreiranje skupa klastering modela $KM = \{ km_2, km_3, \ldots, km_k\}$ s rastućim brojem klastera $k=\{2,3,...,n\}$ (opisano u poglavlju 3.2.2. i Algoritmu 1)
c) odabir najboljeg klastering modela km_i iz skupa KM (opisano u poglavlju 3.2.3. i Algoritmu 2)
d) evaluacije procesa učenja svih klastera u odabranom modelu km_i i rangiranje prema rezultatima evaluacije (opisano u poglavlju 3.2.4. i Algoritmu 3)

Vizualizacija navedenih koraka dana je na slici 9.

Rezultat ovog podsustava je fleksibilnost sustava pri odabiru sugestija koje imaju za cilj usmjeravanje korisnika kako bi njegov proces učenja učinile učinkovitijim.



Slika 9. Funkcija podsustava za klasteriranje korisnika

Prema preporukama o granularnosti sustava za pohranu podataka (Bouchet, 2013; Mostow, 2006), osmišljeni su mehanizmi za pohranu podataka u ovom sustavu, a njihova implementacija u bazi podataka prikazana je u poglavlju 4.1. Podaci o interakcijama u svom izvornom obliku ne mogu se direktno koristiti za klasteriranje korisnika, već se na temelju tih podataka kreiraju nove značajke koje na kvalitetniji način modeliraju način rada korisnika i njegovu učinkovitost (brzinu napredovanja kroz domenu znanja). Postupci za kreiranje spomenutih značajki, vrijednosti kojih predstavljaju ulazni skup podataka za algoritam za klasteriranje, opisani su u sljedećem poglavlju.

3.2.1 Kreiranje značajki za provedbu klasteriranja korisnika

Kako se dubinska analiza podataka u sustavu provodi periodički, dok korisnici savladavaju domenu znanja, proces pripreme značajki mora biti dinamičan, jer se korisnici u trenutku pokretanja analize razlikuju prema stanju u modelu studenta, odnosno u postotku savladanosti domene znanja. Osnovni cilj procesa kreiranja značajki za provedbu klasteriranja korisnika je osigurati jednake vrijednosti kreiranih značajki svim korisnicima koji koriste sustav na jednak način, neovisno o stupnju savladanosti domene. Primjerice, dva korisnika sustava mogu imati jednak broj aktivnosti učenja (npr. 100), ali je prvi tijekom tih aktivnosti savladao 10 pojmova u domeni znanja, dok je drugi savladao 5 pojmova. Prvi korisnik ima dvostruko bolju učinkovitost, što jedna od značajki koja čini model korisnika u ovom sustavu. Nadalje, neka druga dva korisnika sustava mogu, u trenutku pokretanja DAP, imati 10 odnosno 100 izvršenih aktivnosti UČENJA, pri čemu je prvi savladao 1 pojam, dok je drugi savladao 10, što ih s aspekta načina rada čini jednakima. Prema osnovnim funkcionalnostima sustava razvijene su dvije značajke: za aktivnosti Učenja i aktivnosti Ponavljanja. Nadalje, dva korisnika mogu jedan pojam u domeni savladati u 5 odnosno 50 minuta, pa je razvijena značajka koja predstavlja korisnika s aspekta uloženog vremena, odnosno vremena korištenja sustava. Konačan model korisnika čine četiri značajke o: aktivnostima učenja, aktivnostima ponavljanja, uloženom vremenu i učinkovitosti (brzini napredovanja).

Prije definiranja postupaka kreiranja spomenutih značajki potrebno je definirati osnovne skupove podataka o interakcijama korisnika sa sustavom koji se pohranjuju u bazu podataka i koriste postupku kreiranja.

Jedna domena znanja sastoji se od konačnog skupa unesenih pojmova (eng. *knowledge unit*) $KU = \{ku_1, ku_2, .., ku_n\}$, $n \in \mathbb{N}$, pa je ukupan broj pojmova $KU_{total} = |KU|$. Za svaki pojam ku_i definira se raspon vrijednosti $[s_{ku_i}, h_{ku_i}] \mid s_{ku_i}, h_{ku_i} \in \mathbb{R})$ pri čemu s_{ku_i} predstavlja početnu razinu znanja pojma, dok h_{ku_i} predstavlja konačnu razinu („prag“) koji korisnik mora dosegnuti točnim odgovaranjem na postavljena pitanja. Skup korisnika $U = \{u_1, u_2, ... , u_m\}$, $m \in \mathbb{N}$ savladava domenu znanja pri čemu se za svakog korisnika u_i kreira prekriveni (eng. *overlay*) model (znanja) studenta odnosno vektor $\overrightarrow{v_{u_i}}$ koji sadrži trenutnu razinu savladanosti pojma unutar definiranog raspona vrijednosti $s_{ku_i} < v_{ku_i} < h_{ku_i}$ svakog pojma u domeni znanja $\overrightarrow{v_{u_i}} = (v_{ku_1}, .., v_{ku_n}) \mid v_{ku_l} \in \mathbb{R}, l \in [1, n]$. Skup pojmova u domeni znanja koje je

korisnik započeo savladavati naziva se $A_{u_i} \subseteq KU \mid s_{ku_i} < v_{ku_i} < h_{ku_i}; v_{ku_i} \in \overrightarrow{v_{u_i}}$. Brojnost ovog skupa koristi se pri izračunu vrijednosti značajki koje predstavljaju ulazni skup podataka u algoritam za klasteriranje studenata. Skup pojmova u domeni znanja koje je korisnik savladao u trenutku provedbe DAP čine skup $\mathrm{C_{u_i}} \subseteq \mathrm{KU} \mid v_{ku_i} = h_{ku_i}$; $v_{ku_i} \in \overrightarrow{v_{u_i}}$.

Za svaki pojam ku_i, nastavnik unosi skup pitanja UQ = {uq_1,uq_2,...,uq_n}, n ∈ ℕ. Za svako pitanje definira se:

- ku_i – na koji pojam se pitanje odnosi (ku_i ∈ KU)
- *wi* – (eng. *weight increase*) iznos promjene trenutne vrijednosti v_{ku_i} u modelu studenta $\overrightarrow{v_{u_i}}$ u slučaju točnog odgovora
- *wd* – (eng. *weight decrease*) iznos promjene trenutne vrijednosti v_{ku_i} u modelu studenta $\overrightarrow{v_{u_i}}$ u slučaju netočnog odgovora
- *tip* ∈ {1,2,3} gdje vrijednosti skupa označavaju: 1 = pitanje za provjeru znanja, 2 = inicijalno pitanje i 3 = ispitno pitanje.

Pri interakciji sa sustavom korisnik u_i se služi dvama osnovnim funkcionalnostima: učenja (eng. *Learn*) (L) i ponavljanja (eng. *Repetition*) (R). Svaka korisnikova aktivnost učenja $\mathrm{L_{u_i}} = \{\mathrm{L_1}, \mathrm{L_2}, .., \mathrm{L_n}\}, \mathrm{n} \in \mathbb{N}$, odnosno ponavljanja $\mathrm{R_{u_i}} = \{\mathrm{R}, \mathrm{R_2}, .., \mathrm{R_m}\}, \mathrm{m} \in \mathbb{N}$ opisuje se četvorkom {ku_j, *ts*, *te*, QP_{KUi}} pri čemu je:

- ku_j – odabrani pojam za učenje ku_j ∈ KU
- *ts* – (eng. *time started*) vrijeme početka učenja
- *te* – (eng. *time ended*) vrijeme završetka učenja
- QP_{KUi} – skup postavljenih pitanja kao uređenih parova (q ∈ UQ, *točan* ∈ {0,1}), pri čemu *točan* čuva informaciju o točnosti ponuđenog odgovora na prikazano pitanje.

Prilikom pokretanja DAP izračunava se i broj svih aktivnosti korisnika u_i na savladanim pojmovima $\mathrm{lc_{u_i}} = |\mathrm{LC_{u_i}}|$ gdje je $\mathrm{LC_{u_i}} \subseteq \mathrm{L_{u_i}} | ku_j \in \mathrm{C_{u_i}}$, te određuje skup aktivnosti na nesavladanim pojmovima $\mathrm{LA_{u_i}}$ gdje je $\mathrm{LA_{u_i}} \subseteq \mathrm{L_{u_i}} | ku_j \in \mathrm{A_{u_i}}, ku_j \notin \mathrm{C_{u_i}}$.

U trenutku izvođenja analize neki korisnik u_i, s aspekta njegove interakcije sa sustavom, može biti u tri stanja:

a) neaktivan – nema niti jedan učeni niti savladani pojam $A_{u_i} = \{\emptyset\}, \mathrm{C_{u_i}} = \{\emptyset\}$
b) ima učenih i/ili savladanih pojmova $\mathrm{A_{u_i}} \neq \{\emptyset\}, \mathrm{C_{u_i}} \neq \{\emptyset\}$
c) savladao je sve pojmove u domeni $\mathrm{A_{u_i}} = \mathrm{KU}, \mathrm{C_{u_i}} = \mathrm{KU}$.

Sa aspekta savladanosti pojmova u domeni, svaki element vektora $\overrightarrow{v_{u_i}}$ može odražavati stanje:

a) „savladan“ gdje vrijedi $v_{ku_i} = h_{ku_i}$

b) „nije savladan“ gdje vrijedi $v_{ku_i} < h_{ku_i}$

Sa aspekta učinkovitosti, način savladavanja nekog pojma iz domene znanja može imati tri stanja:

a) savladan (*„completed”* stanje)
b) nije savladan, ali nije dosegnut minimalan broj pitanja potrebnih za savladavanje pojma $\boldsymbol{qmin_{ku_i}}$ (*„started”* stanje), odnosno korisnik je započeo s učenjem pojma, ali mu je broj postavljenih pitanja $\boldsymbol{qpost_{ku_i}}$ nego što je potrebno da bi se pojam savladao
c) nije savladan, ali je broj postavljenih pitanja veći od minimalnog broja pitanja potrebnih za savladavanje pojma (*„struggling”* stanje), odnosno korisnik nije na sva postavljena pitanja odgovarao točno (broj točnih odgovara označava se s $\boldsymbol{qcorr_{ku_i}}$).

Minimalan broj pitanja $\boldsymbol{qmin_{ku_i}}$ je broj pitanja potrebnih za svladavanje pojma ako se konstatno daju točni odgovori na postavljena pitanja. Ova stanja govore o dinamici procesa učenja, koju je potrebno predstaviti skupom značajki koje modeliraju korisnika s dva važna aspekta – načina/navike korištenja sustava i učinkovitosti učenja.

Nakon definiranja osnovnih skupova podataka o interakcijama korisnika sa sustavom, definiraju se značajke koje čine model korisnika u_i:

1. broj aktivnosti učenja (L_{final}),
2. broj aktivnosti ponavljanja (R_{final}),
3. uloženo vrijeme učenja (T_{final}),
4. učinkovitost E_{final}.

Značajka L_{final} predstavlja način korištenja funkcionalnosti Učenja. Cilj niže navedenih izraza je da značajka poprimi iste vrijednosti za one korisnike koji koriste funkcionalnost Učenja na jednak način neovisno o broju savladanih pojmova.

$$L_{final} = \begin{cases} \dfrac{\sum_{i=1}^{|A_{u_i}|} |L_{u_i}| * \dfrac{h_{ku_i}}{v_{ku_i}}}{|A_{u_i}|} & ako\ je\ \ C_{u_i} = 0, \\ \left(\dfrac{lc_{u_i}}{|C_{u_i}|} + \dfrac{\sum_{i=1}^{|LA_{u_i}|} la_{u_i} * \dfrac{h_{ku_i}}{v_{ku_i}}}{|A_{u_i} - C_{u_i}|}\right) \Big/ 2 & ako\ je\ 0 < C_{u_i} < A_{u_i}, \\ \dfrac{lc_{u_i}}{|C_{u_i}|} & ako\ je\ C_{u_i} = A_{u_i}\ , \\ 0\ ako\ je\ \ |A_{u_i}| = \{\emptyset\} \end{cases}$$

Pri izračunu vrijednosti značajke L_{final} za pojedinog korisnika koriste se četiri izraza ovisno o brojnosti njegovih interakcija sa sustavom i stanju u modelu studenta u trenutku pokretanja DAP. Prvi izraz koristi se u slučaju da korisnik ima aktivnosti učenja, ali nije savladao niti jedan pojam (npr. ima 100 aktivnosti učenja na 10 pojmova, ali niti jedan pojam još nije savladao). Drugi izraz koristi se u slučaju da korisnik ima aktivnosti učenja na više pojmova, pri čemu je dio savladao, a dio nije. Ukoliko je korisnik savladao sve učene pojmove (ili sve pojmove u domeni) tada se primjenjuje treći izraz. Konačno, ukoliko je korisnik u trenutku provedbe DAP bio neaktivan, odnosno nije niti započeo s korištenjem sustava vrijednost značajke je 0.

Značajka R_{final} prikazuje način korištenja funkcionalnosti Ponavljanja. Cilj niže navedenog izraza je da značajka poprimi iste vrijednosti za one korisnike koji koriste funkcionalnost Ponavljanja na jednak način neovisno o broju savladanih pojmova.

$$R_{final} = \begin{cases} \dfrac{\sum_{i=1}^{|A_{u_i}|} |R_{u_i}| * \dfrac{h_{ku_i}}{v_{ku_i}}}{|A_{u_i}|} & ako \;\; A_{u_i} \neq \{\emptyset\}, \\ 0 \; ako \; je \;\; |A_{u_i}| = \{\emptyset\} \end{cases}$$

Značajka T_{final} prikazuje odnos uloženog vremena (korištenja sustava) i broja savladanih i učenih pojmova tijekom svih aktivnosti učenja ili ponavljanja. Prije izračuna vrijednosti značajke T_{final} izračunavaju se sljedeće pomoćne vrijednosti:

- $LR_{u_i} = L_{u_i} \cup R_{u_i}$
- t_{ku_i} – ukupno vrijeme učenja i/ili ponavljanja nekog pojma ku_i

  $t_{ku_i} = \sum(te\text{-}ts) \;\; \forall LR_{u_i} \mid ku_i \in A_{u_i}$
- tt_{u_i} – ukupno vrijeme učenja svih učenih ili ponavljanih pojmova korisnika u_i
  - $tt_{u_i} = \sum_{i=1}^{|LR_{u_i}|}(te - ts)$

$$T_{final} = \begin{cases} \dfrac{tt_{u_i}}{A_{u_i}} & ako \; je \; C_{u_i} = 0 \;\; i \; A_{u_i} > 0, \\ \dfrac{\sum t_{ku_i} * \dfrac{h_{ku_i}}{v_{ku_i}}}{C_{u_i}}, \; \forall ku_i \in A_{u_i} \; ako \; je \; 0 < C_{u_i} < A_{u_i}, \\ \dfrac{tt_{u_i}}{C_{u_i}} & ako \; je \; C_{u_i} = A_{u_i} \end{cases}$$

Nakon kreiranja vrijednosti značajki L_{final}, R_{final} i T_{final} za sve korisnike iz skupa U, izračunava se prosječna vrijednost μ svake od značajki, te njihova standardna devijacija σ. Nakon toga sve tri značajke svakog korisnika iz skupa U se standardizira. Standardizacija je uobičajeni statistički postupak kojom se osigurava da će vrijednosti u skupu poprimiti vrijednosti koje su manje ili veće od prosječne vrijednosti skupa. Time se dobiva distribucija koja je nužna za provedbu sljedećih aktivnosti u postupku klasteriranja. Vrijednosti se standardiziraju općenitim izrazom:

$$z = \frac{x - \mu}{\sigma}$$

gdje je z standardizirana vrijednost neke vrijednosti x, umanjene za prosječnu vrijednost μ skupa vrijednosti kojem pripada x, podjeljene sa standardnom devijacijom σ tog skupa vrijednosti.

Standardizacijom značajki L_{final}, R_{final} i T_{final} dobiva se novi skup značajki koje se označavaju s L_{std}, R_{std} i T_{std}.

Njima se pridružuje i značajka učinkovitosti učenja. Postupak izračuna te značajke opisan je u sljedećim odlomcima.

Dok ranije spomenute značajke govore *kako* korisnik uči, značajka učinkovitosti E_{final} modelira učinkovitost korisnika, odnosno govori *koliko uspješno (brzo)* korisnik napreduje kroz domenu znanja.

Značajka učinkovitosti nekog korisnika u_i iz skupa U sastoji se od tri dijela:

- učinkovitost na savladanim pojmovima

$$E_1 = \begin{cases} \sum \frac{1}{qpost_{ku_i}/qmin_{ku_i}} * \frac{v_{ku_i}}{h_{ku_i}}, \\ 1, \quad ako\ je\ (qpost_{ku_i} < qmin_{ku_i}) \\ \qquad i\ (v_{ku_i} = h_{ku_i}) \end{cases}$$

- učinkovitost na nesavladanim pojmovima gdje je $qpost_{ku_i} < qmin_{ku_i}$

$$E_2 = \begin{cases} \sum \frac{qcorr_{ku_i}}{qpost_{ku_i}} * \frac{v_{ku_i}}{h_{ku_i}}, \\ 1, \quad ako\ je\ \frac{qcorr_{ku_i}}{qpost_{ku_i}} = 1 \end{cases}$$

- učinkovitost na nesavladanim pojmovima gdje je $qpost_{ku_i} > qmin_{ku_i}$

$$\mathrm{E_3} = \sum \frac{qcorr_{ku_i}}{qpost_{ku_i}} * \frac{v_{ku_i}}{h_{ku_i}} * \frac{1}{qpost_{ku_i}/qmin_{ku_i}}$$

Konačna značajka učinkovitosti dobiva se izrazom:

$$\mathrm{E_{final}} = \frac{\mathrm{E_1} + \mathrm{E_2} + \mathrm{E_3}}{\mathrm{A_{u_i}}}$$

odnosno sumom svih učinkovitosti podjeljene s brojem učenih pojmova.
Ova značajka se ne standardizira, već viša vrijednost predstavlja bolju učinkovitost korisnika.

Konačno, model jednog korisnika koji se označava s DS_{u_i}definira se kao četvorka (L_{std}, R_{std}, T_{std}, E_{final}) dok se konačni ulazni skup podataka SP za sljedeći korak DAP (klasteriranje), dobiva kreiranjem modela za sve korisnike u skupu U, odnosno SP = {DS_1, DS_2,....DS_n} | n ∈ U.

3.2.2 Izrada skupa modela klasteriranja

Kao što je prikazano na slici 9, klasteriranje korisnika započinje izradom skupa modela. Skup modela potrebno je izraditi iz razloga što se broj željenih klastera mora unaprijed zadati algoritmu za klasteriranje kMeans (Lloyd., 1982), a nije moguće unaprijed znati koji će model klasteriranja izrađen nad skupom podataka *SP* biti najkvalitetniji. Algoritam 1. određuje početni maksimalni k (broj klastera), na temelju izraza:

$$\mathrm{k_{max}} = \mathrm{ceil}(\sqrt{\frac{|\mathrm{SP}|}{2}})$$

Kako je broj studenata u grupi tijekom ovog istraživanja bio uvijek jednak i relativno malen (~30) vrijednost *k* će prema ovom izrazu uvijek biti jednaka 4. Kako bi se algoritam za kreiranje skupa modela klasteriranja učinio fleksibilnijim najveći broj klastera za koji će se pokrenuti algoritam kMeans uvećava se za 5. Konačna vrijednost petlje vjerojatno neće biti dosegnuta zbog uvjeta kojim se prekida algoritam izrade modela. U trenutku kada Algoritam 1 dobije iz *kMeans* algoritma model koji ima dva klastera sa samo jednim studentom (jedan

klaster s jednim studentom je dozvoljen kako bi se omogućilo izdvajanje studenata s iznimnim vrijednostima, eng. *outlier*) algoritam se završava.

Algoritam 1: Kreiranje skupa modela klasteriranja

Ulaz: SP, at_i, ak_j, pr_{min}, pr_{akj}

Izlaz: Skup modela s rastućim brojem skupina *modeli[k]=[centroidi, model]*

```
Kreiraj naziv datoteke
Zapiši datoteku na disk
kmax = odaberi inicijalni maksimalni broj klastera
Definiraj polje: modeli
za i = 2 do kmax + 5
    x = kreiraj API poziv(ati, akj, prmin, prakj) // Weka, kMeans, i, SP,…
    rez = izvrši API poziv(x)
    izlaz = obradi(rez)  // [centroidi, model]
    ako je (broj klastera s jednim članom) >= 2)
      prekid
    modeli[i]=rez
vrati modeli
```

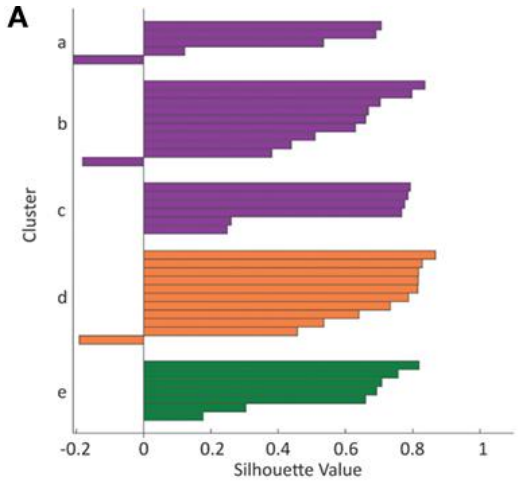
Rezultat svakog pokretanja algoritma kMeans (rez) čine: broj klastera, središnje vrijednosti (centroidi) svih značajki svakog klastera te informacija kojem klasteru pripada svaki student (redak) u ulaznom skupu podataka - model. Algoritam 1 koristi podsustav opisan u podpoglavlju 3.1. (u dvama osjenčanim linijama) pri čemu se kroz taj podsustav izvršava komunikacija s alatima za DAP, odnosno poziva algoritam u željenom alatu, šalje ulazni skup podataka koji se sastoji od ranije opisanih značajki te dohvaćaju rezultati i vraćaju u sustav). Nakon kreiranja skupa modela klasteriranja moguće je pokrenuti postupak procjene kvalitete modela klasteriranja, te na temelju te procjene odabrati najbolji model.

3.2.3 Postupak procjene kvalitete i odabira modela

Postoje brojne metode za procjenu kvalitete distribucije modela odnosno broja klastera u modelu (Symons, 1981; Tibshirani, 2001). Neke od tih metoda zahtijevaju izmjenu osnovnog kôda algoritma za klasteriranje, dok se druge mogu primjeniti na gotovi model. Procjena kvalitete modela metodom siluete (eng. *silhouette statistic*) pripada u drugu skupinu. Ideja

primjene silueta prvi puta je predstavljena u (Rousseeuw, 1987), kao način vizualizacije kvalitete smještaja pojedine instance (npr. korisnika) u određeni klaster. Na slici 10 prikazan je model sa 5 klastera (na y osi), te vrijednost siluete (pouzdanosti da je član smješten u odgovarajući klaster) za svakog člana klastera.

U sljedećem odlomku opisan je postupak izračuna vrijednosti siluete za cjelokupni model klasteriranja, odnosno izračun kvalitete tog modela za zadani broj klastera k.



Slika 10. Primjer vizualizacije kvalitete modela klasteiranja metodom siluete (McGarry, i dr., 2010)

Kako bi se kreirala silueta za neki klaster *m* potrebno je, za svaku vrijednost *a(i)* u modelu podataka (npr. *SP*) s brojem klastera *C* izračunati:

a(i) - euklidsku udaljenost objekta *i* od svih drugih instanci u klasteru

d(i;C) - euklidsku udaljenost objekta *i* od svih instanci u svim drugim klasterima u modelu

avg(d(i;C)- prosječnu udaljenost objekta *i* od svih instanci u svim drugim klasterima u modelu

b(i) = min avg(d(i;C) – odabrati klaster za najmanjim prosjekom udaljenosti između instanci

U konačnici, izračunava se vrijednost siluete za klaster prema pravilima:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{ako je } a(i) < b(i) \\ 0, & \text{ako je } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{ako je } a(i) > b(i) \end{cases}$$

Vrijednost siluete može poprimiti vrijednosti između -1 i 1, pri čemu je 1 idealna vrijednost. Kako bi dobili vrijednost siluete za cijeli model s(g) izračunava se prosječna vrijednost s(i) za sve instance u skupu podataka. Na taj način s(g) odražava prosječnu širinu siluete za cijeli skup podataka, odnosno kompaktnost instanci unutar klastera i međusobnu udaljenost klastera u modelu. Prosječna vrijednost s(g) za cijeli skup podataka tako postaje mjera kvalitete

modela. Algoritam 2 dobiva skup modela klasteriranja dobivenih iz Algoritma 1, te vraća model s najvišom vrijednošću siluete.

Algoritam 2: Procjena kvalitete modela klasteriranja

Ulaz: *modeli* – skup modela klasteriranja

Izlaz: *model* – model iz skupa modeli s najvišom vrijednošću siluete

Definiraj polje: rezultati
za svaki *model* u *modeli*
izradi matricu m*m gdje je m = broj instanci (studenata) u modelu
izračunaj udaljenost svake instance u odnosu na sve instance pripadajućeg klastera
izračunaj udaljenost svake instance u odnosu na sve instance svih ostali klastera
izračunaj prosječnu udaljenost za sve instance u pripadajućem klasteru $(a_1 \ldots a_n) \mid n < m$
izračunaj prosječnu udaljenost za sve instance u svim ostalim klasterima $(b_o \ldots . b_p) \mid o > n, p <= m$
kreiraj matricu (*2xn*) $a_1 \ldots . a_n$, *min(b1,.., bn)*
izračunaj s*(i)* za sve instance
izračunaj *avg s(i)* za sve klastere u modelu *1,…,k*
izračunaj *S(g)* za model
rezultati[k]=S(g)
vrati *max*(rezultati[k])

Nakon odabira najkvalitetnijeg modela, primjenjuje se postupak procjene kvalitete procesa učenja članova svakog klastera u odabranom modelu.

3.2.4 Postupak procjene kvalitete procesa učenja odabranog modela

Kako se DAP provodi na rastućem skupu podataka bilo je potrebno kreirati fleksibilnu metodu za automatsku procjenu kvalitete procesa učenja svakog klastera u odabranom modelu. Metoda se temelji na statističkoj analizi vrijednosti standardiziranih značajki skupa podataka L_{std}, R_{std}, T_{std}, i primjeni sustava bodovanja.

S obzirom da se standardizirane vrijednosti spomenutih značajki izračunavaju na temelju prikupljenih podataka o interakcijama korisnika sa sustavom, uočeno je da distribucija dobivenih vrijednosti odgovara normalnoj raspodjeli. Vrijednosti skupa standardiziranih

značajki mogu se vizualizirati izračunom vrijednosti funkcije razdiobe vjerojatnosti normalne distribucije (eng. *probability density function – pdf*) (Petz, 2004).

$$\text{pdf(x)} = \frac{1}{\sigma\sqrt{2\,\pi}}\, e^{-(x-\mu)^2\,/\,(2\sigma^2)}$$

Radi evaluacije procesa učenja u sustavu opisan je idealni korisnik kao osoba s **ispodprosječnim brojem aktivnosti učenja** (značajka L_{std}) i **ponavljanja** (značajka R_{std}), odnosno učinkovitija u procesu učenja u odnosu na druge korisnike (potrebno joj je manje aktivnosti za isti pomak u savladanosti domene znanja), **izadprosječnim vremenom učenja** (značajka T_{std}) odnosno boljom koncentracijom na analizu nastavnih materijala, umjesto pogađanja odgovora na pitanja) **te što je moguće višom učinkovitošću** (značajka E_{final}).

Koraci metode prikazani su na slici 11. Kako bi se procjenila kvaliteta procesa učenja svakog klastera, uzimaju se prosječne vrijednosti značajki svih članova klastera (centroidi), te određuje njihova pozicija u skupu vrijednosti funkcije *pdf()* za svaku značajku. Ukupan raspon vrijednosti funkcije (u standardnim devijacijama) dijeli se na 13 jednakih intervala, pri čemu je unaprijed svakom intervalu i svakoj od značajki koje čine skup podataka dodjeljen broj „bodova“ od 1 do 13 (Tablica 1). Razlog podjele raspona vrijednosti na 13 intervala dolazi od normalne distribucije, kod koje vrijedi pravilo da se unutar 3 standardne devijacije (SD) nalazi 99,7% vrijednosti skupa podataka. Dakle, s lijeve i desne strane prosječne vrijednosti normalne distribucije nalaze se 3 SD (6 intervala). Radi povećanja osjetljivosti algoritma procjene kvalitete procesa učenja, svaka SD podjeljena je na pola, što daje ukupno 12 intervala. Dodan je i središnji interval koji sadržava prosječne vrijednosti čime je dobiveno ukupno 13 intervala.

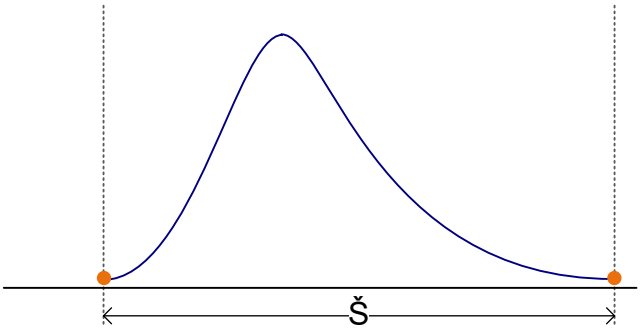
Tablica 1. Mapiranje broja bodova na intervale

INTERVAL	1	2	3	4	5	6	7	8	9	10	11	12	13
L_{std}	1	2	3	4	13	12	11	11	9	8	7	6	5
R_{std}	1	2	3	4	13	12	11	10	9	8	7	6	5
T_{std}	1	2	3	4	9	10	11	12	13	8	7	6	5
E_{final}	izračun [1…13]												

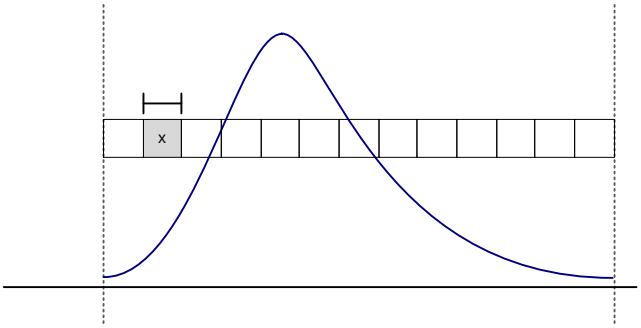
Iz Tablice 1. vidljivo je da se, za značajke L_{std} (Učenje) i R_{std} (Ponavljanje), najveći broj bodova dodjeljuje klasteru(ima) koji se nalaze u intervalima lijevo od prosječne vrijednosti, a više se bodova dodjeljuje „sporijim“ klasterima (sasvim desno u distribuciji) od onih „neaktivnih“ (sasvim lijevo u distribuciji). Kod bodovanja značajke uloženog vremena T_{std}, prema ranije navedenom modelu idealnog korisnika sustava, najviše bodova dodjeljuje se

onom klasteru(ima) koji pokazuju iznadprosječne vrijednosti, nešto manje „sporijim“ klasterima, a najmanje „neaktivnim“.

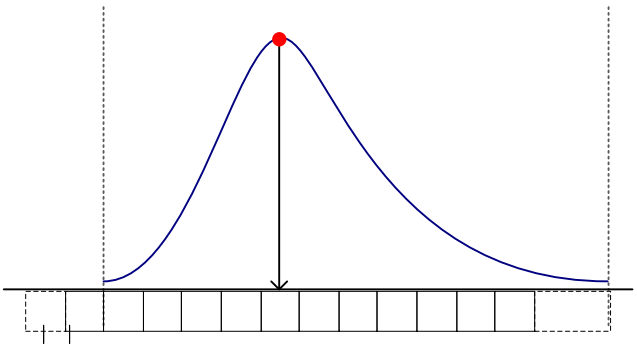
A) Odrediti raspon vrijednosti funkcije P(x) i rubne vrijednosti



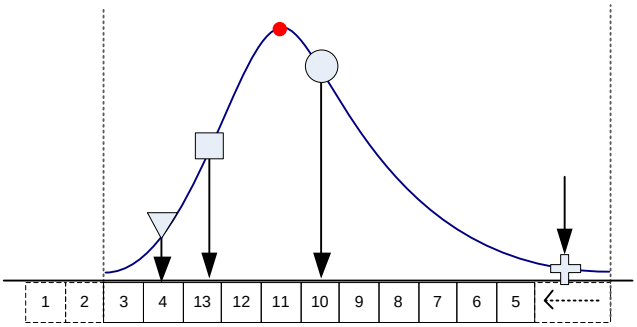
B) Podijeliti raspon na 13 jednakih intervala



C) Pronaći max(pdf(x)), D) Pozicionirati skup intervala



E) Postaviti centroide klastera, F) Odrediti interval



G) Sumirati broj bodova za sve standardizirane značajke

Slika 11. Koraci procjene kvalitete procesa učenja

Radi povećanja osjetljivosti algoritma procjene moguće je 3 osnovne SD podijeliti na 3,4 ili više dijelova. U tom slučaju potrebno je izraditi novu tablicu mapiranja bodova na intervale (Tablica 1). Važno je istaknuti da navedena tablica vrijedi za sve domene znanja u prilagodljivom tutorskom sustavu. Metoda je implementirana u Algoritmu 3.

Algoritam 3: Procjena kvalitete procesa učenja klastera u odabranom modelu

Ulaz: model[k], centroidi[k]

$DS = \{L_{std}, R_{std}, T_{std}, E_{final}\}$

M(f) - tablica mapiranja broja bodova na raspone vrijednosti funkcije P(x)

Izlaz: rangirana lista klastera prema procjeni kvalitete procesa učenja

```
Definiraj polja: poredakKlastera, rezultatiCentroida
za svaku značajku f u DS
   ako je f == Efinal
      rez = izračunajBodove(centrodi(f));
      rezultatiCentroida [f][k]=rez
   inače
       ndf = pdf(f)
       cw = raspon(pdf(f))
       iw = širina_jednog_intervala(cw)
      peak = max(ndf)
      M[f][centar] = postavi sredinu raspona(peak)
      M[f][rubovi]= pronađi rubove svakog intervala(peak, iw, cw, SP(f))
       za svaki centroid ∈ centroidi[f]
          i = pronađi interval (centroid)
          score = dohvati broj bodova(i,M(f))
          rezultatiCentroida [f][k]=score
za svaki rc u rezultatiCentroida
    poredakKlastera[k] = sum(rc[k])
rsort(poredakKlastera) // silazno sortiranje
vrati poredakKlastera
```

Nakon izračuna vrijednosti *pdf(x)*, pronalazi se najveća vrijednost funkcije. Nakon toga se skup od 13 intervala postavlja na pronađenu maksimalnu vrijednost funkcije, što može uzrokovati „izlazak“ pojedinih intervala izvan raspona vrijednosti krivulje ili ne-dosizanje

rubnih dijelova krivulje (Slika 11, korak D). U tom slučaju koristi se aproksimacija vrijednosti.

Dodjela bodova prema vrijednosti centroida značajke učinkovitosti E_{final} provodi se u nekoliko koraka:

1. odabirom centroida s maksimalnom vrijednošću značajke učinkovitosti,
2. dodjelom maksimalnog broja bodova (13) tom centroidu,
3. dodjelom proporcionalnog broja bodova ostalim centroidima *x* u odnosu na maksimalnu vrijednosti centroida prema izrazu:

$$\text{bodova} = \left(\frac{\text{x}}{\max(\text{E}_{\text{final}})}\right) * 13$$

Primjer pozicioniranja vrijednosti centroida klastera, za model s pet klastera, na tablicu s brojem bodova dan je u Tablici 2.

Tablica 2. Primjer dodjele broja bodova klasterima prema pripadnosti intervalu

							Mode							
L rubovi	-2.30	-1.99	-1.67	-1.35	-1.03	-0.72	-0.39	-0.08	0.24	0.55	0.87	1.19	1.51	1.82
L bodovi	1	2	3	4	13	12	11		10	9	8	7	6	5
Pozicije		5	1						4	3				2
R rubovi	-2.45	-2.07	-1.69	-1.32	-0.95	-0.57	-0.19	0.17	0.55	0.92	1.30	1.68	2.05	2.43
R bodovi	1	2	3	4	13	12	11		10	9	8	7	6	5
Pozicije		5				1,3				2,4				
T rubovi	-1.83	-1.55	-1.27	-0.99	-0.71	-0.43	-0.15	0.13	0.41	0.69	0.97	1.25	1.53	1.81
T bodovi	1	2	3	4	9	10	11		12	13	8	7	6	5
Pozicije				1,5		4				2		3		

Kao što je vidljivo iz primjera, više klastera može se naći u istom intervalu, te za jednu značajku dobiti isti broj bodova.

Tablica 3. Konačno bodovanje procesa učenja po klasterima

SP	k=5	Vrijednosti centroida klastera / bodovi				
	Klasteri	1	2	3	4	5
L_{std}	Centroid	-1.388	2.145	0.324	0.104	-1.850
	Bodovi	3	5	9	10	2
R_{std}	Centroid	-0.279	0.790	-0.249	0.821	-2.130
	Bodovi	12	9	12	9	2
T_{std}	Centroid	-0.786	0.520	1.129	-0.275	-0.970
	Bodovi	4	13	7	10	4
E_{final}	Centroid	0.138	0.235	0.489	0.203	0.005
	Bodovi	4	6	13	5	0
Ukupno		**23**	**33**	**41**	**34**	**8**
Ocjena		**D**	**C**	**A**	**B**	**E**

Nakon provedbe bodovanja posljednje značajke, učinkovitosti učenja, dobiva se konačan broj bodova, odnosno ocjena svakog klastera u modelu (prikazano u Tablici 3).

Time se završava proces evaluacije odabranog modela klasteriranja. Nakon provedbe postupka procjene kvalitete procesa učenja po klasterima, moguće je provesti dubinsku analizu podataka nad putanjama kroz domenu znanja koje su tijekom interakcije sa sustavom kreirali članovi svakog klastera zasebno.

3.3 Podsustav za otkrivanje čestih i učinkovitih putanja korisnika

Prije korištenja bilo koje funkcionalnosti sustava korisnik se mora prijaviti u sustav čime kreira novu sesiju tijekom koje može kreirati određen broj interakcija sa sustavom. Preduvjet pokretanja svakog procesa učenja ili ponavljanja je odabir pojma za učenje/ponavljanje. U slučaju kreiranja aktivnosti Učenja, nakon odabira pojma, moguće je učenje niza povezanih pojmova, u slučaju netočnih odgovora na inicijalna pitanja, kao što je opisano u poglavlju 2.3. U slučaju kreiranja aktivnosti Ponavljanja, ovisno o povezanosti netočnog odgovora s drugim pojmom, također je moguće učenje niza povezanih pojmova. Niz zapisa o aktivnosti učenja postaje sekvenca $P = \langle L_i, L_j, \ldots, L_x \rangle \mid i, j, \ldots, x \in KU$, koja može sadržavati ponavljanje vrijednosti *i,j,...x*. Sve aktivnosti učenja nastale tijekom jedne aktivnosti Učenja pripadaju istoj sekvenci. Prilikom pokretanja aktivnosti Ponavljanja uvijek se kreira nova sekvenca koja se uvijek sastoji od jedne aktivnosti ponavljanja (koja se odnosi na odabrani pojam za kojeg se vrši ponavljanje, te jedne ili više aktivnosti Učenja, u slučaju slanja netočnih odgovora povezanih sa drugim pojmovima, npr. $P = \langle R_i, L_j, \ldots, L_x \rangle \mid i, j, \ldots, x \in KU$.

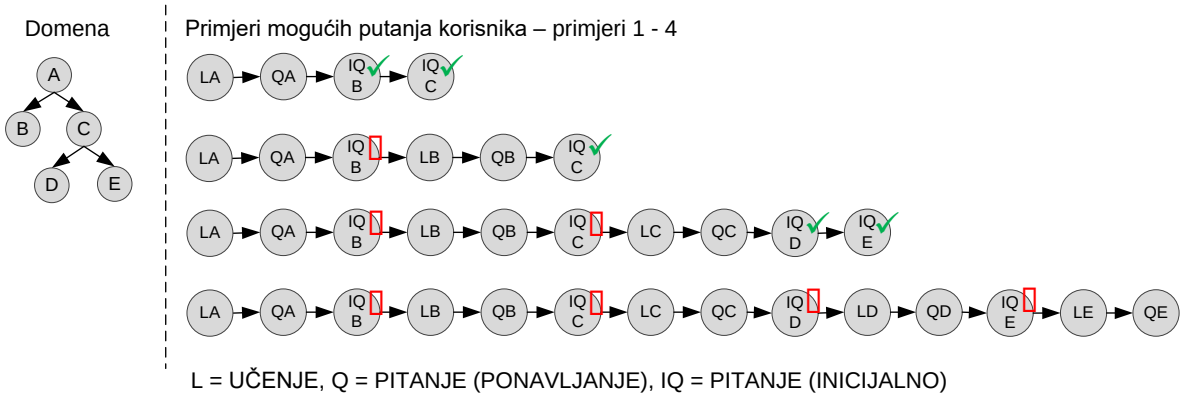
Prije pokretanja DAP s ciljem otkrivanja čestih i učinkovitih sekvenci, sve sekvence korisnika, nastale u svim sesijama koje je korisnik kreirao do trenutka pokretanja (nove) analize podataka, potrebno je:

a) procjeniti učinkovitost svih sekvenci korisnika,
b) transformirati skup sekvenci korisnika u dva skupa:
    a. „*prefiks*“ skup – niz aktivnosti/učenih pojmova prije odabranog pojma
    b. „*sufiks*“ skup – niz aktivnosti/učenih pojmova nakon odabranog pojma,
c) dodati sadržaj skupova u ukupni skup „prefiks“ i „sufiks“ sekvenci za klaster kojem pripada korisnik te pokrenuti postupak otkrivanja čestih i učinkovitih sekvenci.

Spomenuti koraci opisani su u sljedećim poglavljima.

3.3.1 Model procjene učinkovitosti putanja korisnika

Svaki element sekvence P se evaluira, nakon čega se P proširuje procjenjenom učinkovitošću aktivnosti EP, pa je $P = \langle (L_i, EP_{L_i}), (L_j, EP_{L_j}), \ldots, (L_x, EP_{L_x}) \rangle \mid i,j,..,x \in KU, EP \in \mathbb{R}$. Tijekom savladavanja domene znanja, korisnici samostalno odabiru početni pojam za učenje ili ponavljanje, nakon čega tutorski model preuzima vođenje odabranog procesa do njegovog završetka. Pri tome mogu nastati putanje različite duljine.



Slika 12. Primjeri nastanka različitih putanja kroz zamišljenu domenu znanja

Različita duljina putanja učenja može nastati uslijed dva razloga:

- različito stanje u modelu studenta
    - Prema zamišljenoj domeni znanja prikazanoj na slici 12, korisniku u_1 koji odabere pojam A, biti će ponuđeno jedno pitanje za provjeru znanja o pojmu A, te po jedno inicijalno pitanje za pojmove B i C. Ukoliko je u_1 prethodno savladao pojmove B i C, neće biti ponuđena inicijalna pitanja za spomenute pojmove.
- netočni odgovori na inicijalna pitanja
    - Prema zamišljenoj domeni znanja prikazanoj na slici 12, ukoliko korisnik u_1 odgovori točno na sva postavljena pitanja, putanja P_{u1} će se sastojati od tri elementa (duljinu putanje čine, odnosno biti će jednaka minimalnom mogućem broju elemenata za tog studenta, prema trenutnom stanju u njegovom modelu studenta (odgovara prvom primjeru putanje korisnika na slici 12). Ako korisnik odgovori netočno na neko od inicijalnih pitanja broj elemenata u P_{u1} može se povećati (može odgovarati nekim od primjera putanja 2, 3 i 4 na slici 12).

Na evaluaciju učinkovitosti svake putanje utječu sljedeće vrijednosti:

- pojmovi u domeni znanja mogu imati različite gornje granice (pragove) h_{ku_i} raspona vrijednosti za savladavanje pojma ku_i pa utjecaj točnog odgovora na postotak savladanosti pojma može biti manji ili veći u odnosu na točan odgovor na pitanje o nekom drugom pojmu ku_j koji ima manju ili veću vrijednost gornje granice (praga) h_{ku_j}
- neka pitanja mogu imati veću vrijednost uvećanja *wi* odnosno umanjenja stanja *wd* u modelu studenta ovisno o točnosti odgovora što se dovodi u vezu s h_{ku_i} u izrazu za izračun učinkovitosti putanje
- duljina putanje, tj. $|P_{u1}|$ ima sve veći negativni utjecaj na konačnu učinkovitost putanje što je veća od idealne putanje, odnosno najkraće moguće duljine $min|P_{u1}|$ (putanje s većim brojem netočnih odgovora su manje učinkovite od putanja s većim brojem točnih odgovora)
- najkraća moguća duljina putanje za odabrani pojam ku_a za korisnika u_i može se promjeniti u trenutku kada $v_{b,ui} = h_{ku_b}$ ili $v_{c,ui} = h_{ku_c}$ odnosno kada savlada jedan ili oba pojma podređena pojmu *A* u zamišljenoj domeni znanja prikazanoj na slici 12, što znači da je $min|P_{u1}|$ potrebno izračunati kod svake evaluacije putanje provjerom stanja u modelu studenta u trenutku kreiranja aktivnosti.

Tablica 4 prikazuje primjer mogućih iznosa promjena u modelu studenta nastalih pri kreiranju mogućih putanja prikazanih na slici 12. Inicijalna pitanja većinom uvećavaju ili umanjuju stanje u modelu studenta za određeni pojam za -0.1 do 0.2, dok pitanja za provjeru većinom kreiraju promjenu od -1,5 do 3.

Tablica 4. Mogući primjeri učinkovitosti i dužine putanja prikazanih na slici 12.

ku	***A***	***B***	***C***	***D***	***E***	$\sum\Delta v$	$\lvert ku\rvert$	$\lvert P_{u1}\rvert$	$min\lvert P_{u1}\rvert$	$\lvert Q_{Pu1}\rvert$	$\lvert IQ_{Pu1}\rvert$
h_{ku}	10	10	6	8	10						
	+2	+0.2	+0.2			2.4	1	3	3	1	2
	+3	-0.1 +2	+0.2			5.1	2	4	3	2	2
	+2	-0.1 +2	-0.1 +2	+0.2	+0.2	6.2	3	7	3	3	4
	+1	-0.1 +1	-0.1 +1	-0.1 +2	-0.1 +2	6.6	5	9	3	5	4

Izraz za izračun učinkovitosti putanje za aktivnosti Učenja (L) uključuje izračun promjena u modelu studenta nastalih temeljem broja odgovora na pitanja za ponavljanje (Q) i inicijalna pitanja (IQ), kao i točnosti na dane odgovore. Pri tome cq_i poprima vrijednost *wi* (iznos

uvećanja stanja u modelu studenta za pojam na koje se pitanje odnosi, u slučaju točnog odgovora) ukoliko je pitanje točno odgovoreno, odnosno *wd* (iznos umanjenja stanja u modelu studenta za pojam na koje se pitanje odnosi, u slučaju netočnog odgovora). Na jednak način vrijednost poprima ciq_i.

$$\mathrm{EP_i} = \left(\frac{\sum_{i=1}^{|\mathrm{KUi}|} \mathrm{cq_i}/\mathrm{h}_{ku_i}}{|\mathrm{Q_{Pu_i}}|} + \frac{\sum_{i=1}^{|\mathrm{KUi}|} \mathrm{ciq_i}/\mathrm{h}_{ku_i}}{|\mathrm{IQ_{Pu_i}}|} \right) * \frac{\min |\mathrm{P_{u1}}|}{|\mathrm{P_{u1}}|}$$

Izraz za izračun učinkovitosti putanje Ponavljanja (R) prilagođen je algoritmu procesa Ponavljanja opisanom u (Kovačić, 2002) i na slici 29. Tijekom procesa ponavljanja korisnik odgovara na 1 ili više pitanja o odabranom pojmu, pri čemu je moguće da, uslijed netočnog odgovora na pitanje i postojanja poveznice netočnog odgovora i nekog drugog pojma u domeni znanja (zadaje profesor pri kreiranju pitanja), korisnik bude preusmjeren na učenje povezanog pojma, nakon čega odgovara na jedno pitanje o tom pojmu te se u konačnici vraća na ponavljanje prvotno odabranog pojma.

Prvi dio izraza izračunava učinkovitost svih postavljenih pitanja Q_{Pu_i} za neki inicijalno odabrani pojam za ponavljanje ($\mathrm{ku_p} \in \mathrm{KU}$) na način da sumira omjer promjene (uvećanja ili umanjenja) u modelu studenta i praga odabranog pojma h_{ku_p} i dijeli dobivenu vrijednost s brojem postavljenih pitanja.

Drugi dio izraza izračunava učinkovitost danih odgovora na pitanja za ponavljanje svih učenih pojmova (ukoliko je to preusmjeravanja na učenje jednog ili više pojmova došlo tijekom procesa ponavljanja). Učeni pojmovi predstavljaju se oznakom $\mathrm{KP} \subseteq \mathrm{KU}$, a skup svih pitanja postavljenih nakon svakog učenog pojmova označava se s $\mathrm{Q_{KP}}$. Prema algoritmu procesa Ponavljanja, nakon prikaza materijala za učenje postavlja se samo jedno pitanje zbog čega je drugi dio izraza istovjetan prvom dijelu prethodnog izraza.

Prvi i drugi dio izraza množe se omjerom idealne duljine putanje (za proces Ponavljanja kao idealna dužina uzima se vrijednost 1) i stvarne duljine putanje (koja se može produljiti preusmjeravanjem na učenje pojma(ova) uslijed netočnih odgovora).

$$\mathrm{EP_i} = \left(\frac{\sum_{i=1}^{|\mathrm{Q_{Pu_i}}|} \mathrm{cq_i}/\mathrm{h}_{ku_i}}{|\mathrm{Q_{Pu_i}}|} + \frac{\sum_{i=1}^{|\mathrm{KP}|} \mathrm{cq_i}/\mathrm{h}_{ku_i}}{|\mathrm{Q_{KP}}|} \right) * \frac{1}{1 + |\mathrm{KP}|}$$

Evaluacija učinkovitosti putanja implementirana je u Algoritmu 4.

3.3.2 Oblikovanje skupa putanja korisnika

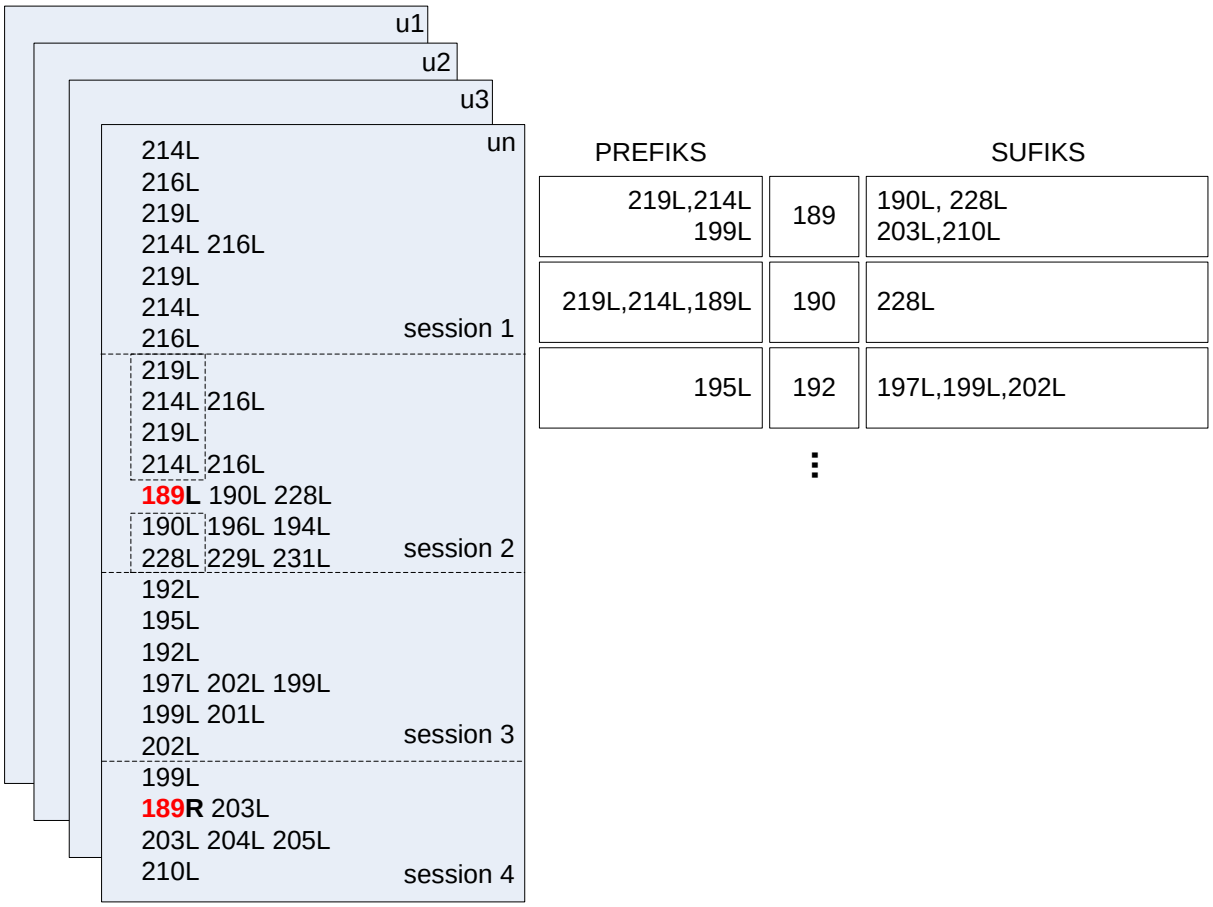
Cilj postupka oblikovanja skupa putanja je transformacija skupa putanja svakog korisnika u oblik skupa podataka koji omogućava provedbu dubinske analize putanja s ciljem otkrivanja čestih i učinkovitih putanja korisnika. S obzirom da je cilj analize dobiti zasebne skupove sugestija koje će se nuditi korisniku pri početku odnosno po završetku aktivnosti Učenja i Ponavljanja, razvijen je algoritam transformacije putanja koji uključuje i evaluaciju njihove učinkovitosti prema izrazima opisanima u poglavlju 3.3.1.

Na lijevom dijelu Slike 13 dan je primjer početnog skupa putanja koje je kreirao korisnik prilikom savladavanja domene znanja. Korisnik je imao četiri prijave u sustav, odnosno kreirao četiri sesije, tijekom kojih je kreirao više aktivnosti.

Algoritam za transformaciju skupa putanja, kreira novi skup podataka koji za svaki pojam u domeni znanja, kreira skup putanja, koje sadrže nizove aktivnosti koje je kreirao korisnik prije početka učenja pojma („*prefiks"*), odnosno nakon završetka učenja pojma („*sufiks"*). Algoritam kreira dva privremena skupa (*tempPrefix* i *tempSufix*) u koje se dodaju elementi jedne transformirane putanje, koja se dodaje glavnom skupu putanja korisnika. Oba skupa mogu sadržavati maksimalno šest elemenata. Na temelju preliminarnih istraživanja, te analize drugih znanstvenih radova (Nkambou, 2011), uočeno je da veći broj elemenata ima slabu povezanost s odabranim pojmom, odnosno da aktivnost učenja pojma sedam ili više aktivnosti prije pojma za koji se kreira skup putanja ima malu vjerojatnost da je dio strategije korisnika za savladanje pojma za koji se kreira skup putanja. Pri tome se za pomicanje *tempPrefix* koristi metoda *FIFO* (eng. *first in, first out*) kako bi se omogućilo njegovo pomicanje od početka skupa podataka (u trenutnoj sesiji), do traženog pojma. Na slici 13, pojam za koji se kreira skup putanja ima identifikator ***189***, a *tempPrefix* je prikazan kao pravokutnik isprekidanog obruba iznad prvog pronađenog indikatora traženog pojma u sesiji 2 korisnika u_n. U skupovima *tempPrefix* i *tempSufix* ne može se pojaviti više pojmova s istim identifikatorom pojma i aktivnosti kako bi se izbjeglo dobivanje putanja s vrlo malom informacijom (npr. 190L, 190L, 190L). Može se pojaviti više pojmova s istim identifikatorom pojma, ali različitim indikatorom aktivnosti (npr. 190L, 190R), ali bez ponavljanja parova aktivnosti.

Nakon pronalaska traženog pojma i dodavanja *tempPrefix* u glavni skup putanja *prefiks* kojeg čine sve putanje svih studenata jednog klastera, počinje se popunjavati *tempSufix*. Ukoliko se

u *tempSufix* doda maksimalnih šest elemenata ili se dosegne kraj sesije, *tempSufix* se dodaje u *sufiks* i nakon čega se prazni. Na slici 13 *tempPrefix* je prikazan kao pravokutnik isprekidanog obruba ispod prvog pronađenog indikatora traženog pojma u sesiji 2 korisnika u_n.



Slika 13. Primjer početnog i transformiranog skupa putanja korisnika

Ukoliko se indikator traženog pojma (na slici 13 je to *189*) pojavljuje u istoj sesiji nakon manje od šest aktivnosti, odnosno prije nego *tempSufix* dosegne maksimalan broj elemenata, velika je vjerojatnost da su aktivnosti nastale između dvaju aktivnosti pojma *189* nastale kao dio strategije učenja trenutnog korisnika kako bi uspješnije savladao pojam *189*.

Iz tog razloga uvedeno je pravilo kojim se definira prebacivanje elemenata iz *tempSufix* u *tempPrefix* i autmatsko dodavanje *tempPrefix* u *prefiks*, ukoliko se indikator traženog pojma pronađe prije nego *tempSufix* dosegne maksimalan broj elemenata (6). Kako su putanje koje se sastoje od više aktivnosti uvjetovane netočnim odgovorima na inicijalna pitanja, ne mogu se dodavati u putanje *sufiks* ili *prefiks* skupa, već se preskaču.

Algoritam 4: Transformacija skupa putanja

Ulaz: u_i – identifikator korisnika, *KU* – skup pojmova u domeni znanja

Izlaz: TP_{ui} – transformirani skup putanja korisnika

```
PS = privremeni skup podataka
// evaluacija
sesije = dohvati sve sesije korisnika (ui)
za svaki s iz sesije
   PS[s] = 0
   putanje = dohvati sve putanje korisnika ui u sesiji s
   za svaki p iz putanje
      ep = evaluacija_putanje(p)
      PS[s] =[p,ep]
// transformacija
SP = skup transformiranih putanja
prefiks={},sufiks={}
za svaki ku iz KU
   tempPrefix = {}, tempSufix = {}
   za svaki ps iz PS
      transformiraj_putanju(ps)   // popunjava prefiks i sufiks
   SP[ku]={prefiks, sufiks}
vrati SP
```

Algoritam 4 izvodi transformaciju skupa podataka o putanjama korisnika kroz sve sesije korisnika, te se time ostvaruje ključan preduvjet za provedbu postupka otkrivanja čestih i učinkovitih putanja opisanog u sljedećem poglavlju. Algoritam 4 poziva se iz Algoritma 5 kako bi se kreirao konačan skup svih *prefiks* i *sufiks* putanja svih članova klastera.

3.3.3 Postupak otkrivanja čestih i učinkovitih putanja

Nakon pripreme skupa podataka provodi se dubinska analiza s ciljem otkrivanja čestih i učinkovitih putanja za svaki pojam u domeni znanja, za oba skupa putanja (*prefiks* i *sufiks*). Ovaj postupak oslanja se na algoritam USPAN (Yin, 2012) kojem se, kroz podsustav za komunikaciju s alatima za dubinsku analizu podataka šalje pripremljeni skup putanja. USPAN pripada skupini algoritama za otkrivanje visoko učinkovitih skupova (eng. *high-utility*

itemsets mining - HUIM) u bazi putanja, ali je specifičan po tome što kombinira algoritam za otkrivanje visoko učinkovitih skupova s algoritmom za otkrivanje čestih sekvenci (eng. *high-utility sequential pattern mining - HUSPM*) odnosno uzima u obzir redoslijed elemenata pri otkrivanju čestih uzoraka. USPAN omogućuje proširenje osnovnog skupa sekvenci (koji se kod ostalih SPM algoritama sastoji isključivo od numeričkih identifikatora elemenata putanja – primjerice pojmova u domeni znanja) informacijom o vrijednosti (eng. *utility*, *profit*) svakog elementa sekvence, te sumom vrijednosti cijele sekvence.

Tablica 5. Primjer ulaznog skupa podataka – baze sekvenci za USPAN algoritam

	Sekvence	**Vrijednost sekvence**
s1	{1[1],2[4]},{3[10]},{6[9]},{7[2]},{5[1]}	27
s2	{1[1],4[12]},{3[20]},{2[4]},{5[1],7[2]}	40
s3	{1[1]},{2[4]},{6[9]},{5[1]}	15
s4	{1[3],2[4],3[5]},{6[3],7[1]}	16

Tablica 5 prikazuje primjer ulaznog skupa podataka za USPAN algoritam koji se sastoji od četiri sekvence u kojoj se svaki element sastoji od numeričkog identifikatora objekta, te vrijednosti objekta. Iz primjera je vidljivo da se sekvence mogu sastojati od skupova objekata (eng. *itemsets*) odvojenih vitičastim zagradama. Pri transformaciji skupa putanja korisnika u prefiks i sufiks skupove, u konačan skup podataka uzimane su samo prve aktivnosti u putanjama koje su se sastojale od više aktivnosti, pa se skupovi ne pojavljuju u konačnom skupu putanja.

USPAN algoritam prihvaća četiri argumenta:

- *ulazna_datoteka* – baza sekvenci s vrijednostima
- *izlazna_datoteka* – naziv datoteke u koju će se pohraniti rezultat analize
- *minimalna_vrijednost* – minimalna vrijednost otkrivenog uzorka kako bi bio uvršten u rezultate analize
- *maksimalna_duljina* – maksimalna duljina, odnosno broj elemenata u otkrivenoj čestoj i učinkovitoj sekvenci

Za potrebe analize podataka o putanjama, pri pokretanju analize unaprijed su definirane početne vrijednosti argumenata:

- *ulazna_datoteka* – naziv datoteke se kreira dinamički
- *izlazna_datoteka* – naziv datoteke se kreira dinamički

- *minimalna_vrijednost – 10*, na temelju rezultata predistraživanja, otkrivene sekvence sortiraju se od najviše vrijednosti prema najnižoj, pa *minimalna_vrijednost* ima samo ulogu eliminiranja neučinkovitih putanja, dok će se kod odabira pojmova za sugestije uvijek koristiti najučinkovitije putanje
- *maksimalna_duljina – 6,* prema ranije određenoj maksimalnoj duljini putanja

Algoritam za provedbu postupka otkrivanja čestih i učinkovitih putanja korisnika, koristi podsustav za komunikaciju s alatima za DAP (osjenčane linije) koji pokreće USPAN algoritam nad svakim skupom putanja zasebno – za prefiks i sufiks skup svakog pojma u domeni znanja, te za svaki klaster zasebno.

Algoritam 5: Provedba postupka otkrivanja čestih i učinkovitih putanja korisnika

Ulaz: M[k][studenti] - model klasteriranja korisnika
KU – skup pojmova u domeni znanja

Izlaz: PT – skup otkrivenih čestih i učinkovitih putanja za svaki klaster za svaki pojam

za svaki *k* u M[k][studenti]

$SP = \{(prefiks, sufiks)_1, (prefiks, sufiks)_2, \ldots, (prefiks, sufiks)_n\} \mid n \in KU$

$PT = \{(prefiks, sufiks)_1, (prefiks, sufiks)_2, \ldots, (prefiks, sufiks)_n\} \mid n \in KU$

za svaki *student* u *studenti*

$SP(prefiks,sufiks)$ = Algoritam 4(*KU*)

za svaki *ku* u *KU*

ako je $|SP(prefiks)_{ku}| \geq 5$

$PT(prefiks,učinkovitost)_{ku}$ = kreiraj API poziv(*SPMF,USPAN, 10, 6,* $SP(prefiks)_{ku}$)

ako je $|SP(sufiks)_{ku}| \geq 5$

$PT(sufiks,učinkovitost)_{ku}$ = kreiraj API poziv(*SPMF,USPAN, 10, 6,* $SP(sufils)_{ku}$)

vrati *PT*

Uvjet za pokretanje algoritma za otkrivanje čestih i učinkovitih putanja za određeni pojam je da se u skupu putanja nalaze putanje barem pet korisnika. Time se izbjegava pokretanje analize na premalom skupu putanja, na temelju kojeg nije moguće dobiti relevantne putanje. Također se osigurava da će se u skupu nalaziti minimalno pet putanja, ali ih može biti znatno više u ovisnosti od intenziteta aktivnosti korisnika na pojmu.

3.4 Unaprijeđeni tutorski model sustava

Postupci i algoritmi opisani u prethodnim poglavljima pokreću se periodički nad rastućim skupom podataka o interakcijama korisnika. Unaprijeđeni tutorski model sustava koristi se rezultatima provedenih analiza kako bi korisnicima pri svakom pokretanju i završavanju aktivnosti Učenja ili Ponavljanja, ponudio sugestije o tome koji pojam učiti prije odnosno nakon odabranog pojma.

Algoritam 6: Algoritam za odabir sugestija

Ulaz: u_i – korisnik $\in U$
$k(o)$ – klaster i ocjena klastera kojem pripada korisnik
ku_j – odabrani pojam $\in KU$
ps – prefiks ili sufiks $\in \{P,S\}$
Izlaz: ku_m – sugerirani pojam $\in KU$

```
bolji_klaster = k(o+1)
P(klaster_korisnika)= dohvati putanje (ku_j , k(o), ps)
ako je |P(klaster_korisnika)|>0
   sortiraj putanje po učinkovitosti (silazno)
   za svaku putanja  iz P
      elementi_putanje = podijeli na pojmove(putanja)
      za svaki p u elementi_putanje
      ako je p ∩ C_KUi = ∅
         temp_sugestija = p
         sugerirani = dohvati ranije sugerirane pojmove (u_i)
         ako je temp_sugestija ∩ sugerirani = ∅
            sugestija = temp_sugestija
           vrati sugestija
inače
   P(bolji_klaster)= dohvati putanje (ku_j , bolji_klaster, ps)
   ako je |P(bolji_klaster)|>0
   sortiraj putanje po učinkovitosti (silazno)
   za svaku putanja_bk iz P
      elementi_putanje_bk = podijeli na pojmove(putanja_bk)
      za svaki p_bk u elementi_putanje_bk
         ako je p_bk ∩ C_KUi = ∅
            temp_sugestija = p_bk
            sugerirani = dohvati ranije sugerirane pojmove (u_i)
            ako je temp_sugestija ∩ sugerirani = ∅
               sugestija = temp_sugestija
ako !sugestija
    sugestija = sugestija_koju_su_korisnici_najviše_koristili(ku_j, k(o), ps)
ako !sugestija
   vrati ∅
```

Ukoliko u trenutku pokretanja aktivnosti, podsustavi za dubinsku analizu podataka nisu pronašli podatke na temelju kojeg bi se one kreirale, sugestije se korisniku ne prikazuju, te

korisnici koriste postojeće funkcionalnosti sustava odnosno tutorski model sustava vodi aktivnosti Učenja i Ponavljanja na jednak način kao i prije proširenja arhitekture sustava metodama dubinske analize podataka.

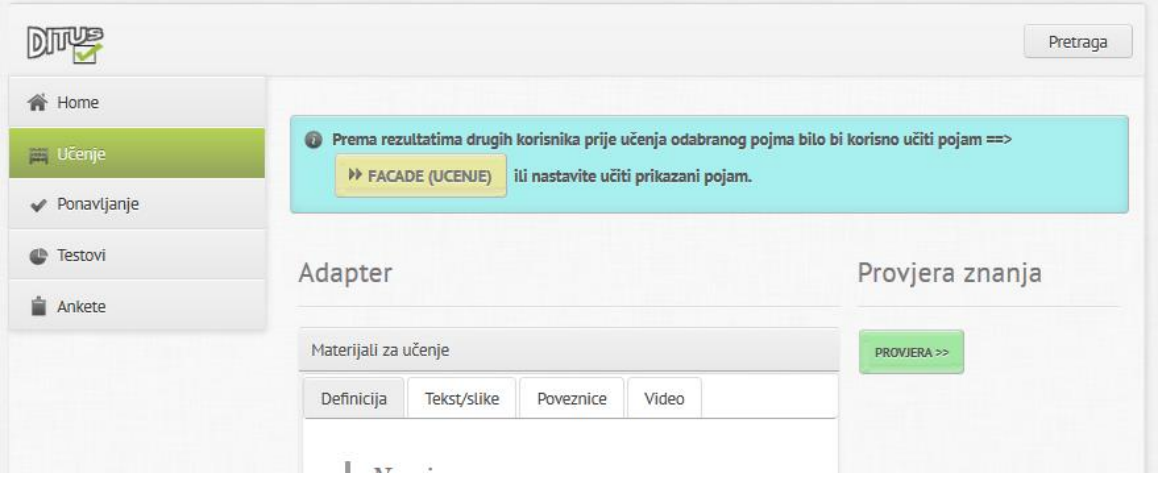
Algoritam 6. dohvaća sugestiju na temelju odabranog pojma za učenje/ponavljanje, informacije o pripadnosti korisnika klasteru, te skupa otkrivenih čestih i učinkovitih putanja vezanih za odabrani pojam.

Prošireni tutorski model sustava Algoritmu 6 također daje informaciju o tome u kojem skupu otkrivenih putanja je potrebno pronaći sugestiju – prefiks ili sufiks. Pronađene sugestije prezentiraju se korisniku u obliku poruke i poveznice ovisno o vrsti i statusu aktivnosti (Tablica 6).

Tablica 6. Pregled poruka korisniku za odabir sugestije

Početak aktivnosti Učenja ili Ponavljanja	*Prema rezultatima drugih korisnika prije učenja odabranog pojma bilo bi korisno učiti pojam {**sugestija**} ili nastavite učiti odabrani pojam.*
Kraj aktivnosti Učenja ili Ponavljanja	*Što dalje? Sugerirani pojam je: {**sugestija**} ili se vratite na popis pojmova*

Primjer prikaza sugestija u sučelju korisnika prikazan je na slici 14.



Slika 14. Primjer poruka korisniku u sučelju sustava

4 Implementacija sustava

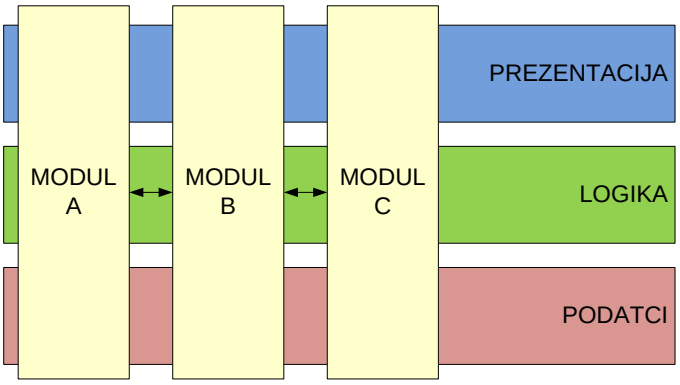
U ovom poglavlju opisuje se način implementacije opisanog teorijskog modela kao prototipa programskog sustava u obliku Web aplikacije. Sustav je nazvan DITUS – *Department of Informatics TUtoring* System. U Poglavlju 2 naveden je podatak o relativno malom broju tutorskih sustava temeljenih na Web-u u odnosu na tutorske sustava rezvijene kao samostalne (eng. *desktop*) aplikacije, pa ovaj sustav predstavlja novi doprinos razvoju takvih sustava. Posebno treba istaknuti povezanost sustava s alatima za dubinsku analizu podataka, te implementaciju cjelovitog postupka dubinske analize obrazovnih podataka, kao i postupaka za automatsku interpretaciju dobivenih rezultata, po čemu je ovaj sustav, prema našim informacijama, jedinstven.

Sustav je implementiran kao web aplikacija na tzv. LAMP (Linux, Apache, MySQL, PHP) platformi. Dakle, nije riječ o razvojnoj platformi odnosno skupu klasa za razvoj softvera (kakve su primjerice Java EE ili .NET) već o infrastrukturi za razvoj i puštanje u rad (eng. *deployment*) koja se sastoji od (poslužiteljske verzije) operativnog sustava Linux, web poslužitelja Apache (Apache, 2016), poslužitelja relacijske baze podataka MySQL (MySQL, 2016), te poslužiteljskog skriptnog jezika PHP (PHP, 2016). Za sam razvoj korišteno je integrirano razvojno okruženje NetBeans 8.0, dok je za čuvanje verzija sustava korišten sustav Git (GitSCM, 2016).

Prema definiciji (Kappel, 2005), web aplikacije su interaktivni sustavi koji se isporučuju i koriste putem usluge Interneta nazvane World Wide Web, koji se temelji na HyperText Transfer Protokolu. Interaktivnost ostvaruju primjenom obrazaca i hiperveza kojima se korisnicima omogućuje unos, dohvaćanje i manipulacija podacima iz (relacijske, objektne, noSQL) baze podataka.

Arhitektura takvih sustava tradicionalno se opisuje „troslojnom“ (eng. *3-tier*), pri čemu se slojevi dijele na sloj za pohranu i pristup podacima, sloj poslovne logike koji manipulira podacima i upravlja radom aplikacije, te prezentacijski sloj odnosno sučelje koje služi za prikaz podataka, te omogućavanje interakcije korisnika sa sustavom. Razvijeni prototip sustava pored spomenute arhitekture, primjenjuje i uzorak dizajna *Model-View-Controller* (MVC) (Krasner, 1988) koji pripada skupini uzoraka dizajna arhitekture softvera, čime se svaka od (ključnih) funkcionalnosti softvera dijeli na spomenute tri razine. Usporedba „tradicionalne“ troslojne arhitekture i MVC arhitekture prikazana je na slici 15. Primjenom MVC uzorka, implementacija (kôd) svakog modula aplikacije dijeli se u tri dijela odnosno u tri mape nazvane *Models*, *Views* i *Controllers*. Ključni dio aplikacije postaje datoteka

„index.php“ odnosno početna stranica aplikacije koja dohvaća postavke aplikacije, te povezuje sve dijelove u jednu funkcionalnu cjelinu (aplikaciju).



Slika 15. Arhitektura 3-slojnih i MVC aplikacija

Pri razvoju sustava DITUS korištena je platforma Phalcon (Phalcon, 2016) koja podrazumijeva implementaciju MVC uzorka, a specifična je po tome što je njezina jezgra kompajlirana, te se implementira u Apache poslužitelj odnosno njegov PHP interpreter kao zasebni modul. Razlog za to je značajno poboljšanje performansi rada platforme u odnosu na druge, slične platforme za razvoj web aplikacija.

Kako bi se omogućilo korištenje aplikacije na različitim uređajima (PC, tableti, pametni telefoni), odnosno prilagodljivost sučelja različitim rezolucijama ekrana (engl. *responsive design*) implementiran je stilski predložak nazvan Whitelabel razvijen primjenom najnovijih Web tehnologija (HTML5 (HTML, 2016), CSS3 (CSS3, 2016)), te platforme JQuery (JQuery, 2016) i JQuery Mobile (JQuery, Mobile, 2016) čime je sustav optimiziran za rad s ekranima osjetljivima na dodir.

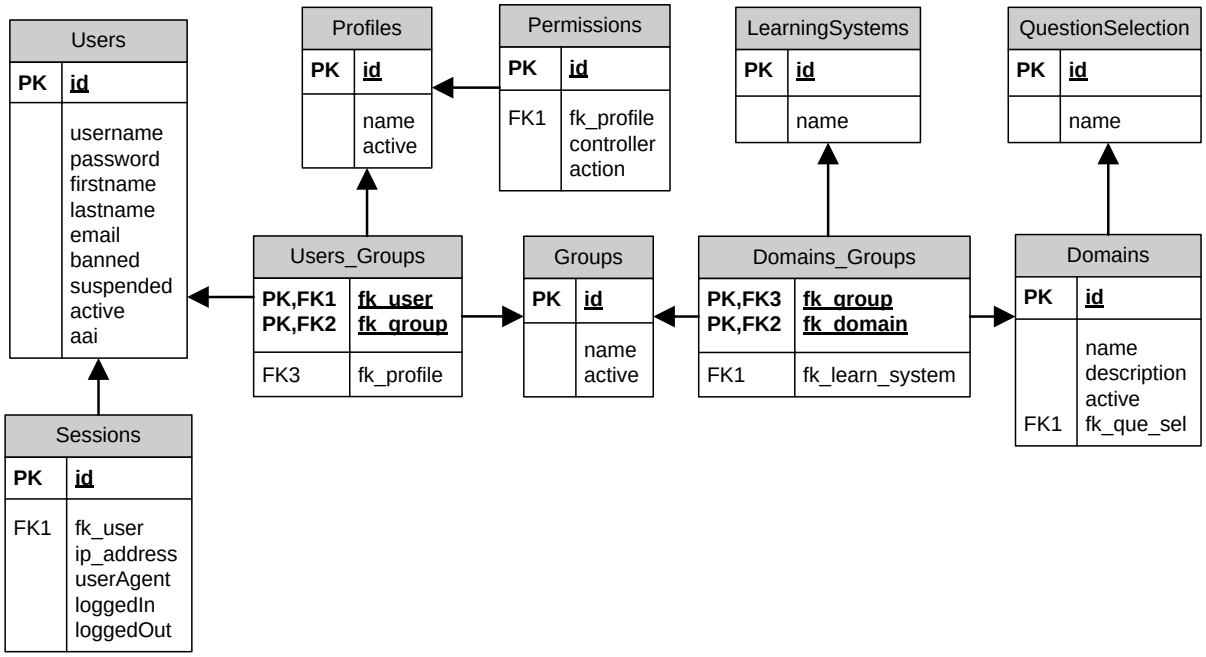
Korisnici sustava su administratori, nastavnici i studenti. Administrator upravlja osnovnim postavkama sustava, definira grupe studenata, te ih povezuje s domenama. Nastavnik popunjava domenu znanja pojmovima, povezuje ih u strukturu, te za svaki pojam unosi nastavne materijale i pitanja (inicijalna, ispitna te pitanja za ponavljanje). Ispitna pitanja koriste se isključivo za završnu provjeru znanja, nakon savladavanja domene. Studenti koriste sustav za savladavanje domene znanja, pri čemu tutorski model prilagođava strukturu učenja trenutnom stanju u modelu studenta, dok proširenja razvijena u ovom istraživanju omogućuju i prikaz sugestija kreiranih temeljem provedbe dubinske analize podataka o interakcijama studenata sa sustavom.

4.1 Baza podataka

Model baze podataka razvijen je metodom entiteti-veze (Chen, 1976). Za pohranu podataka sustava korištena je relacijska baza podataka. Prema izrađenim modelima baza je razvijena na MySQL poslužitelju alatom PhpMyAdmin. Model baze podataka razdvojen na nekoliko niže prikazanih i opisanih podmodela radi bolje razumljivosti. Neke relacije pojavljuju se u više modela pri čemu se (kod ponovnog pojavljivanja) prikazuju samo atributi važni za prikazani model.

4.1.1 Upravljanje korisnicima

Dio modela baze podataka za upravljanje korisnicima prikazan je na slici 16. Prilikom prve prijave putem AAI sustava korisnici se registriraju u sustavu. Relacija Groups i s njome povezane relacije daju fleksibilnost u definiranju pristupa korisnicima. Agregacija „Domains_Groups“ određuje način rada sustava za studente određene grupe čime se omogućava provedba eksperimenta opisanog u Poglavlju 6. Za potrebe ovog istraživanja razvijen je prototip sustava koji omogućuje definiranje načina rada tutorskog sustava (*Learning Systems*) koji omogućuje kreiranje kontrolne skupine studenata za koje će sustav demonstrirati postojeće funkcionalnosti, dok će za eksperimentalnu skupinu demonstrirati nove funkcionalnosti (sugestije) potpomognute dubinskom analizom podataka.

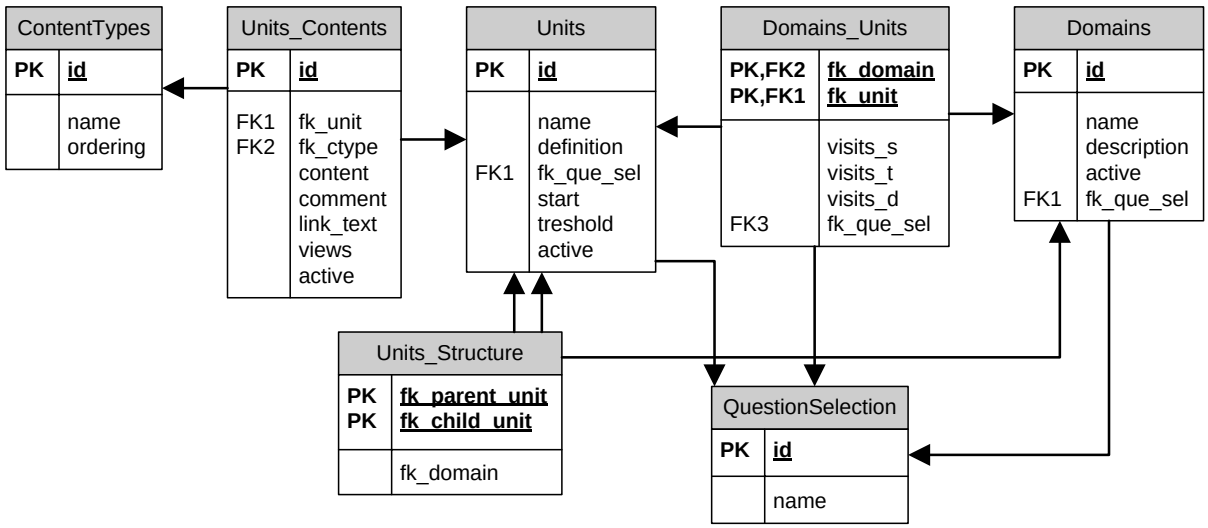


Slika 16. Dio modela baze podataka za upravljanje korisnicima

Kontrola pristupa pojedinim funkcionalnostima aplikacije izvedena je definiranjem uloge odnosno profila (*Profiles*) svakog korisnika u svakoj domeni, te pojedinačnim definiranjem skupa funkcionalnosti dostupnih svakom profilu (*Permissions*).

4.1.2 Upravljanje sadržajima u domenama znanja

Dio modela baze podataka za upravljanje sadržajima domena znanja prikazan je na slici 17. Nakon kreiranja nove domene znanja, nastavnik kreira pojmove (*Units*) u domeni, pri čemu se za svaki pojam unosi naziv, kratka definicija, te početna i konačna vrijednost (prag) savladanosti pojma. Pojam može biti skriven od korisnika domene (atribut *active*) dok nastavnik ne izradi sve potrebne sadržaje, pitanja, te ne definira mjesto pojma u strukturi domene (*Units_Structure*). Nakon unosa naziva i definicije pojma, nastavnik unosi proizvoljan broj sadržaja (*Units_Contents*) koji mogu biti nekog od definiranih tipova (*ContentTypes*): hipertekst, primjeri kôda, vanjske poveznice, video materijali.



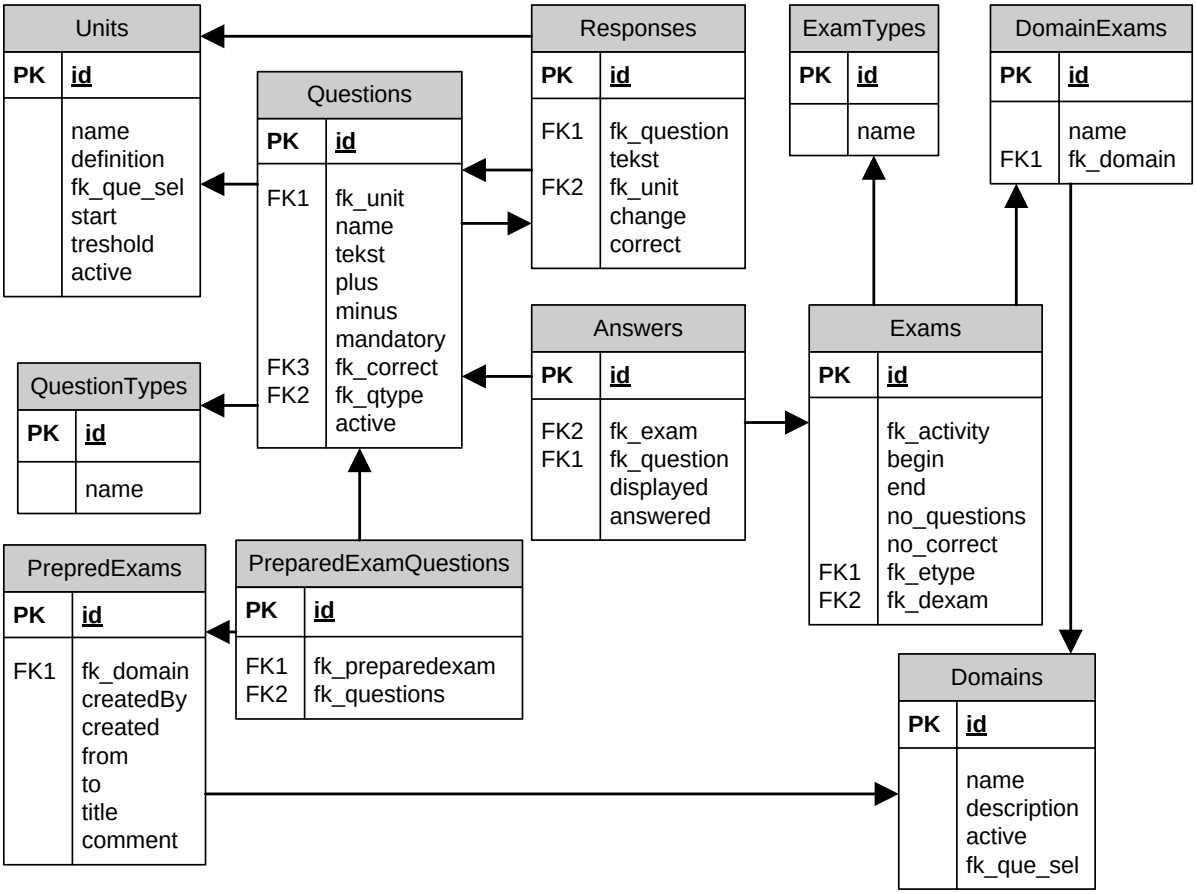
Slika 17. Dio modela baze podataka za upravljanje domenom znanja

Za potrebe statističke analize u bazi se bilježi broj pristupa pojmu po svakom od tri vrste rada sustava, odnosno načina učenja (slobodno učenje, tutoring, tutoring podržan dubinskom analizom podataka).

4.1.3 Pohrana podataka o provjerama znanja

Dio modela baze podataka za pohranu podataka o provjerama znanja prikazan je na slici 18. Prilikom unosa pitanja (*Questions*) za pojam, nastavnik definira tip pitanja (*QuestionTypes*), određuje iznos promjene u modelu studenta (atributi *plus, minus*), te da li je pitanje obavezno

(atribut *mandatory*). Nakon unosa pitanja unosi se proizvoljan broj odgovora (*Responses*), uz mogućnost označavanja više odgovora kao točnih (u tom slučaju se korisniku prikazuju *checkbox* elementi u obrascu za slanje odgovora). Također, svaki netočan odgovor može biti povezan s nekim pojmom u domeni znanja (na nepoznavanje kojeg odabir tog, netočnog odgovora, ukazuje). Ukoliko nastavnik poveže netočan odgovor s pojmom, može unijeti i iznos negativne promjene u matrici znanja za povezani pojam. Ova mogućnost koristi se kod procesa ponavljanja, dok se kod procesa učenja ignorira. Prilikom svakog odgovaranja na pitanja, bilo tijekom procesa učenja (kada se korisniku mogu prikazati i pitanja tipa „ponavljanje“ ili tipa „inicijalno“), bilo tijekom procesa ponavljanja (kada se korisniku mogu prikazati samo pitanja tipa „ponavljanje“) unosi se novi zapis u relaciju ispiti (*Exams*), dok se sva odgovorena pitanja u tom procesu/ispitu, pohranjuju u relaciju *Answers*. Također se, za potrebe analize podataka pohranjuje vrijeme prikaza i odgovora na pitanje.

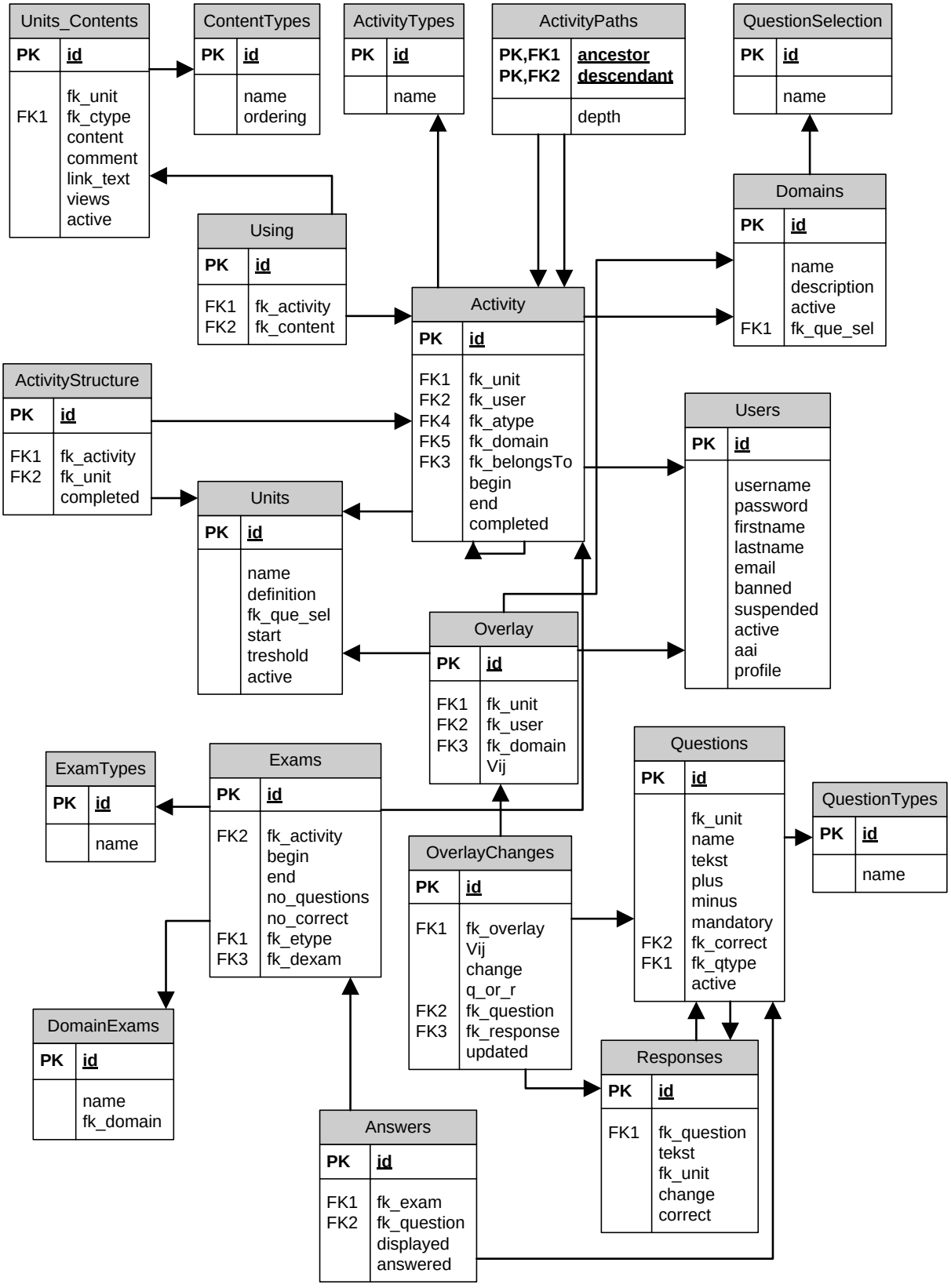


Slika 18. Dio modela baze podataka za provjere znanja

Nastavnik može pripremiti testove znanja i u njih dodati proizvoljan broj pitanja tipa „ispitno“. Također je moguće definrati u kojem je razdoblju ispit dostupan korisnicima. Nakon kreiranja, korisnik može pristupiti ispitu korištenjem odgovarajuće funkcionalnosti.

4.1.4 Pohrana podataka o aktivnostima korisnika

Dio modela baze podataka za pohranu podataka o aktivnostima korisnika prikazan je na slici 19. Ključna relacija ovog podmodela je relacija za pohranu aktivnosti korisnika (*Activities*) koju određuje tip (*ActivityTypes*) koji odgovara ključnim procesima/funkcionalnostima sustava – Učenje ('L') i Ponavljanje ('R'). S obzirom da aktivnosti korisnika mogu biti hijerahijski povezane postoji veza relacije same sa sobom omogućene uvođenjem atributa *belongs_to*. Zbog neadekvatnosti relacijskih baza podataka za pohranu hijerarhijskih struktura podataka, te zbog potreba naknadne analize podataka, uvedena je relacija *ActivityPaths* koja implementira pristup *closure tables* (Killaby, 2016) odnosno pohranjuje sve hijerahije aktivnosti. Prilikom kreiranja svake nove aktivnosti tipa: „Učenje“, sustav analizira stanje u prekrivenom (*Overlay*) modelu studenta koji sadrži trenutnu razinu savladanosti (*Vij*) svakog pojma u domeni, za svakog studenta u sustavu. Na temelju stanja u modelu studenta, sustav definira strukturu povezanih i nesavladanih pojmova niže razine (*ActivityStructure*) za koje sustav treba postaviti inicijalna pitanja, nakon provjere znanja o prvotno odabranom pojmu. Spomenute relacije su ključne za provedbu procesa učenja i ponavljanja.



Slika 19. Dio modela baze podataka za aktivnosti učenja i model studenta

Sve promjene u modelu studenta pohranjuju se u relaciju *OverlayChanges* kako bi se mogao re-kreirati put studenta, odnosno kreirati krivulje učenja (Amari, 1992) za svaki pojam, za

svakog studenta ili na razini domene znanja. Kod procesa učenja bilježi se i koje je sadržaje student koristio (*Using*) što se kasnije može koristiti kod analize procesa učenja.

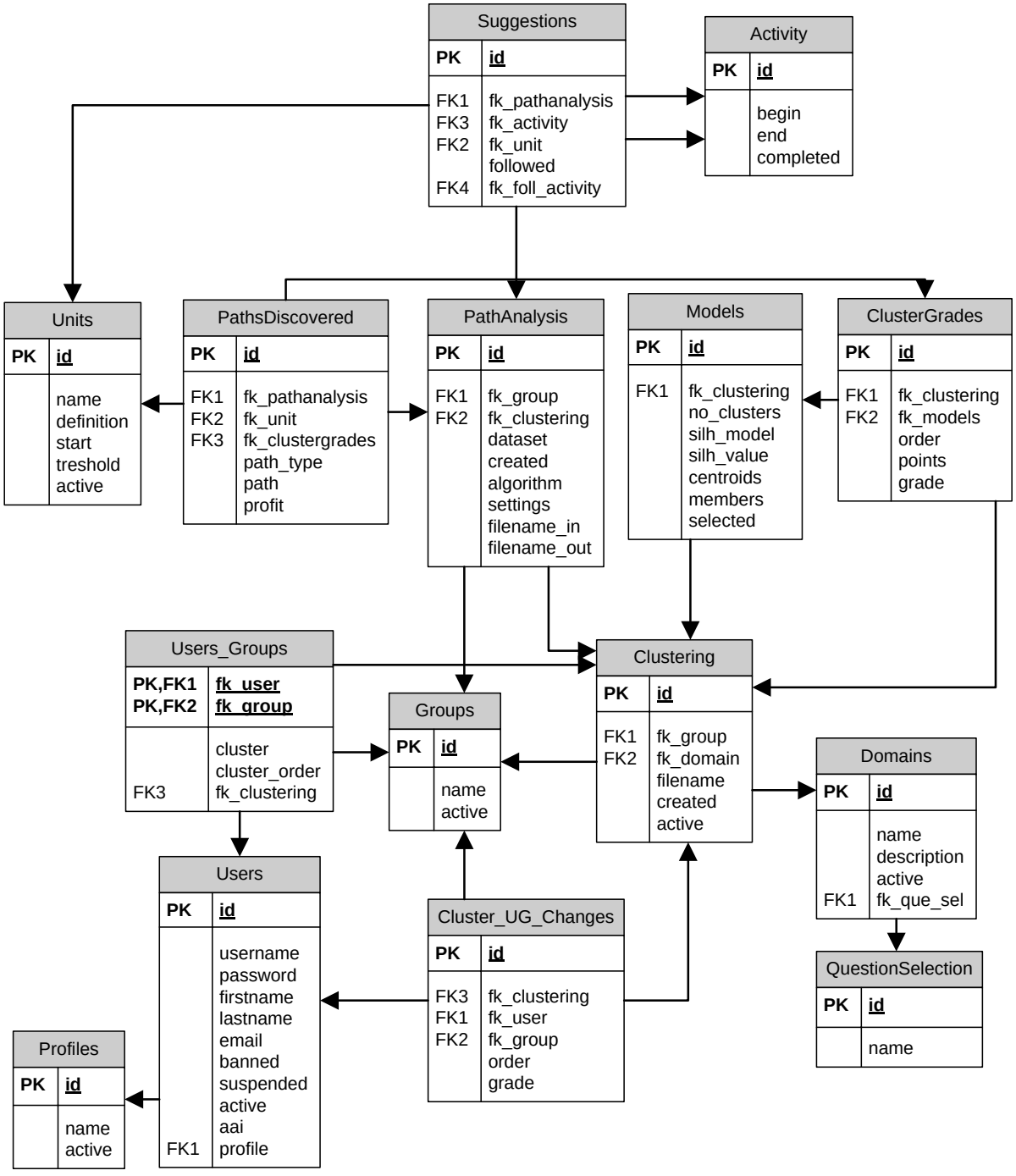
4.1.5 Pohrana podataka o DAP i sugestijama

U skladu s modelima pripreme i provedbe dubinske analize podataka prezentiranima u Poglavlju 4. razvijen je relacijski podmodel za pohranu podataka o provedenim analizama. Preciznije, omogućena je pohrana podataka o provedenim analizama grupiranja/razredovanja studenata, otkrivenim čestim i učinkovitim putanjama, te u konačnici ponuđenim sugestijama kreiranima na temelju otkrivenih putanja.

Dio modela baze podataka za pohranu podataka o dubinskim analizama i sugestijama prikazan je na slici 20. Kao što je opisano u Algoritmu 4, prva analiza koja se provodi je klasteriranje (*Clustering*), pri čemu nastaje više modela (*Models*), od kojih se odabire (atribut: *selected*) model s najvišom vrijednosti siluete (atribut: *silh_value*). Zatim se provodi evaluacija kvalitete procesa učenja svake skupine/klastera u odabranom modelu (*ClusterGrades*), te u konačnici ažurira model studenta. U ovom modelu se u relaciji *User_Groups* pojavljuju atributi *fk_clustering* (poveznica prema pripadajućoj analizi), *cluster_order* (slovna oznaka redoslijeda grupe), te *cluster* (redni broj otkrivenog klastera u modelu). Također se bilježe sve promjene pripadnosti korisnika klasteru (*Cluster_UG_Changes*), kako bi se, prilikom analize, moglo uočiti kretanje korisnika kroz klastere različitih razina, odnosno procjeniti koliko je sustavu primjerice trebalo da izvrsnog studenta smjesti u najbolje ocjenjeni klaster (ukoliko je sustav počeo koristiti kasnije od drugih studenata vjerojatno će pripadati klasteru ocjenjenom nižom ocjenom). Ukoliko želi, nastavnik može napraviti određene ispravke rezultata automatske evaluacije (odabira modela) ili interpretacije (procjene kvalitete procesa učenja po skupinama odabranog modela) klastering analize, te po završetku aktivirati klastering analizu (atribut: *active*) nakon čega sustav počinje koristiti tu analizu i s njom povezane rezultati analize putanja.

Nakon provedbe analize i evaluacije klastera, provodi se proces otkrivanja čestih i učinkovitih putanja studenata svakog klastera. Pri tome se u bazu podataka pohranjuje svaka provedba analize putanja (*PathAnalysis*) koja sadrži poveznicu prema aktivnoj analizi klasteriranja, te ostale podatke koji mogu biti potrebni za naknadnu analizu podataka. Sve putanje otkrivene USPAN algoritmom u svakoj analizi zapisuju se u relaciju *PathsDiscovered*. Putanje se sastoje od jednog ili više identifikatora pojmova u domeni znanja, zapisanih kao niz znakova. Kako bi se otkrivene putanje mogle koristiti za ostvarivanje ključne funkcionalnosti proširene

arhitekture sustava – prikaz sugestija, uz svaku putanju zapisuju se informacije koje: a) povezuju putanju s pojmom (*fk_unit*), b) određuju tip putanje (*prefix/sufix*), c) u kojem je klasteru putanja otkrivena (*fk_clustergrades*), i d) kolika je učinkovitost putanje (*profit*).



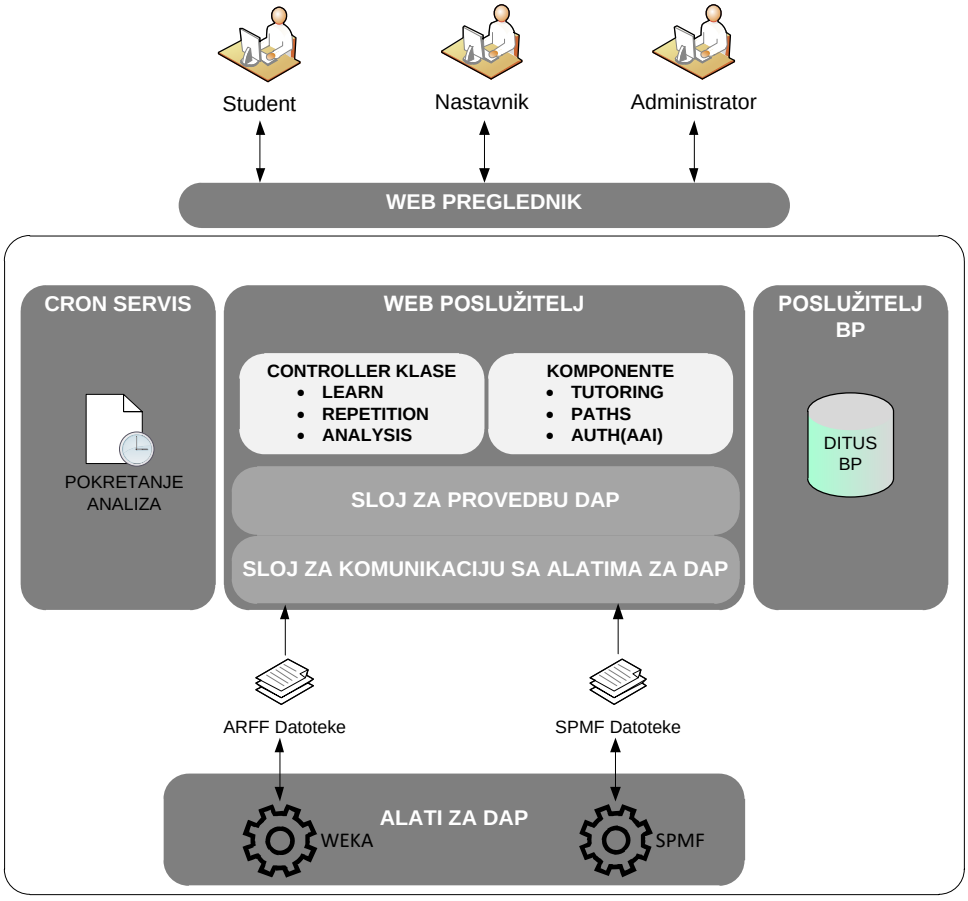
Slika 20. Dio modela baze podataka za provedene DAP

Na temelju otkrivenih čestih i učinkovitih putanja korisnika kroz domenu znanja, sustav korisnicima prezentira sugestije, s ciljem poboljšaja učinkovitosti procesa učenja, odnosno poboljšavanja cjelokupnog iskustva korištenja sustava za korisnika. U bazu podataka se pohranjuje svaka prikazana sugestija, pri čemu je ključno zabilježiti kada je sugestija praćena

(odnosno kada je korisnik prihvatio sugeriranu putanju), te zabilježiti poveznicu između dvaju aktivnosti, za potrebe naknadne analize podataka.

4.2 Arhitektura sustava

Arhitektura sustava DITUS oslanja se na alate za dubinsku analizu podataka koji su implementirani kao Java aplikacije i smješteni na istom poslužitelju (http://ditus.uniri.hr) na kojem je smješten i web poslužitelj, poslužitelj (relacijske) baze podataka, te sama web aplikacija DITUS. Kako je DITUS razvijen korištenjem skriptnog programskog jezika PHP, komunikacija s alatima za dubinsku analizu podataka provodi se otvaranjem procesa iz web aplikacije u ljusci operativnog sustava, kroz koji se šalju oblikovani pozivi putem komandne linije, te dohvaćaju rezultati. Prije slanja poziva, sloj za komunikaciju s alatima kreira potrebne datoteke na podatkovnom sustavu poslužitelja, te po potrebi dohvaća sadržaj izlaznih datoteka koje kreiraju alati. *Cron* vremenski servis se koristi za periodičku provjeru aktivnosti korisnika, te pokretanje analiza podataka. Pregled arhitekture prikazan je na slici 21.



Slika 21. Pregled arhitekture sustava DITUS

Ključne *controller* klase odnosno moduli aplikacije koji omogućuju osnovne funkcionalnosti sustava su:

- *Learn* i *Repetition* (provedba procesa Učenja i Ponavljanja), te
- *Analysis* (sučelje za nastavnika koje uz potpunu administraciju domene znanja, pruža funkcionalnosti vizualizacije podataka o aktivnostima korisnika, te mogućnost pokretanja analiza podataka).

Ključne komponente sustava su:

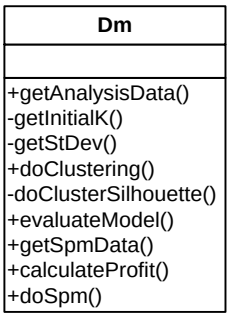
- *Tutoring* – tutorski model, pohrana podataka o aktivnostima korisnika,
- *Paths* – dohvaćanje i pohrana podataka o sugestijama, te
- *Auth* – autentikacija (prijava) korisnika putem AAI infrastrukture ili kao lokalni korisnici.

4.2.1 Implementacija podsustava za komunikaciju s alatima za DAP

Podsustav za provedbu dubinske analize i komunikaciju s alatima za dubinsku analizu podataka Weka i SPMF implementiran je u obliku dvije klase: Dm i Io. U klasi Dm definirane su metode za pripremu skupova podataka, te izvršavanje pojedinih analiza, koje se pokreću iz klase AnalaysisController koja je detaljnije opisana u sljedećem podpoglavlju. UML dijagram klase Dm prikazan je na slici 22.

Metode klase Dm implementiraju sljedeće funkcionalnosti:

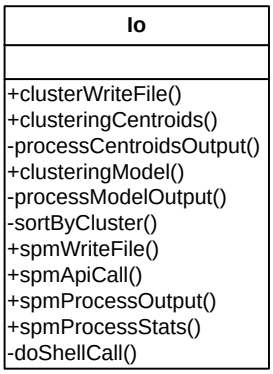
- getAnalysisData – kreiranje skupa podataka za potrebe klasteriranja korisnika
- getInitialK - odabir privremenog najvećeg broja klastera (pomoćna metoda)
- getStDev – izračun standardne devijacije (pomoćna metoda)
- doClustering – izvršenje Algoritma 1
- doClusterSilhouette – izvršenje Algoritma 2
- evaluateModel – izvršenje Algoritma 3
- getSpmData – izvršenje Algoritma 4
- calculateProfit – izračun efikasnosti putanje (pomoćna metoda)
- doSpm – izvršenje Algoritma 5



Slika 22. Klasa za provedbu analiza DAP

U klasi Io definirane su metode za kreiranje datoteka na temelju poslanih skupova podataka, izvršenje komunikacije s alatima za dubinsku analizu podataka, te inicijalnu obradu rezultata. UML dijagram klase Io prikazan je na slici 23. Metode klase Io implementiraju sljedeće funkcionalnosti:

- clusterWriteFile – zapisuje skup podataka za grupiranje korisnika na disk, kao tekstualnu datoteku oblikovanu prema potrebama alata/algoritma
- clusteringCentroids – dohvaća centroide klastera iz rezultata izvršenja algoritma poslanih iz alata za DAP na standardni izlaz procesa
- processCentroidsOutput – oblikuje podatke dobivene iz standardnog izlaza procesa u polje
- clusteringModel – kreira API poziv prema odabranom alatu, dohvaća klastering model iz rezultata izvršenja algoritma poslanih iz alata za DAP na standardni izlaz procesa
- processModelOutput - oblikuje podatke dobivene iz standardnog izlaza procesa u polje
- sortByCluster – uređivanje poretka klastera (pomoćna metoda)
- spmWriteFile - zapisuje bazu sekvenci korisnika na disk, kao tekstualnu datoteku oblikovanu prema potrebama alata/algoritma
- spmApiCall – kreira API poziv prema odabranom alatu
- spmProcessOutput - dohvaća otkrivene česte uzorke iz datoteke nastale kao rezultat izvršenja algoritma u alatu za DAP, oblikuje u polje
- spmProcessStats - oblikuje statističke podatke o rezultatima provedbe SPM analize iz datoteke nastale kao rezultat izvršenja algoritma u alatu za DAP
- doShellCall – izvršava generirani API poziv.



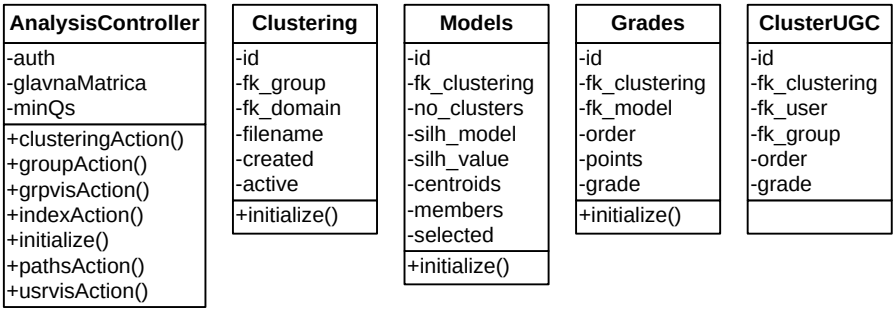
Slika 23. Klasa za komunikaciju sa alatima za DAP

Ključna metoda ovih dvaju klasa je metoda *doShellCall()*, jer omogućuje slanje API poziva iz PHP procesa prema ljusci (eng. *shell*) operativnog sustava na poslužitelju na kojem se nalazi sustav za e-učenje. Poziv se dalje izvršava na način da se pokreće jedan od dva korištena alata za dubinsku analizu podataka (Weka, SPMF). Oba alata su Java aplikacije, što znači da API poziv najprije pokreće Java platformu unutar koje se pokreće traženi alat. Na temelju argumenata ugrađenih u API poziv traženom alatu se šalje informacija o tome koji algoritam je potrebno izvršiti, nad kojim skupom podataka, s kojim osnovnim parametrima, te kako vratiti rezultate provedene analize.

Implementacijom ovog podsustava DITUS sustavu je omogućena direktna komunikacija s alatima i kreirana lako proširiva platforma za korištenje bilo kojeg algoritma ugrađenog u spomenute alate, čime DITUS postaje alat za provedbu istraživanja u području primjene dubinske analize podataka u sustavima za e-učenje.

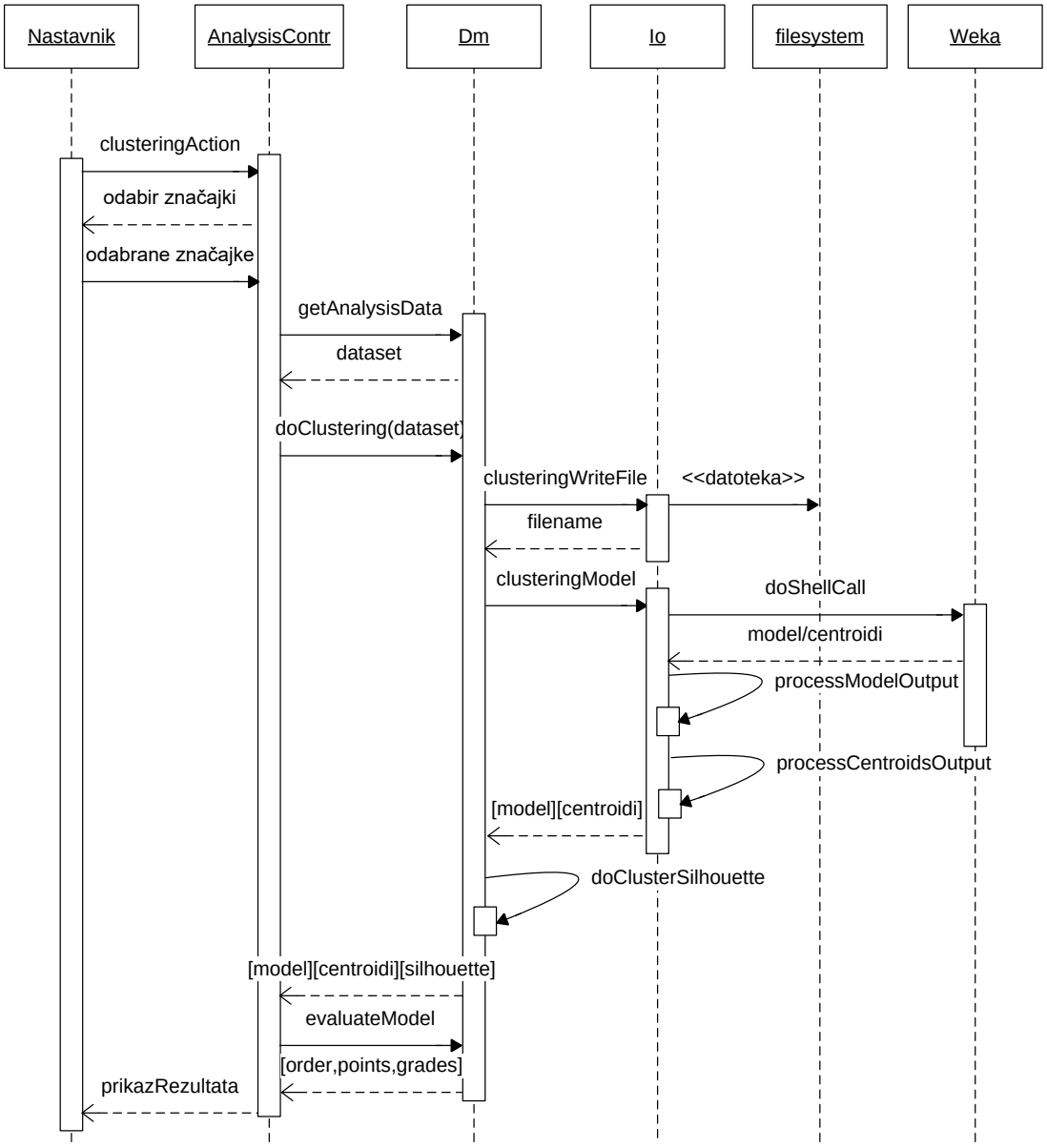
4.2.2 Implementacija podsustava za klasteriranje korisnika

Podsustav za klasteriranje korisnika koristi ranije opisani podsustav za komunikaciju s alatima za dubinsku analizu podataka. Sama analiza se pokreće putem metode *clusteringAction()* koja ujedno omogućava odabir skupa značajki koje će se ugraditi u skup podataka. Iz metode *clusteringAction()* pokreće se metoda *doClustering()* u klasi *Dm* podsustava za komunikaciju s alatima za DAP, dok pojedine metode klase *Dm* koriste više metoda klase *Io* na način koji je opisan u prethodnom podpoglavlju. UML dijagram klasa korištenih za implementaciju ovog podsustava prikazan je na slici 24.



Slika 24. Klase uključene u proces grupiranja korisnika

Model tijeka izvođenja poziva u sustavu prikazan je dijagramom sekvenci na slici 25.

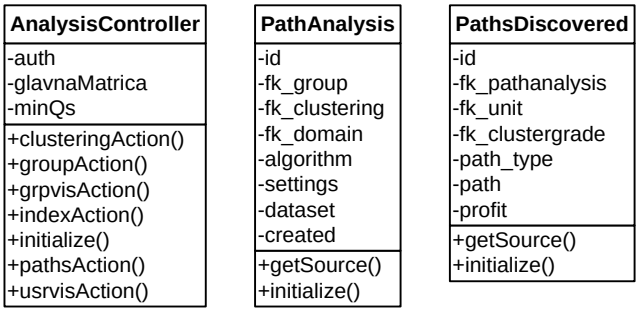


Slika 25. Dijagram sekvenci za provedbu grupiranja korisnika (Algoritmi 1, 2 i 3)

Proces klasteriranja korisnika, nastavnik započinje odabirom željenih značajki, nakon čega se kreira matrica (*dataset*) popunjena vrijednostima značajki opisanih u poglavlju 3.2.1. Matrica se zatim šalje metodi za kreiranje skupa modela klasteriranja, nakon čega se primjenom metode klase Io zapisuje na disk kao ispravno oblikovana tekstualna datoteka imenovana jedinstvenim nazivom generiranim na temelju ulaznih parametara (domena, grupa, te vrijeme pokretanja analize). Nakon toga pokreće se postupak pripreme poziva prema alatu za dubinsku analizu podataka i njegovo izvršavanje pomoću metode *doShellCall()*, nakon čega se rezultati dohvaćaju sa standardnog izlaza otvorenog procesa. Rezultati se pretvaraju u polja koja sadrže centroide klastera i model (pripadnost studenata svakom klasteru), nakon čega se vraćaju metodi *doClustering()*. Nakon kreiranja modela klasteriranja, provodi se procjena kvalitete modela metodom *doClusterSilhouette()*, čiji rezultat se dodaje dobivenom modelu klasteriranja. Opisani postupci izvode se u petlji definiranoj u Algoritmu 1. Rezultati provedene analize se prikazuju korisniku i pohranjuju u bazu podataka (podmodel prikazan u 5.1.5) posredstvom m*odel* klasa prikazanih na slici 24.

4.2.3 Implementacija podsustava za analizu čestih putanja

Podsustav za provedbu analize odnosno otkrivanja čestih i učinkovitih putanja korisnika oslanja se na podsustav za komunikaciju s alatima za DAP prikazan u poglavlju 3.1. Analiza se pokreće iz sučelja nastavnika (*AnalysisController*) pozivanjem metode *pathsAction(),* nakon čega se koriste ranije opisane metode iz klasa *Dm* i *Io*.

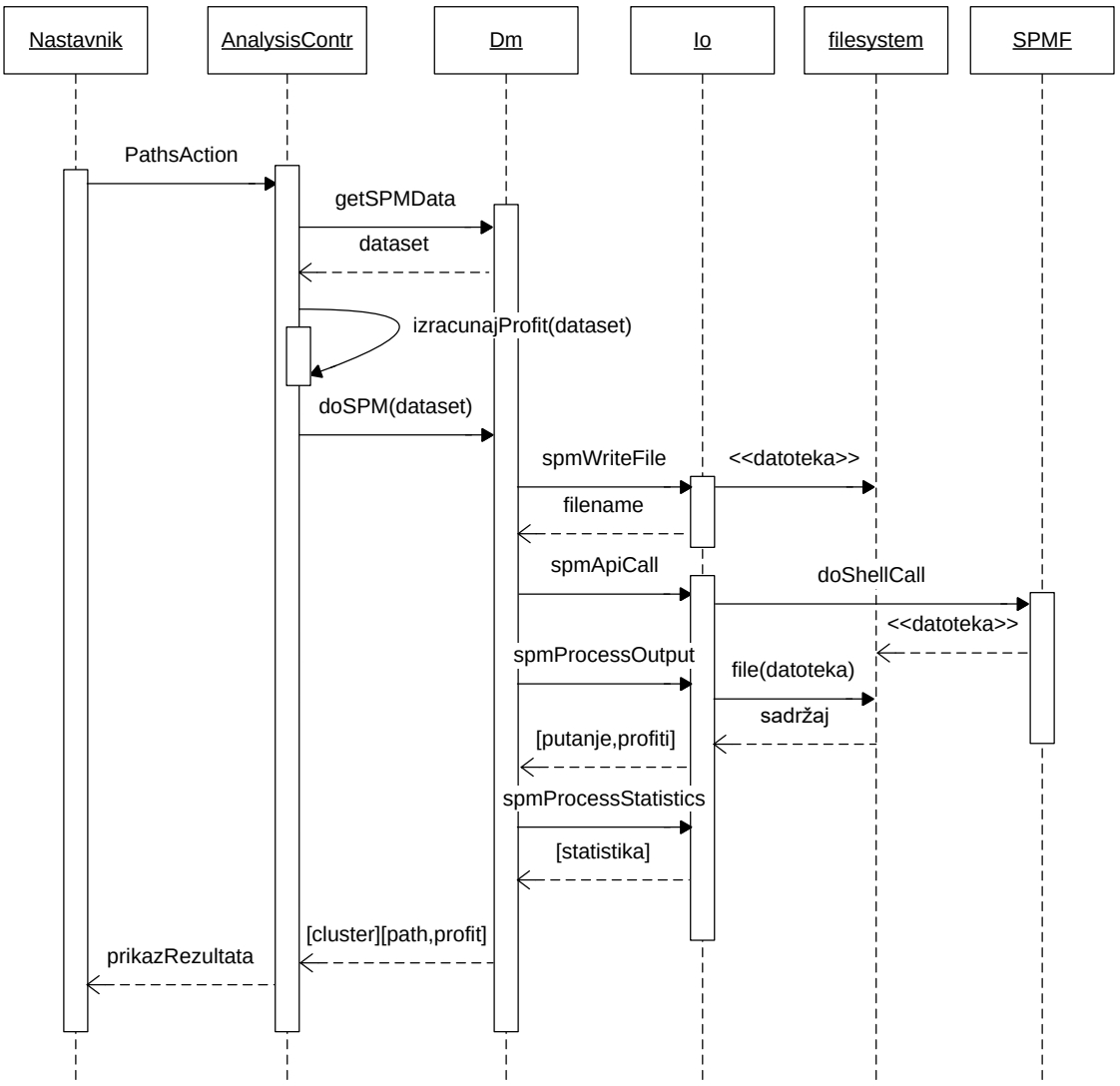


Slika 26. Klase uključene u proces otkrivanja čestih i učinkovitih putanja korisnika

Tijek izvođenja procesa otkrivanja čestih i učinkovitih putanja korisnika u sustavu prikazan je UML dijagramom sekvenci na slici 27.

Proces otkrivanja čestih i učinkovitih putanja putem komunikacije s alatima za DAP provodi se za sve klastere korisnika u trenutno aktivnom modelu. Unutar svakog klastera proces se provodi za svaki pojam u domeni znanja. Nakon pokretanja analize, kreira se skup putanja

svih korisnika u klasteru, nakog čega se svaka putanja evaluira, a potom se cijeli skup transformira u skup *prefiks* i *sufiks* putanja svakog pojma. Svaki skup se zapisuje na disk kao ispravno oblikovana tekstualna datoteka, imenovana jedinstvenim nazivom generiranim na temelju ulaznih parametara (domena, grupa, te vrijeme pokretanja analize). Nakon toga se pokreće postupak pripreme poziva prema alatu za dubinsku analizu podataka i njegovo izvršavanje.



Slika 27. Dijagram sekvenci za provedbu analize čestih i učinkovitih putanja

Kod ove analize rezultat se pohranjuje u izlaznu datoteku, nakon čega se njezin sadržaj preoblikuje u polje i vraća metodi DoSPM. Također se dohvaćaju i osnovne statistike provedene analize (trajanje, broj otkrivenih putanja). U konačnici se otkrivene putanje zapisuju u bazu podataka posredstvom model klasa prikazanih na slici 26.

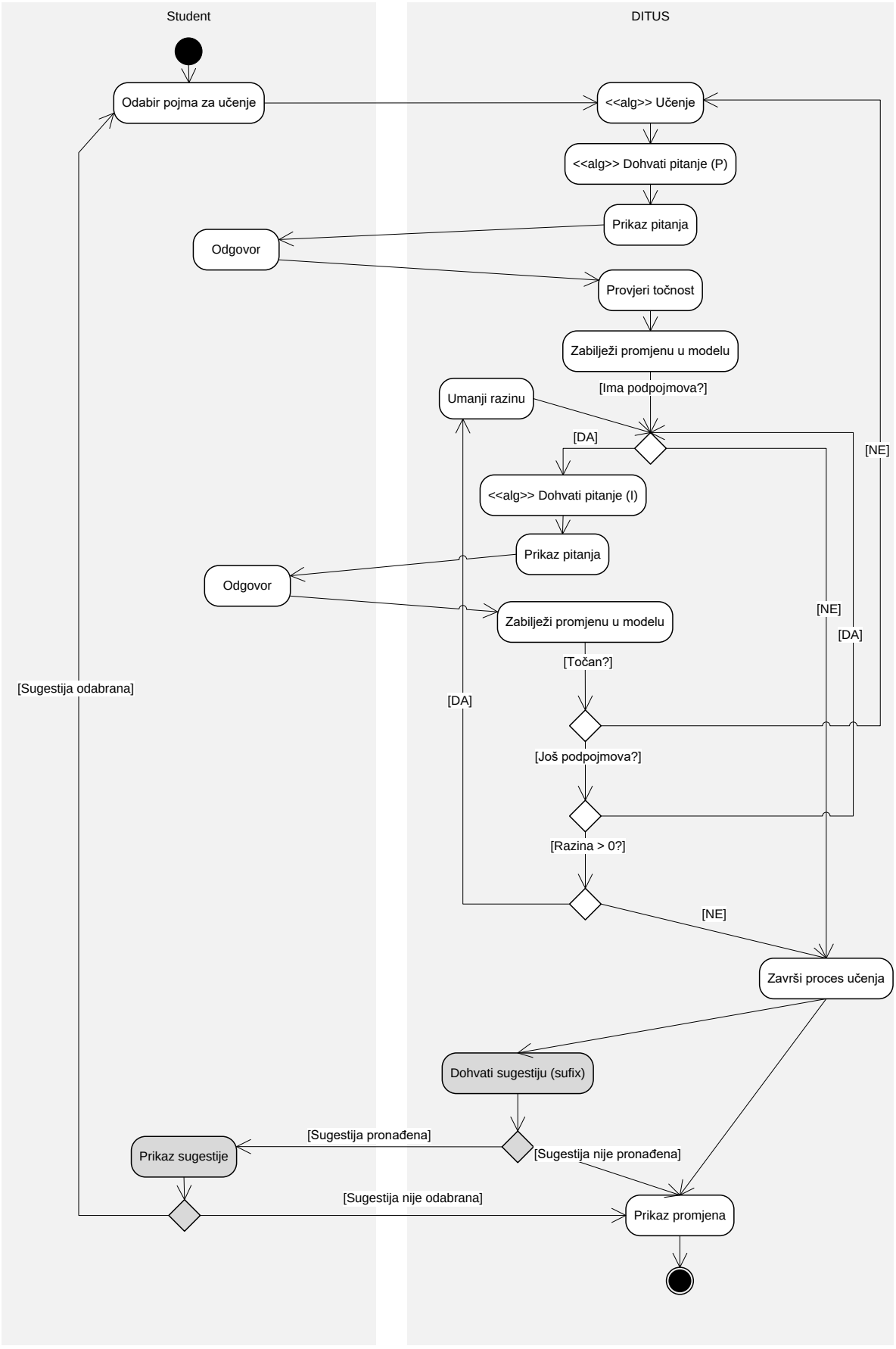
4.2.4 Implementacija poboljšanog tutorskog modela

Poboljšani tutorski model sustava oslanja se na proširenu model studenta (informacija o pripadnosti klasteru i ocjeni odnosno rangu tog klastera u odnosu na druge), te na skup otkrivenih čestih i učinkovitih putanja unutar svakog klastera, za svaki (učeni) pojam u domeni znanja. Na temelju tih informacija, proizašlih iz dubinske analize objektivnih podataka o aktivnostima učenja korisnika sustava, poboljšani tutorski sustav može kod početka i kod završetka svake aktivnosti Učenja, odnosno kod završetka svake aktivnosti Ponavljanja ponuditi korisniku sugestije koje imaju za cilj pomoći studentu u savladavanju domene znanja. Kako bi se sugestije implementirale u postojeći tutorski model, bilo je potrebno izmjeniti tutorski model sustava na način koji je prikazan sljedećim UML dijagramima.

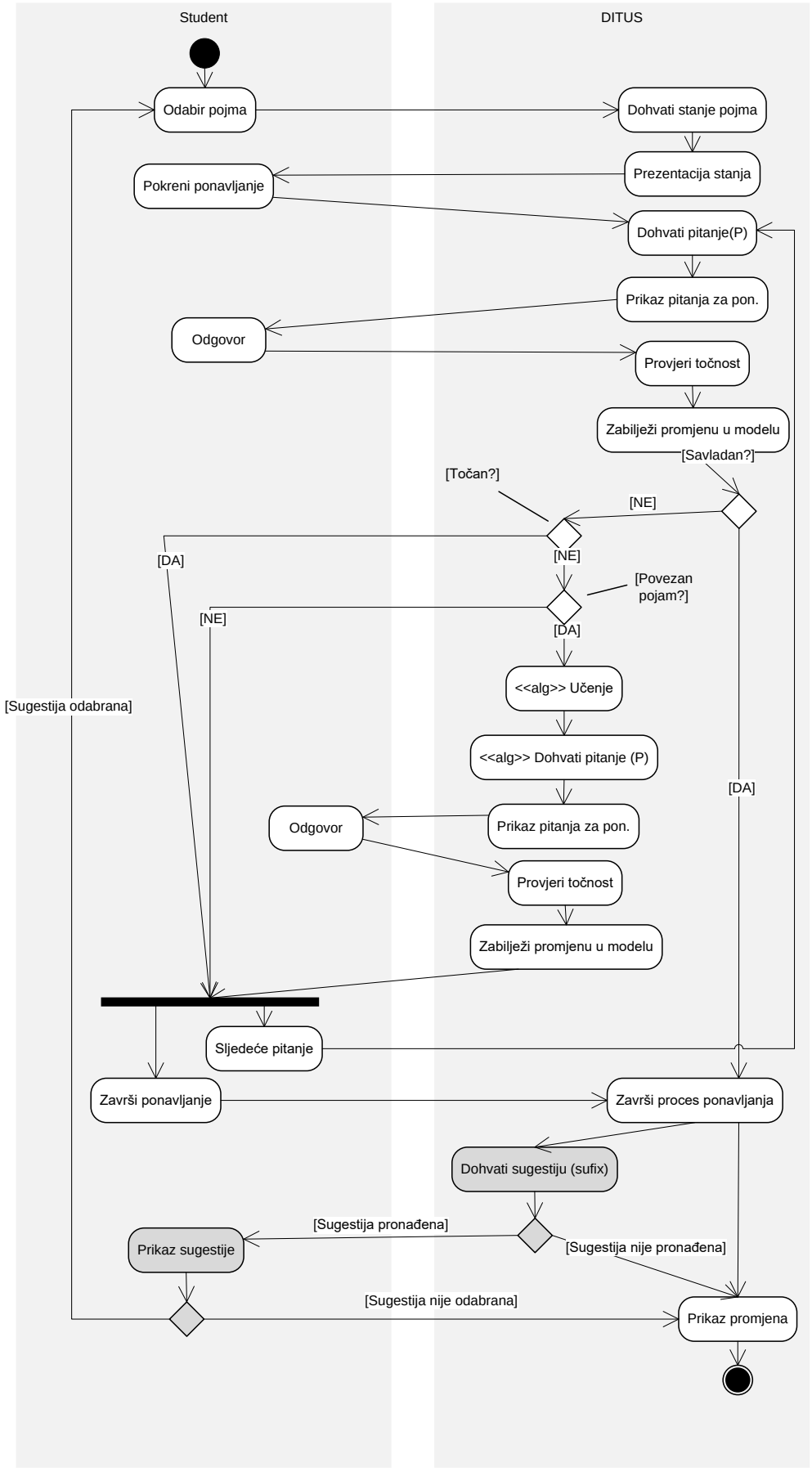
UML dijagram aktivnosti cjelokupnog procesa UČENJE prikazan je na slici 28. Po završetku procesa korisniku se prikazuju promjene nastale u modelu studenta na temelju odgovora na postavljena pitanja. Kako bi se omogućio prikaz sugestija tipa SUFIX, koje korisniku sugeriraju sljedeću aktivnost (učenje ili ponavljanje, trenutnog ili nekog drugog pojma, ovisno o odabranoj sugestiji), model je proširen akcijom odabira sugestije (izvršenje Algoritma 6). Ukoliko se algoritmom pronađe sugestija, ona se prikazuje korisniku. Ukoliko korisnik odabere ponuđenu aktivnost, proces se pokreće ponovno.

U modelu se pojavljuje akcija s oznakom „X“ koja označava skup akcija obojanih sivom bojom, odnosno skup akcija za dohvaćanje i prikaz sugestije, te promjenu odabranog pojma u slučaju odabira sugestije.. Svaki put kada proces završava i prikazuje promjene u modelu studenta izršava se postupak odabira/prikaza sugestije, a korisnik odabire da li će završiti proces ili odabrati ponuđenu sugestiju. Isti postupak se ponavlja i kod procesa Ponavljanja (Slika 29).

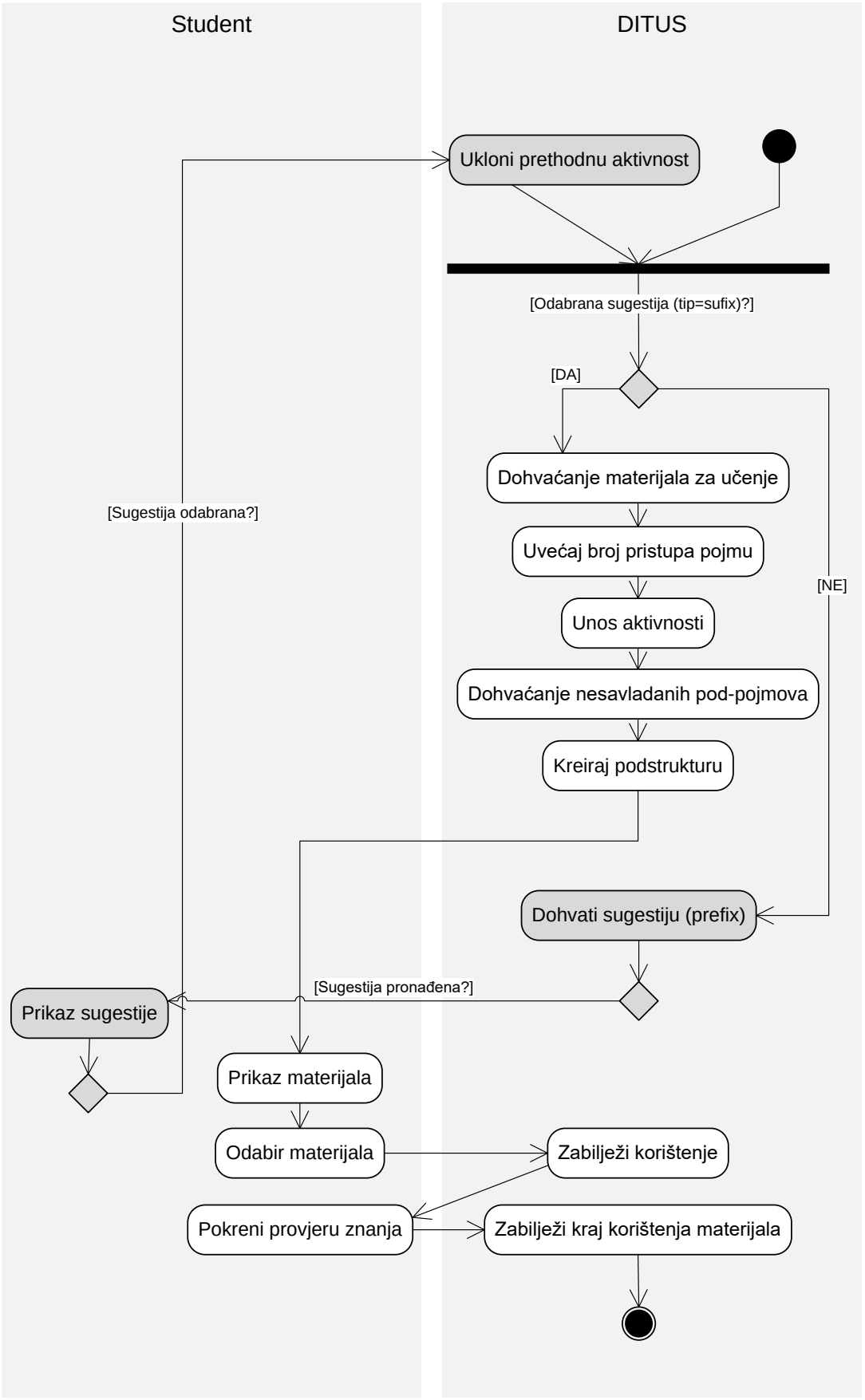
Prikaz sugestija tipa PREFIX ugrađen je u akciju koja je na modelu prikazana kao **<<alg>>Učenje**. Model tog algoritma je prikazan na slici 30.



Slika 28. UML dijagram aktivnosti procesa „Učenje“



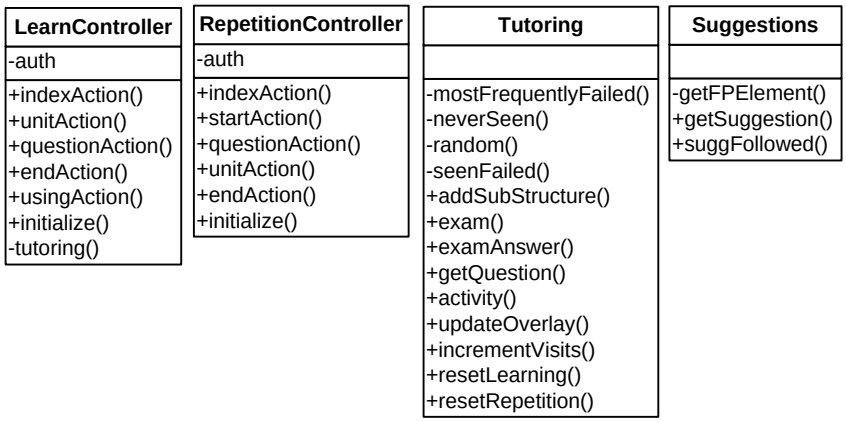
Slika 29. UML dijagram aktivnosti procesa „Ponavljanje“



Slika 30. UML dijagram aktivnosti algoritma „Učenje“

Model algoritma „Učenje“ prikazuje proširenja osnovnog modela, kojima je omogućen prikaz sugestija tipa PREFIX. Sugestije se ne prikazuju u slučaju pokretanja algoritma u sklopu procesa PONAVLJANJE. Kod prikaza nastavnih materijala, prikazuje se i sugestija koja korisniku predlaže učenje nekog drugog pojma, prije trenutno prikazanog, s ciljem poboljšanja učinkovitosti procesa učenja. U slučaju da korisnik odabere ponuđenu sugestiju sustav će ukloniti određene zapise o prethodnoj aktivnosti, kako bi skup podataka o aktivnostima korisnika odražavao ispravljenu putanju korisnika, te na taj način dopridonosio budućim analizama putanja.

Spomenuti procesi implementirani su u dvije ključne *Controller* klase – *LearnController i RepetitionController*. Klasa *Tutoring* je pomoćna klasa (nije dio MVC arhitekture) koja omogućava pohranu podataka o aktivnostim korisnika, ažuriranje modela studenta, ali i osigurava ispravan rad sustava. Poboljšani tutorski model koristi još jednu pomoćnu klasu, klasu *Suggestions,* koja implementira Algoritam 6 (vidi poglavlje 3.4)., te pohranjuje podatke o praćenim sugestijama u bazu podataka. UML dijagram spomenutih klasa prikazan je na slici 31.

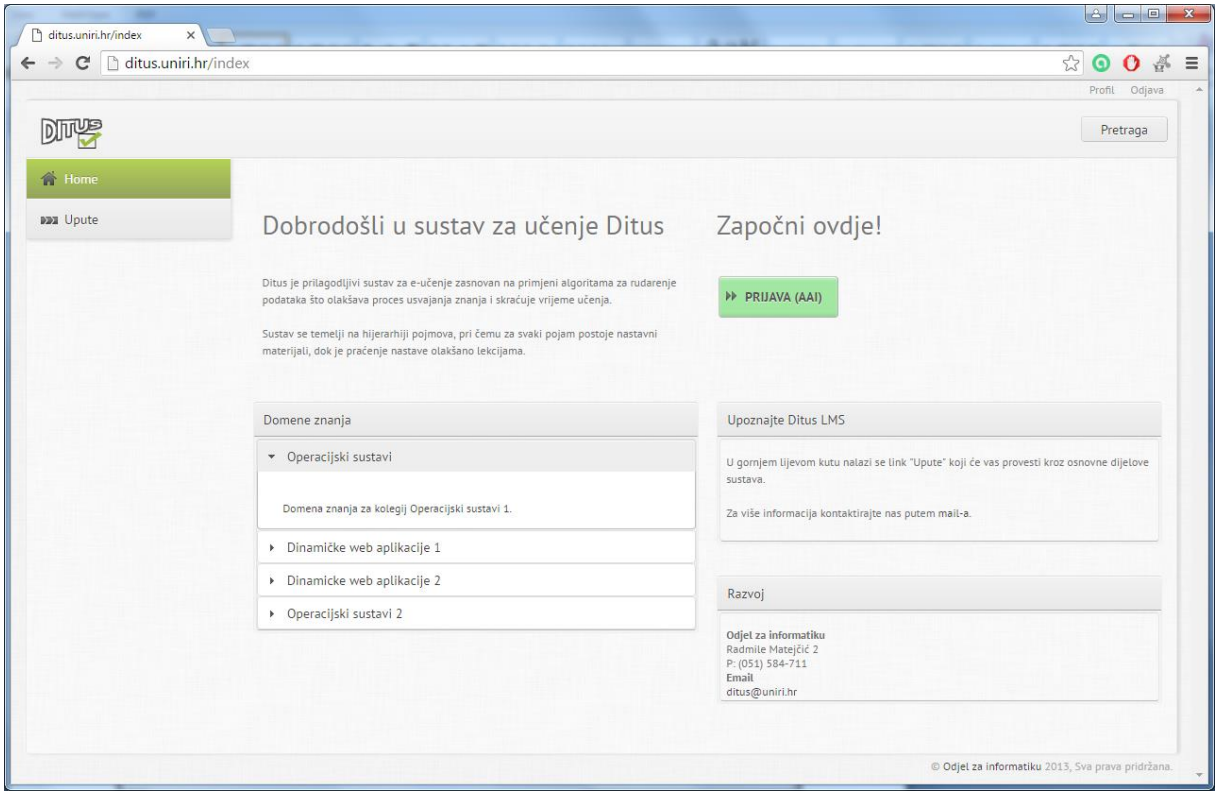


Slika 31. Klase uključene u provedbu procesa Učenja i Ponavljanja, te odabira sugestija

4.3 Sučelja sustava

Sučelja sustava odvojena su od razina manipulacije podacima i logike primjenom Model-View-Controller uzorka. Cjelokupni predložak sučelja sustava razvijen je primjenom HTML5, CSS3 i JavaScript tehnologija kako bi se automatski prilagođavao dimenzijama ekrana korisnikovog uređaja. Redoslijed prikaza sučelja prati dinamiku rada svih kategorija

korisnika sa sustavom, odnosno cjelokupni proces od kreiranja domene znanja, do korištenja sustava od strane studenata i analize podataka od strane profesora.

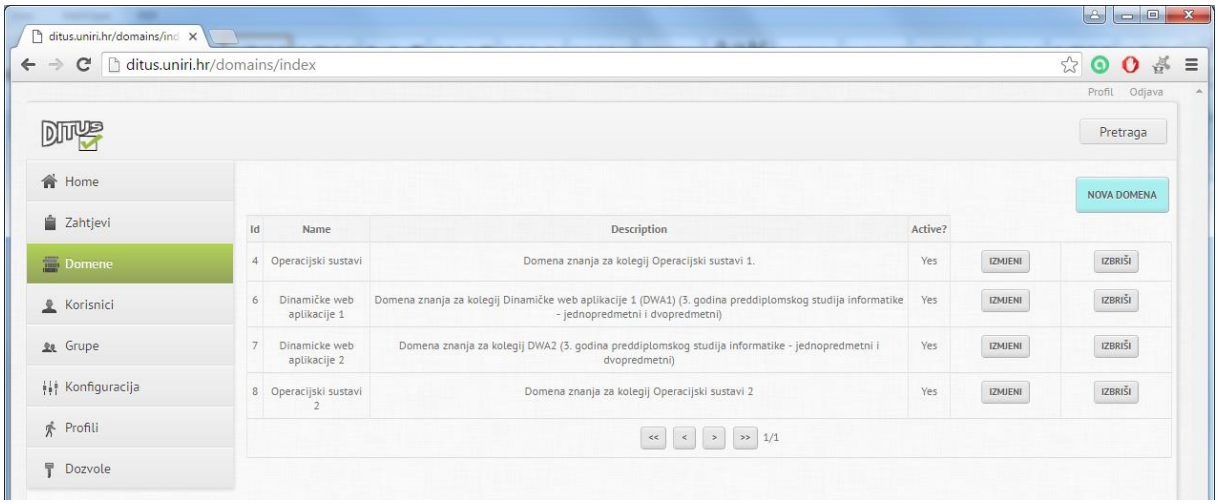


Slika 32. Početna stranica DITUS sustava

Nakon dolaska na početnu stranicu aplikacije DITUS (Slika 32) korisnik se mora prijaviti u sustav – pri čemu je moguća prijava putem AAI sustava ili putem lokalne autentikacije.

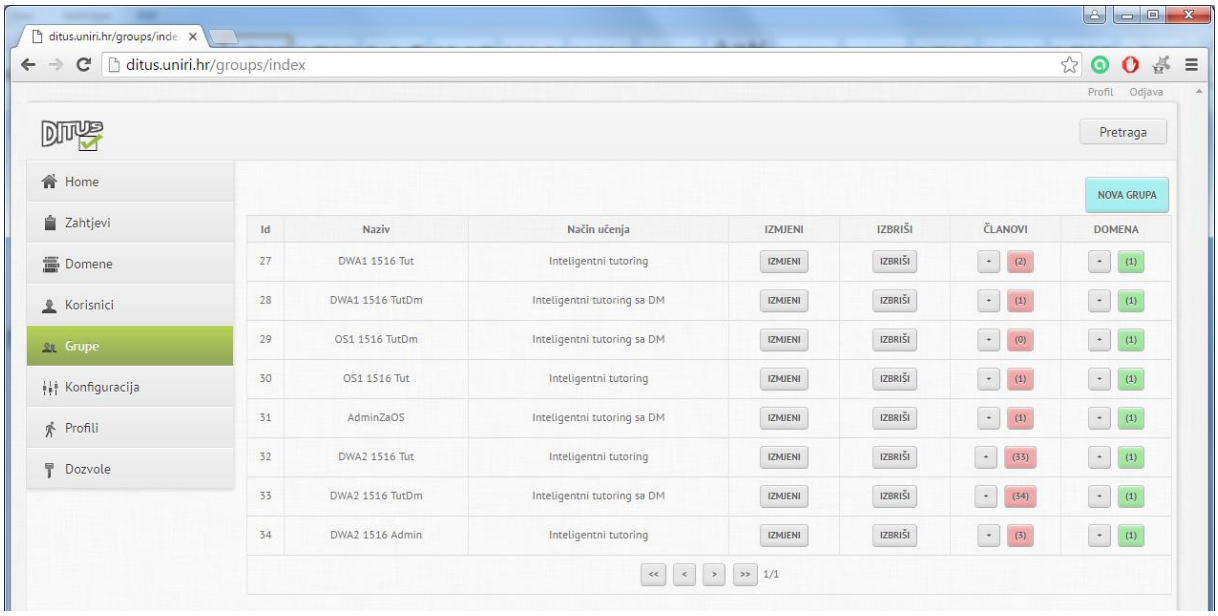
4.3.1 Sučelja administratora sustava

Razvoj domene znanja započinje administrator sustava kreiranjem novog zapisa o domeni. Domenu je potrebno učiniti dostupnom kao preduvjet za daljnje korake. Pregled domena u sustavu, s poveznicama na akcije kreiranja, izmjene i brisanja domena prikazan je na slici 33.



Slika 33. Sučelje za rad s domenama znanja

Kako bi profesor mogao pristupiti kreiranju pojmova, pripremi nastavnih materijala i pitanja, a studenti koristiti domenu, administrator kreira grupe korisnika, pri čemu se za svaku grupu definira način učenja, nakon čega se grupa povezuje s jednom ili više domena, te jednim ili više korisnika.

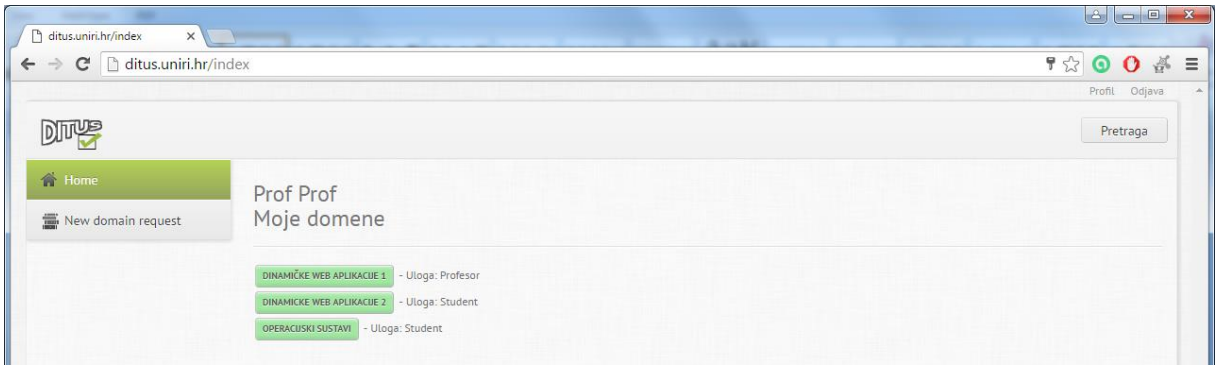


Slika 34. Sučelje za organizaciju grupa, domena i studenata

Sučelje za pregled grupa, te akcijama za kreiranje, izmjenu, brisanje, dodavanje i brisanje korisnika grupe, te dodavanje i brisanje domena grupe prikazano je na slici 34.

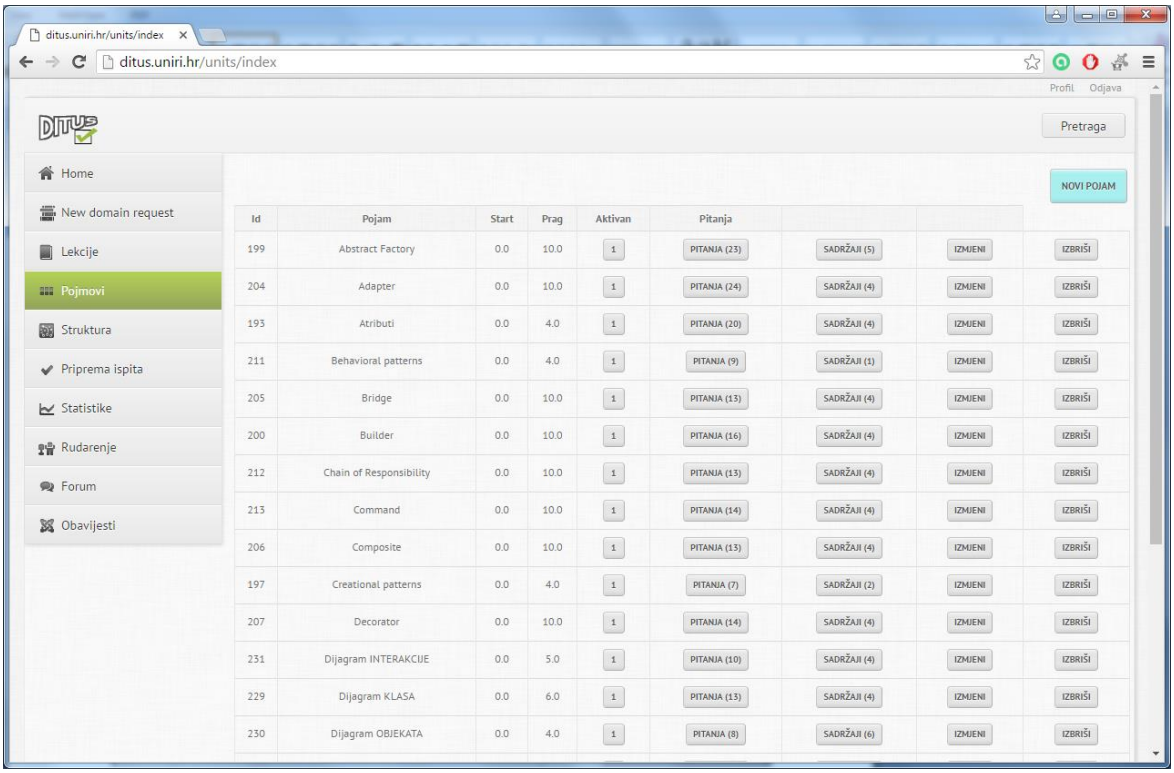
4.3.2 Sučelja za popunjavanje domene znanja

Nakon prijave, profesor odabire domenu znanja u kojoj želi raditi. Sučelje za odabir domene znanja prikazano je na slici 35.



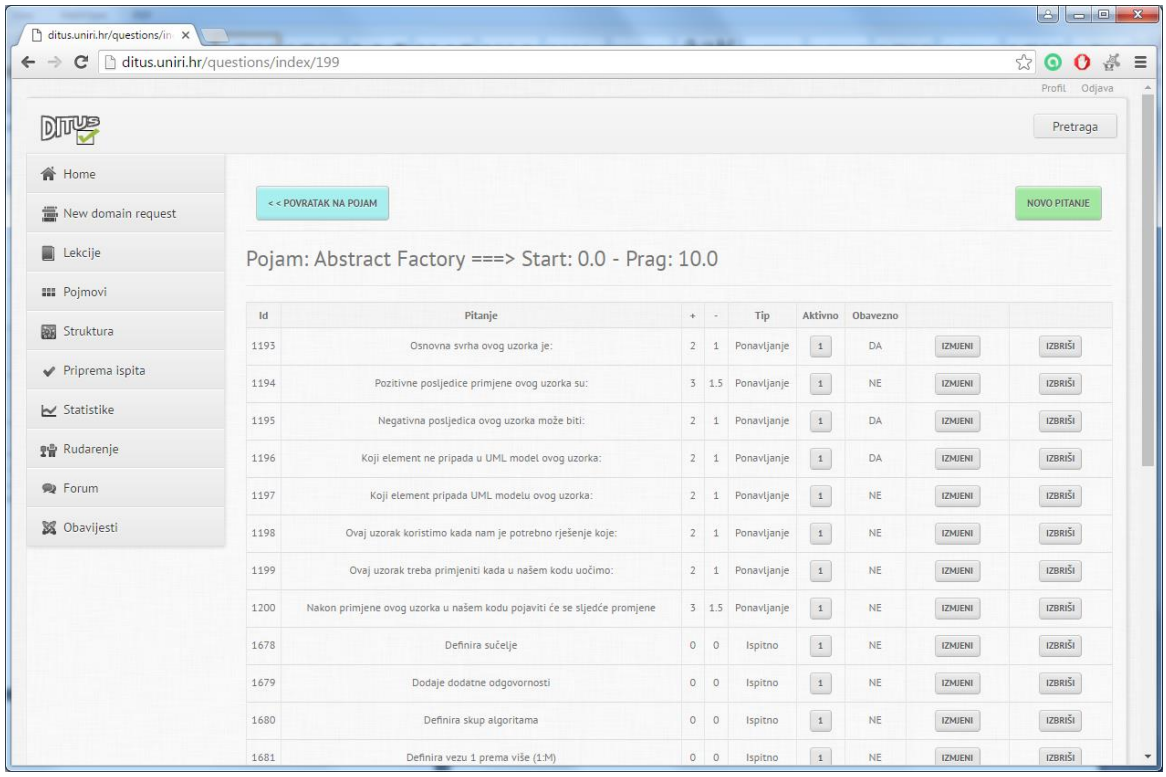
Slika 35. Početni ekran profesora nakon prijave – odabir domene

Po odabiru domene, profesor može odabrati prikaz popisa pojmova s informacijama o postavljenom pragu savladavanja pojma, te poveznicama na sučelja za dodavanje novog pojma, unos pitanja odnosno nastavnih materijala za pojedini pojam, kao što je prikazano na slici 36.



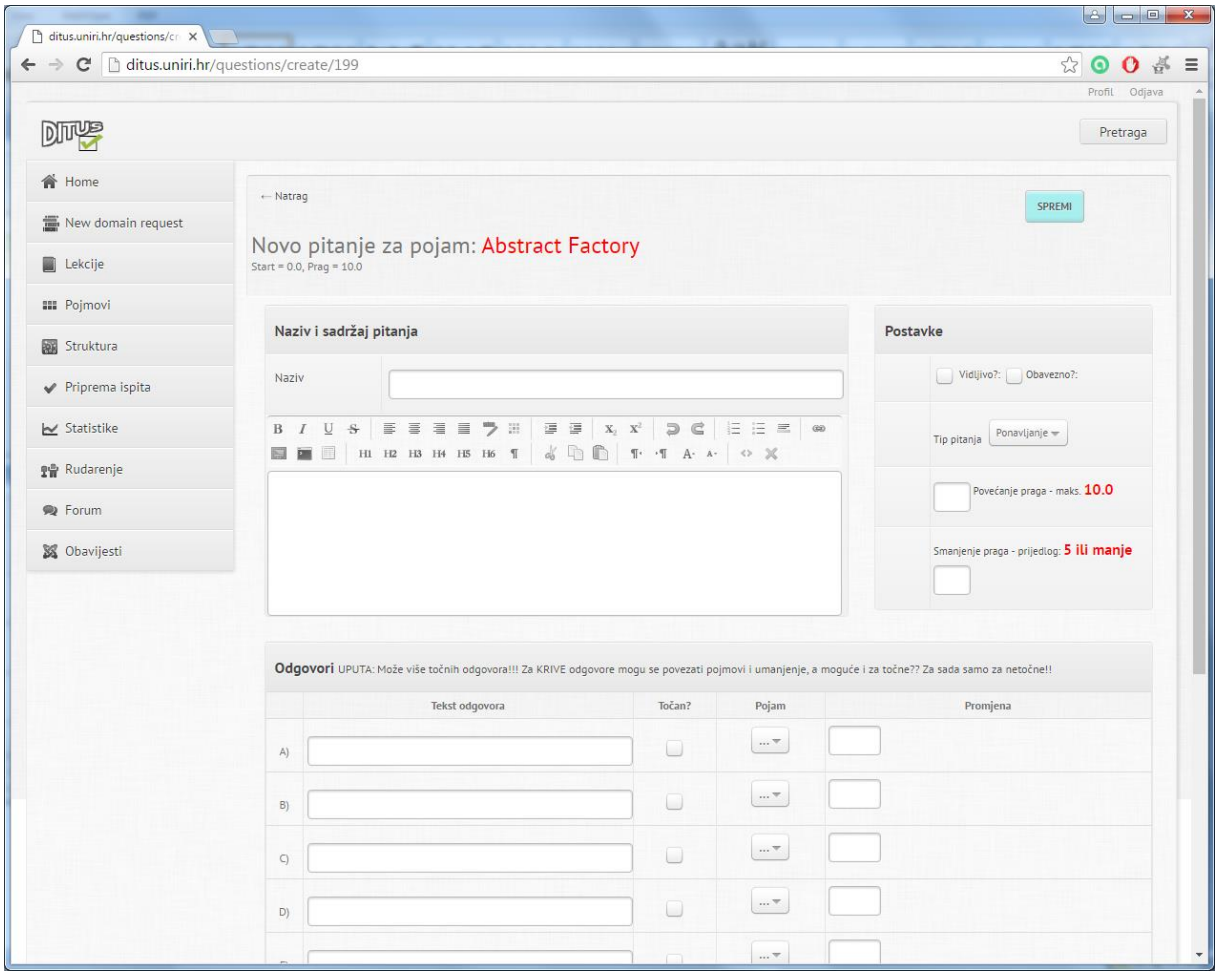
Slika 36. Pregled pojmova s akcijama unosa pitanja, sadržaja, izmjenu i brisanje pojma

Nakon odabira akcije za pregled unesenih pitanja za pojam prikazuje se sučelje koje omogućuje izmjenu i brisanje pitanja, promjenu vidljivosti/dostupnosti pitanja, te unos novih pitanja, što je vidljivo na slici 37.



Slika 37. Prikaz pitanja za odabrani pojam

Sučelje za unos novog pitanja, prikazano na slici 38, omogućuje unos sadržaja pitanja, odabir vrste pitanja, unos iznosa promjene u modelu (znanja) studenta, unos odgovora uz određivanje točnih odgovora, uz mogućnost povezivanja neispravnih odgovora s nekim od pojmova iz domene znanja. Ukoliko se netočan odgovor poveže s pojmom, moguće je unijeti iznos (negativne) promjene u modelu (znanja) studenta. Ova poveznica omogućuje dio funkcionalnosti algoritma za ponavljanje, koji će u tom slučaju prikazati materijale za učenje za povezani pojam, provjeriti znanje jednim pitanjem, te se vratiti na provjeru znanja onog pojma kojeg je korisnik inicijalno odabrao.

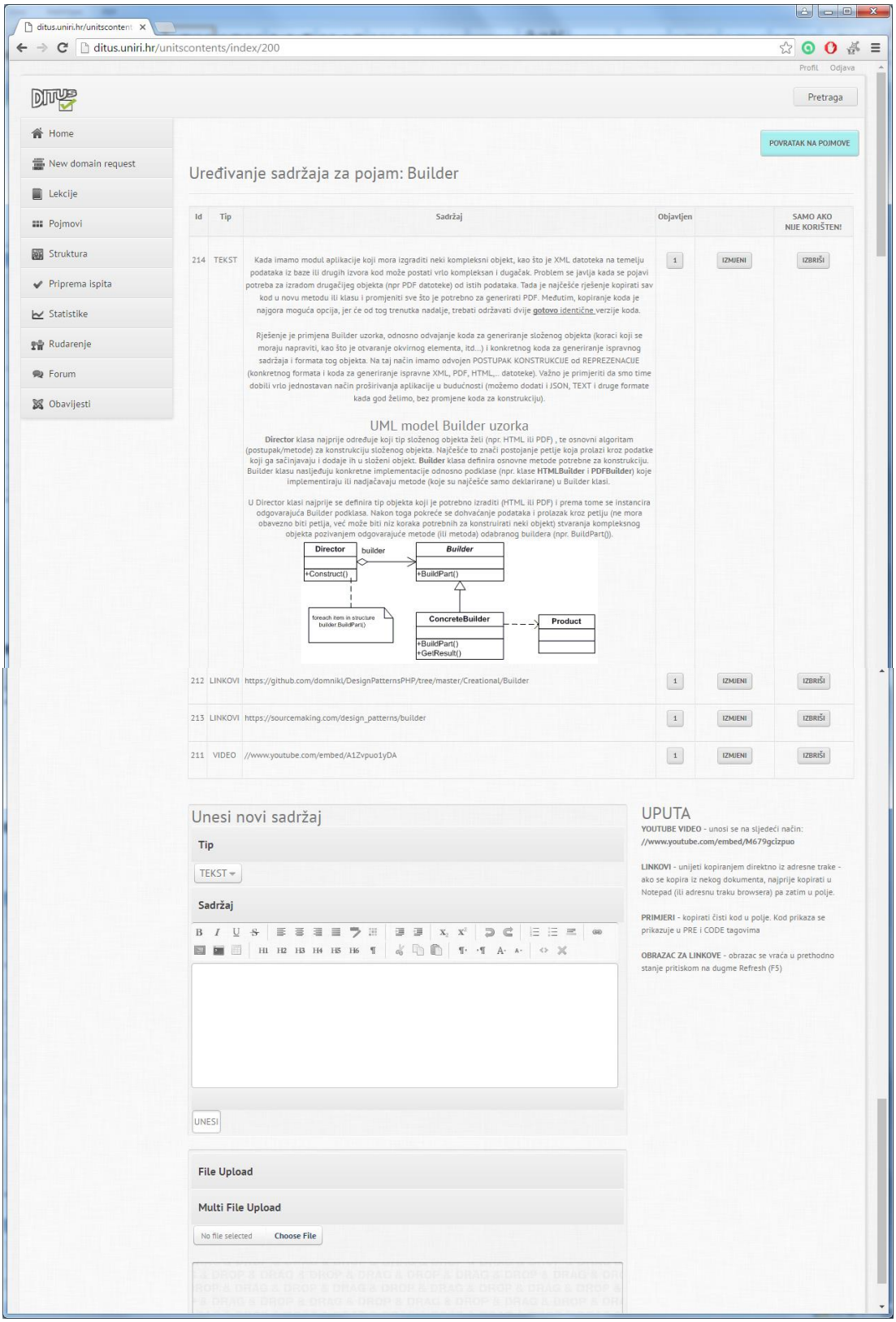


Slika 38. Sučelje za unos pitanja za odabrani pojam

Jedna od ključnih funkcionalnosti sustava je unos nastavnih materijala za odabrani pojam. Sučelje za unos sadržaja prikazano je na slici 39. Omogućen je unos četiri vrste sadržaja:

- HTML sadržaji (tekst, slike, tablice, hiperveze, itd.)
- primjeri kôda
- poveznice na druga web sjedišta ili sadržaje
- ugradnja video materijala s Youtube web sjedišta

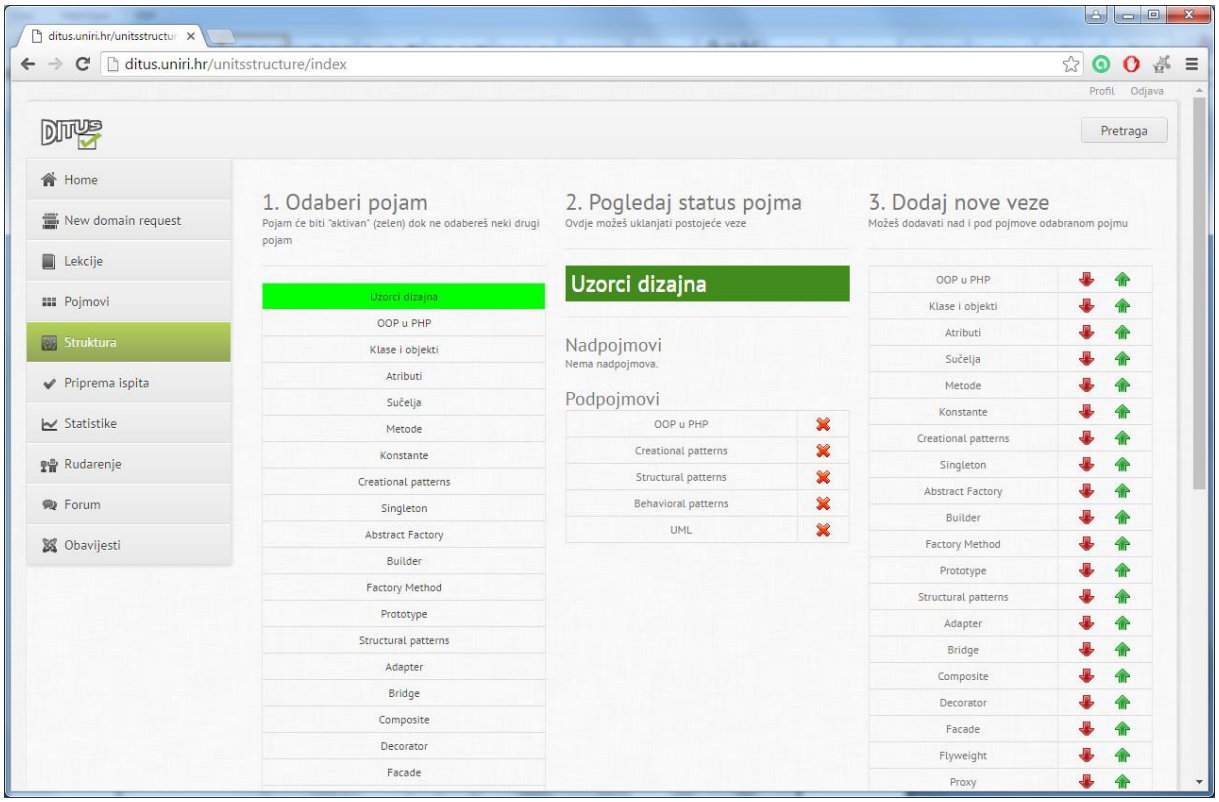
Za svaki sadržaj moguće je definirati vidljivost/dostupnost.



Slika 39. Sučelje za rad s nastavnim materijalima

Nakon kreiranja pojmova, unosa pripadajućih pitanja i nastavnih materijala, profesor kreira strukturu pojmova definiranjem poveznica tipa „nadpojam“ odnosno „podpojam“ za svaki

odabrani pojam. Ovakav način kreiranja strukture zahtjeva vizualnu pripremu strukture od strane profesora prije započinjanja postupka. Sučelje je prikazano na slici 40.



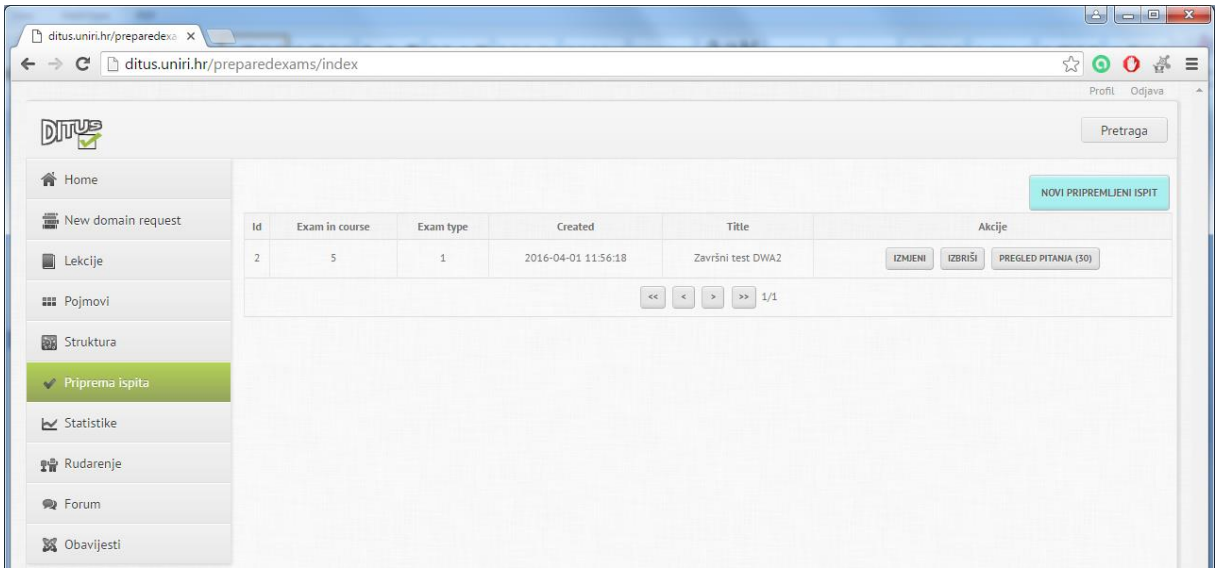
Slika 40. Sučelje za izradu strukture pojmova

U sljedećim iteracijama razvoja sustava razviti će se vizualno sučelje (primjenom vektorske grafike u web pregledniku, odnosno biblioteke D3JS) čime će se značajno olakšati kreiranje strukture, kao i olakšati naknadne izmjene. Profesor također može kreirati ispite, te u njih dodati (ispitna) pitanja, te odrediti kojoj će grupi i u kojem vremenskom intervalu ispit biti dostupan. Sučelje za pregled unesenih pripremeljenih ispita, s akcijom za dodavanje pitanja u ispit prikazano je na slici 41.

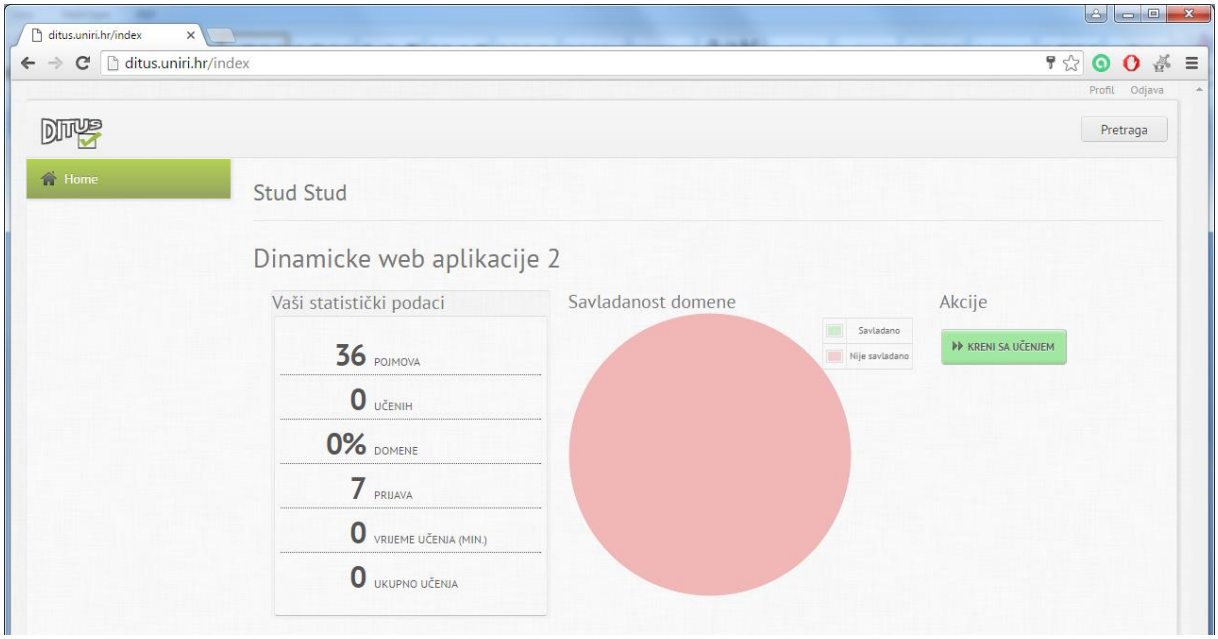
Nakon što je profesor popunio domenu znanja pojmovima, dodao sve nastavne materijale, te unio inicijalna pitanja, pitanja za provjeru i ispitna pitanja, te pripemio test(ove) znanja, studenti mogu započeti s korištenjem sustava.

4.3.3 Sučelja za provedbu procesa učenja

Nakon postupka prijave, studentu se prikazuje pregled svih domena znanja kojima ima pristup, uz osnovne statističke pokazatelje, te vizualizaciju savladanosti domene. Sučelje pregleda domena prikazano je na slici 42.



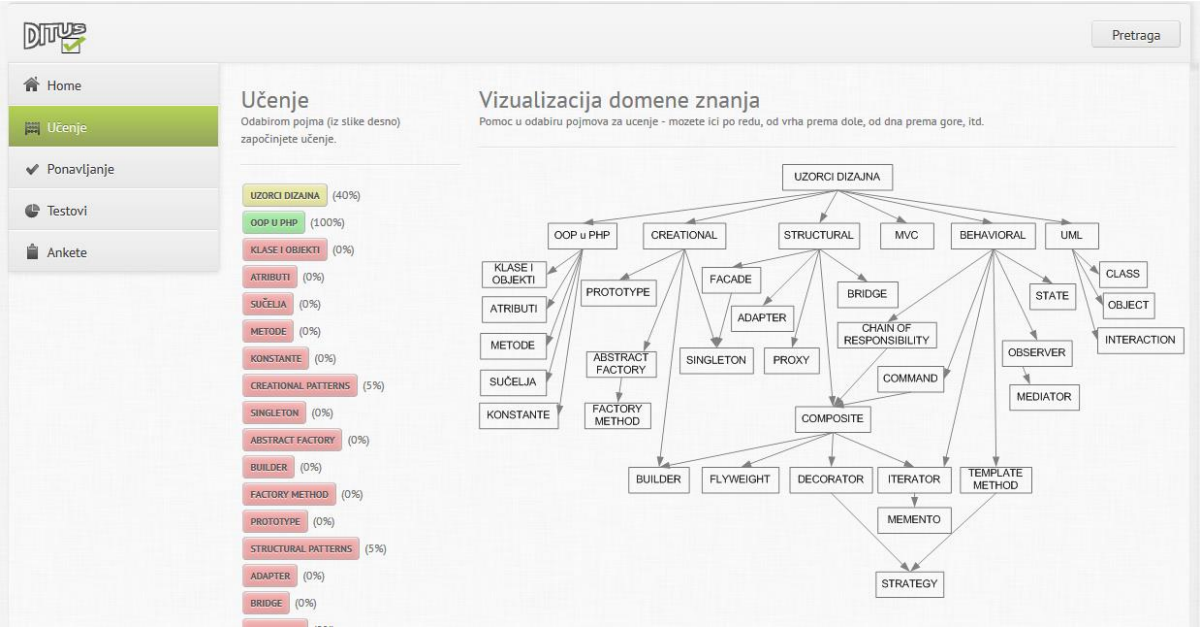
Slika 41. Sučelje za kreiranje ispita i odabir pitanja za ispit



Slika 42. Pregled domena nakon prijave studenta

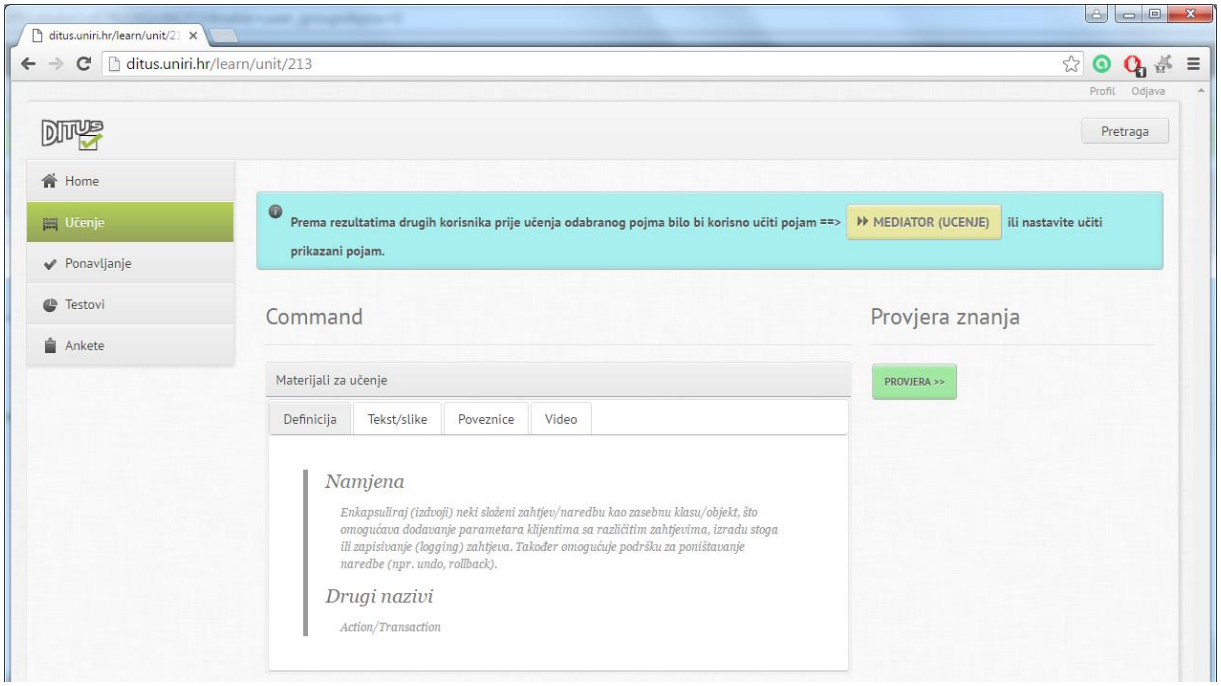
Po odabiru domene, odnosno početka procesa učenja, studentu se prikazuje popis mogućih akcija – dvije ključne funkcionalnosti sustava (učenje, ponavljanje), te dodatne funkcionalnosti (pristup testu znanja i ispunjavanje ankete). Popis pojmova u domeni daje informaciju o savladanosti svakog pojma (uz vizualno kodiranje – pojam je prikazan crvenom bojom do 20% savladanosti, u intervalu 20%<x<99% prikazan je žutom bojom, dok su savladani pojmovi prikazani zelenom bojom. Pojam koji želi učiti student odabire iz

vizualizacije strukture domene na desnoj strani sučelja. Početno sučelje funkcionalnosti učenja prikazano je na slici 43.



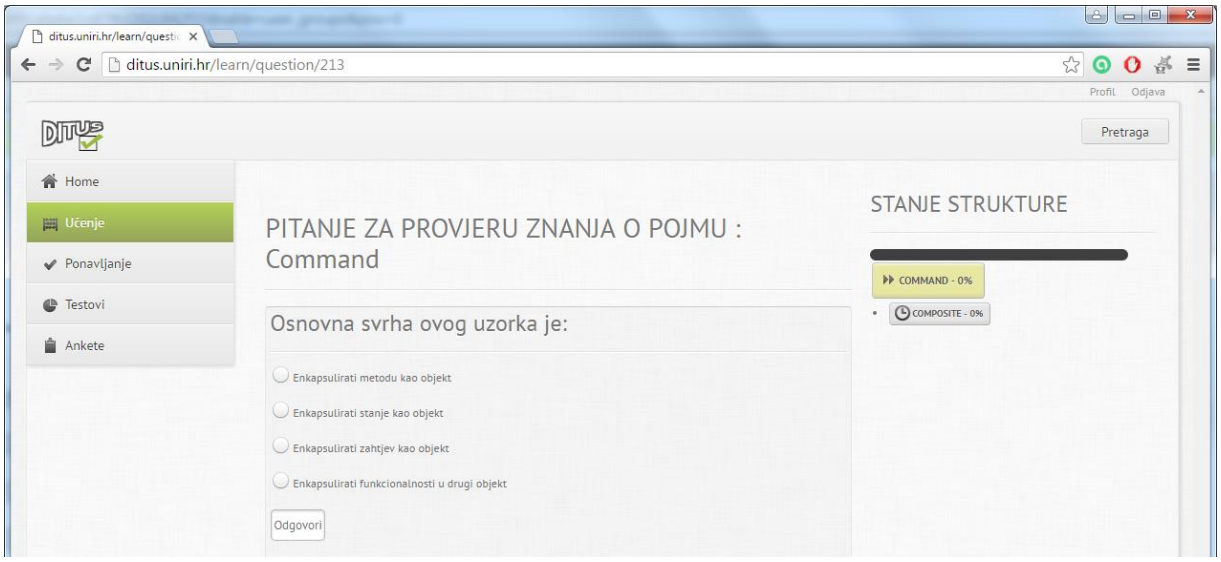
Slika 43. Početna stranica funkcionalnosti učenja

Po odabiru pojma za učenje, pokreće se algoritam dijaloga, koji započinje prikazom nastavnih materijala (prikazano na slici 44). Ovisno o dostupnosti, sustav na ovom koraku prikazuje sugestije tipa „prefix“ odnosno sugerira odabir drugog pojma, za kojeg se dubinskom analizom podataka i interpretacijom rezultata utvrdilo da ga je bolje učiti prije učenja trenutno odabranog pojma.



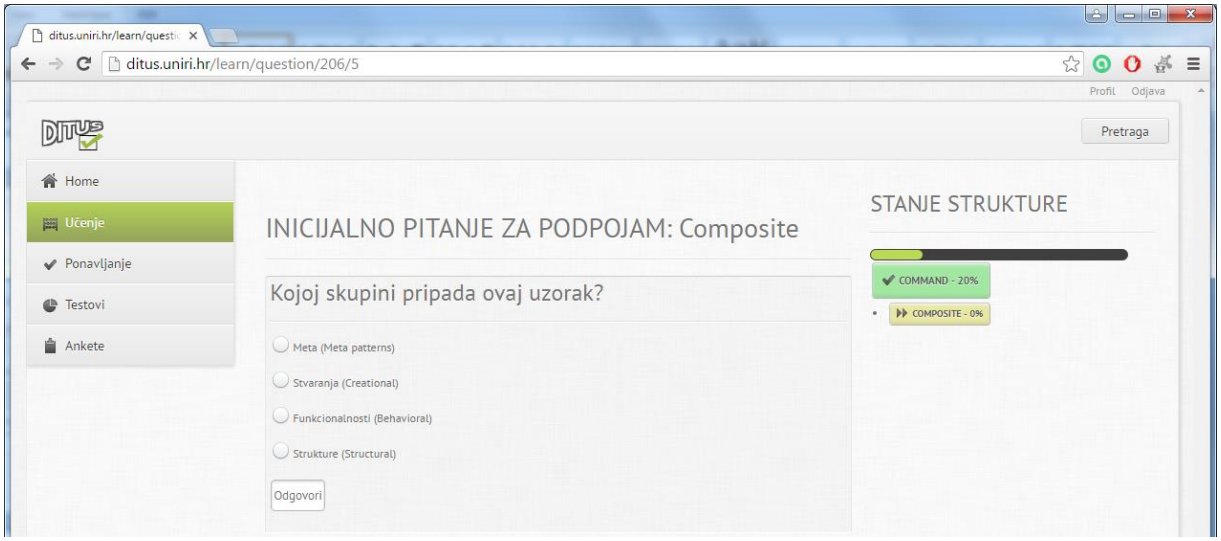
Slika 44. Prikaz materijala za učenje i sugestije

Na ovaj način, sustav pokušava pomoći studentu da pojmove savladava redoslijedom koji se pokazao učinkovitim za druge studente. Nakon analize nastavnih materijala korisnik pristupa provjeri znanja. Algoritam dijaloga postavlja jedno pitanje za provjeru znanja o odabranom pojmu, te kreira podstrukturu povezanih pojmova, koje student u tom trenutku nije savladao (savladanost je <100%). Sučelje za provjeru znanja prikazano je na slici 45.



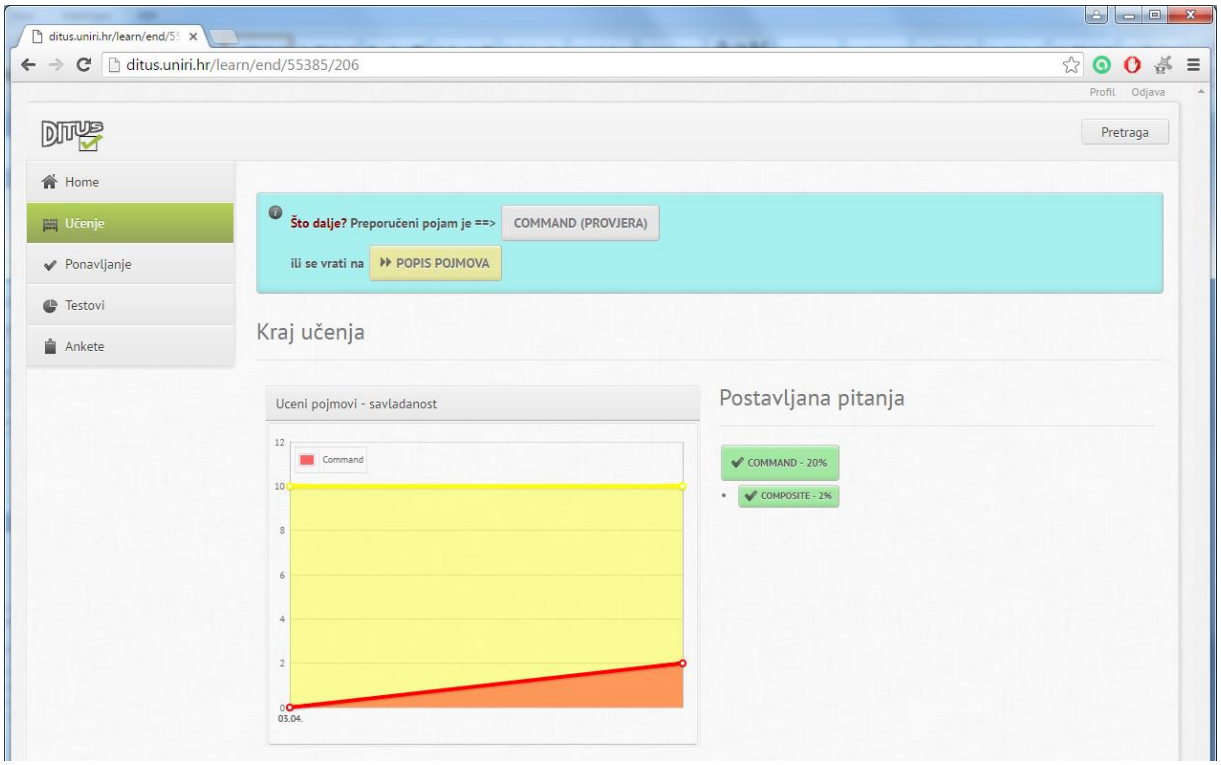
Slika 45. Provjera znanja nakon prikaza materijala za učenje

Nakon postavljanja pitanja za provjeru znanja, sustav će postaviti incijalno pitanje za svaki od pojmova postavljenih u strukturu (Slika 46). Netočan odgovor na inicijalno pitanje, uzrokovati će ponovno pokretanje algoritma dijaloga, prema modelu opisanom u poglavlju 2.3.



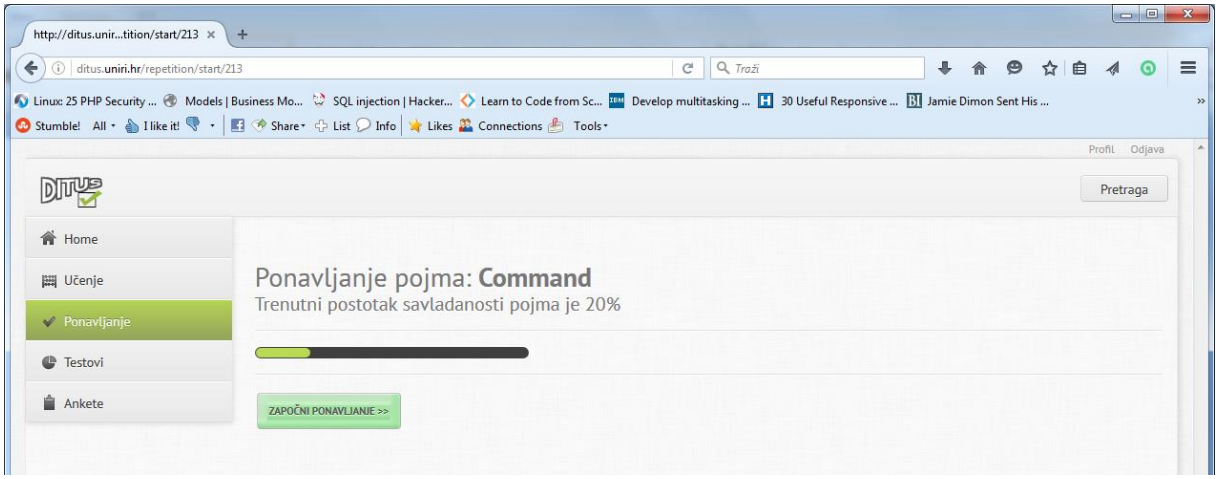
Slika 46. Incijalno pitanje za sve podpojmove u strukturi učenja

Po završetku algoritma dijaloga, sustav prezentira rezultate i promjene u modelu (znanja) studenta. Na ovom ekranu moguć je, ovisno o dostupnosti, prikaz sugestija tipa „sufix“, odnosno sugestija studentu o tome koji pojam učiti sljedeći. Na slici 47. prikazan je završetak procesa učenja, s prikazom sugestije.



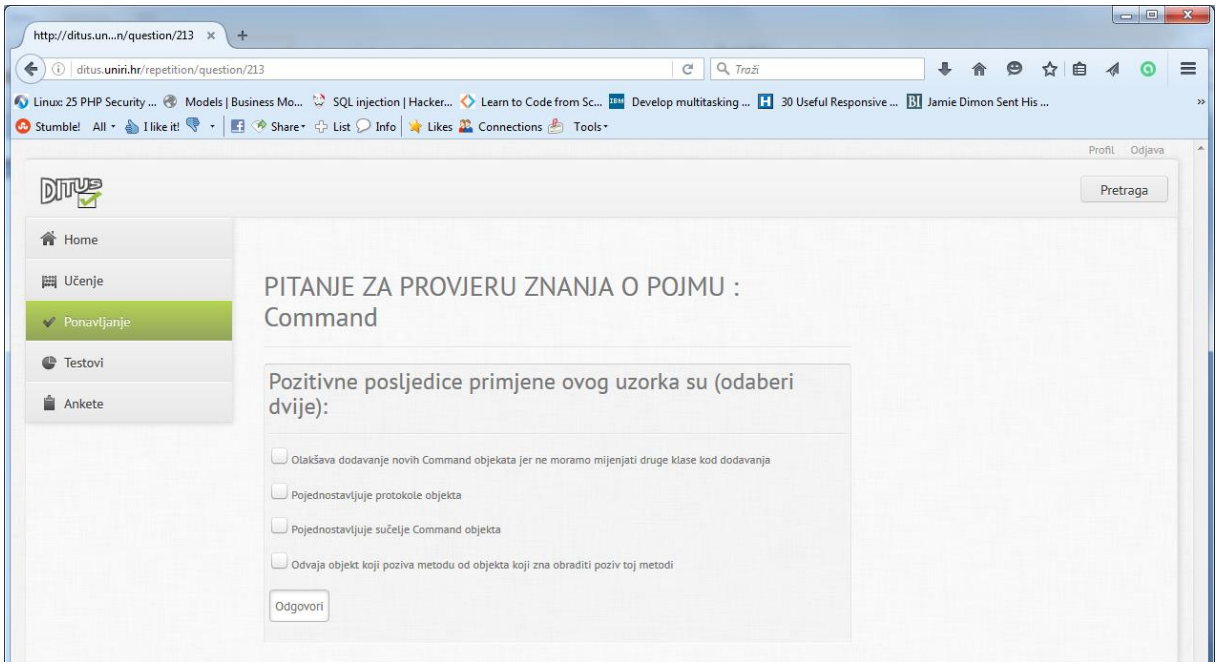
Slika 47. Završni ekran učenja s prikazanom sugestijom

Funkcionalnost ponavljanja započinje odabirom pojma, kao što je prikazano na slici 48, nakon čega student dobiva informaciju o trenutnom stanju savladanosti pojma, te započinje proces postavljanja pitanja.



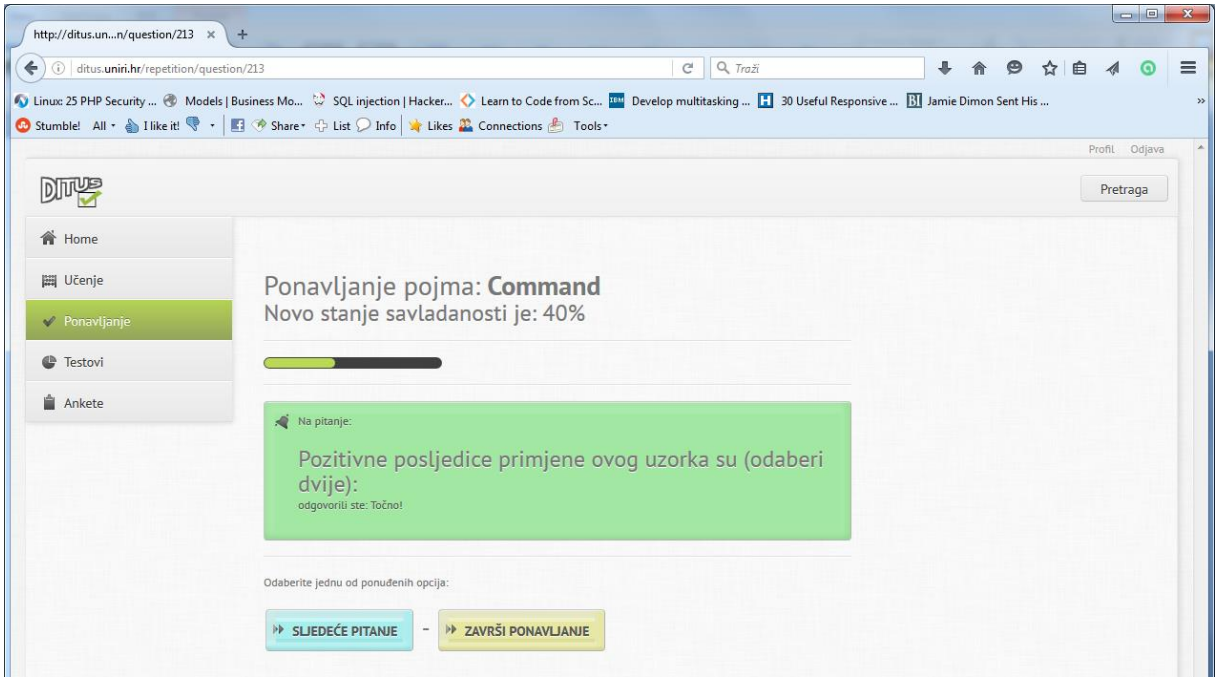
Slika 48. Početak procesa ponavljanja nakon odabira pojma

Time se pokreće algoritam za provjeru znanja, koji dohvaća pitanje za provjeru i prikazuje ga studentu (prikazano na slici 49).



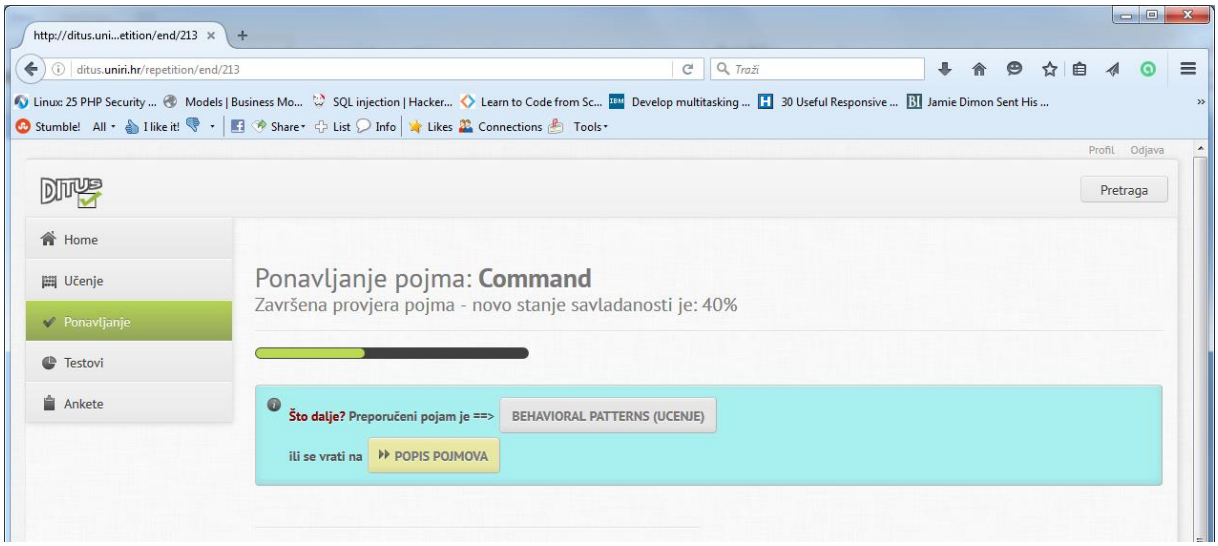
Slika 49. Prikaz pitanja u procesu ponavljanja

Nakon slanja odgovora, student dobiva informaciju od ispravnosti odgovora na pitanje. Ukoliko je student točnim odgovorom dosegnuo 100% savladanosti pojma, proces provjere će završiti, prikazom završnog sučelja (Slika 50). U suprotnom, student će imati dvije mogućnosti: nastavak ili završetak provjere.



Slika 50. Prikaz rezultata odgovora na pitanje u procesu ponavljanja

Po završetku provjere, student dobiva informaciju o promjeni u modelu (znanja) studenta. Na ovom ekranu moguć je, ovisno o dostupnosti, prikaz sugestija tipa „sufix“, odnosno sugestija studentu o tome koji pojam učiti sljedeći.

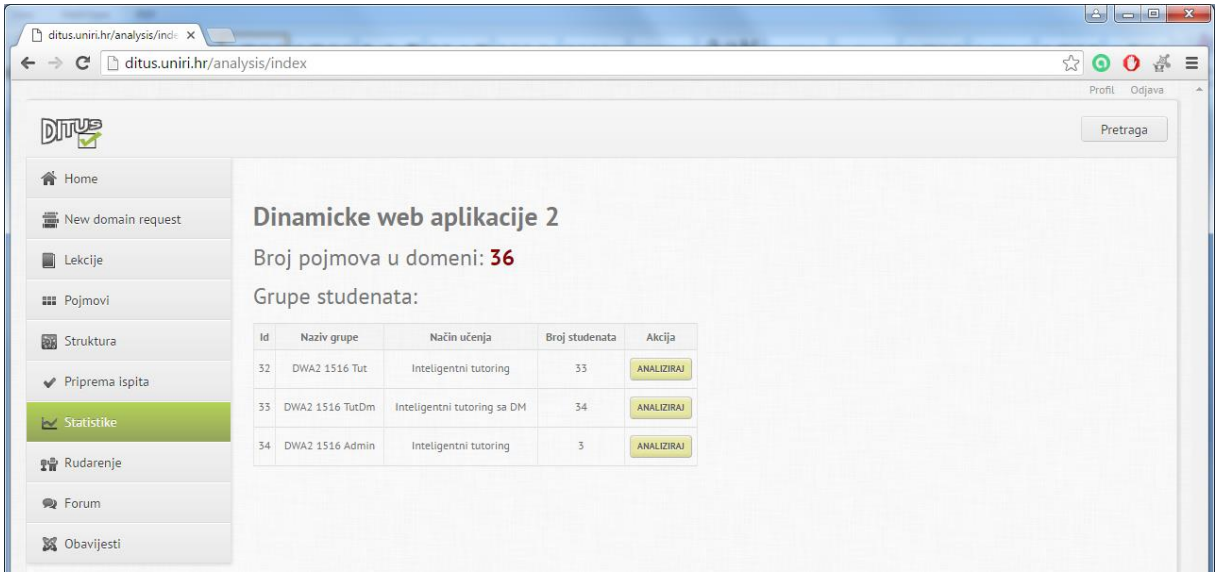


Slika 51. Završni ekran procesa ponavljanja s prikazanom sugestijom

Završetak procesa ponavljanja prikazan je na slici 51.

4.3.4 Sučelja za analizu podataka o učenju

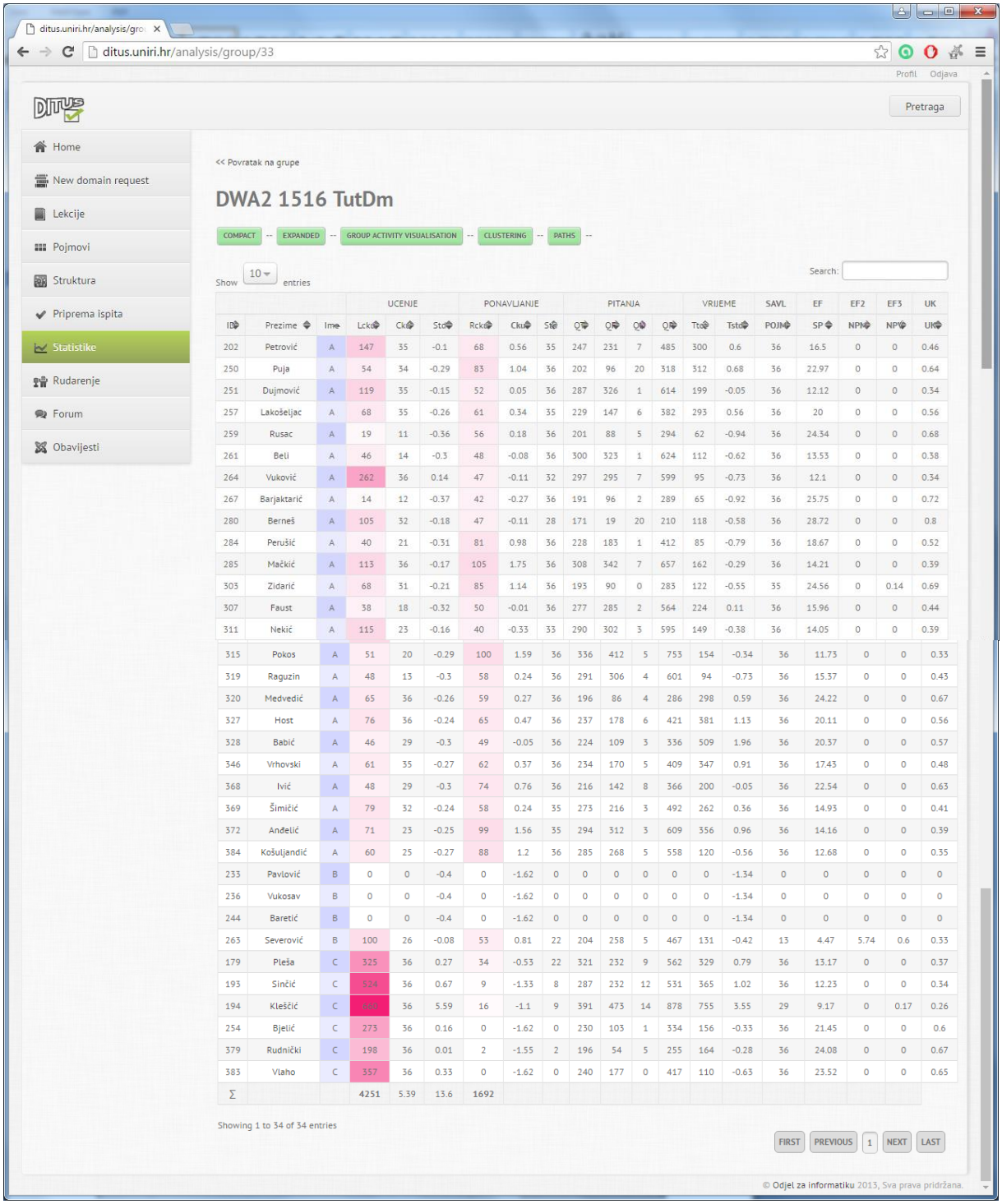
Dok studenti koriste sustav ili nakon završetka perioda predviđenog za savladavanje domene znanja, profesor može koristiti modul za analitiku učenja kako bi dobio bolji uvid u aktivnosti studenata. Po odabiru modula za analitiku, profesor odabire grupu studenata čije zapise o učenju želi analizirati (Slika 52).



Slika 52. Početni ekran modula za analitiku učenja i rad s modulima za DAP

Početna stranica modula za analitiku koja daje popis svih studenata s pregledom ključnih značajki aktivnosti (broja aktivnosti učenja, ponavljanja, statistike postavljenih pitanja, vremena učenja, efikasnosti) prikazana je na slici 53. Radi lakšeg snalaženja, tablica s podacima može se sortirati po svim stupcima, te filtrirati unosom željene vrijednosti u polje u

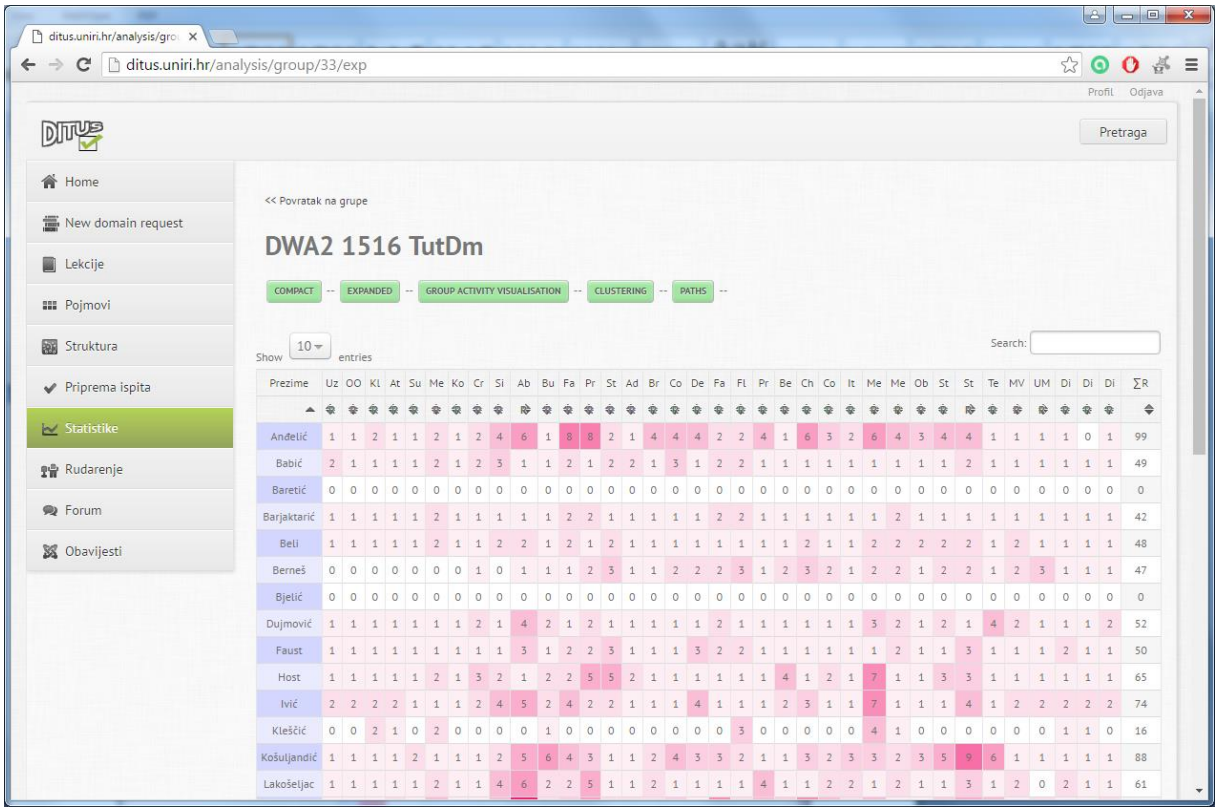
gornjem desnom kutu. Radi lakšeg uočavanja distribucije vrijednosti koristi se metoda vizualizacije „intenzitetom topline“ (eng. *heat map*) odnosno promjenjivi intenzitet boje pozadine, kao što je prikazano na slici 53.



Slika 53. Pregled podataka o učenju odabrane grupe

Odabirom proširenog pregleda, profesor dobiva pregled aktivnosti učenja ili ponavljanja članova grupe. Intenzitet boja olakšava uočavanje pojmova kojeg su studenti najviše učili, ali i uočavanje najaktivnijih studenata. Tablicu je moguće sortirati po svakom pojmu (stupcu

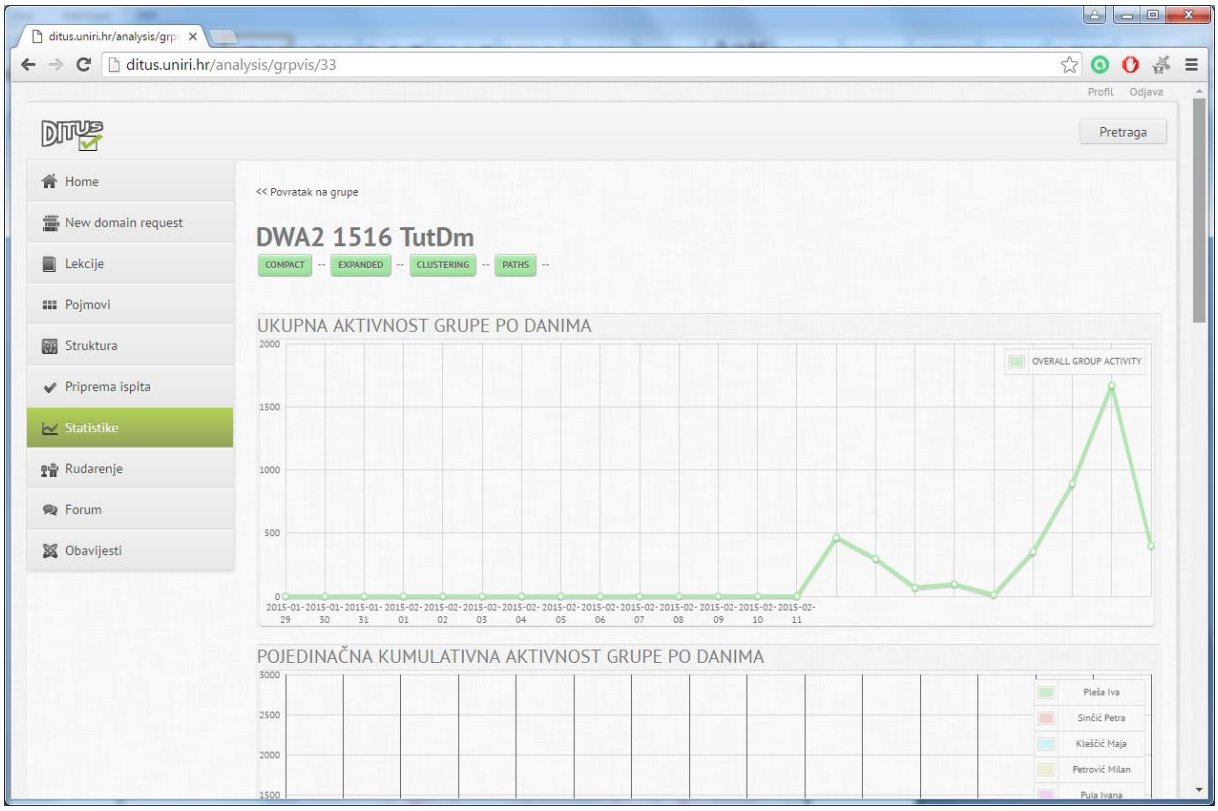
tablice), čime se olakšava traženje informacija. Tablica prikazana na slici 54, poredana je po prezimenima studenata. Stupci tablice su skraćeni nazivi pojmova (puni naziv je vidljiv nakon postavljanja pokazivača na skraćeni naziv). Klikom na strelice ispod skraćenog naziva pojma moguće je sortirati cijelu tablicu po odabranom stupcu silazno ili uzlazno.



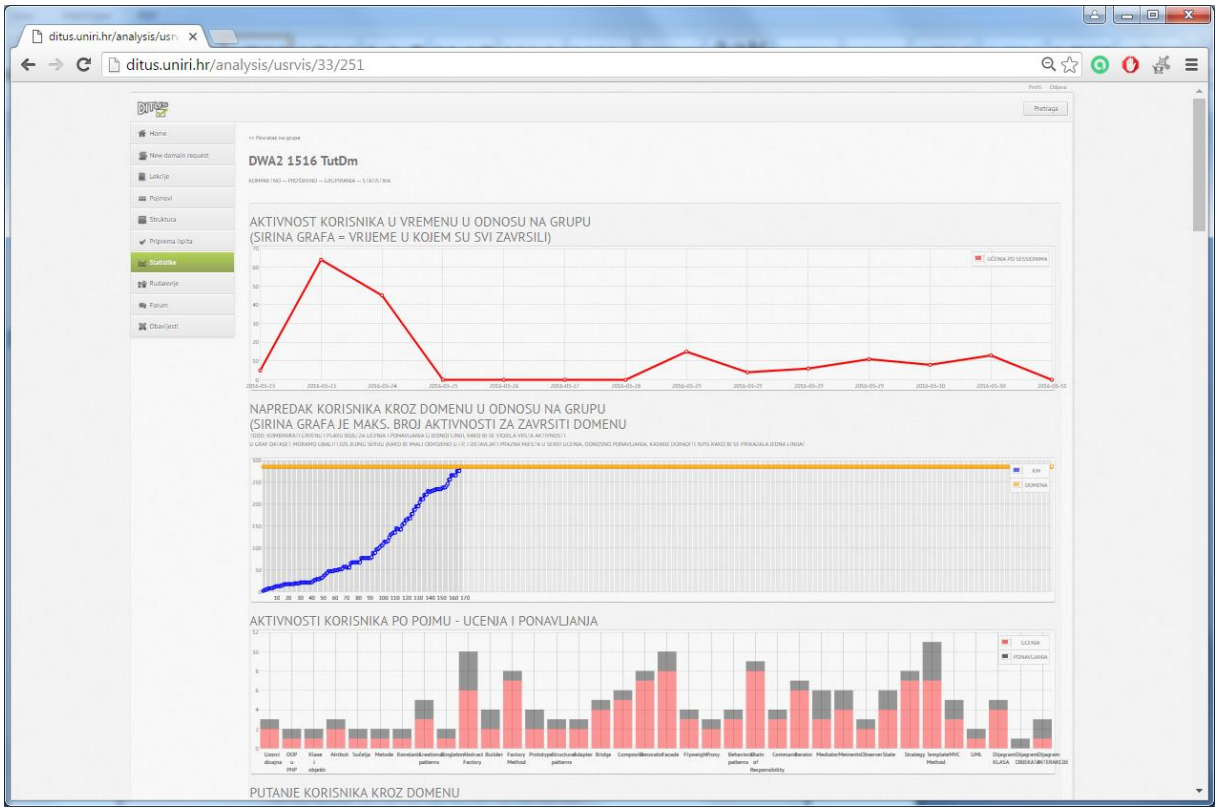
Slika 54. Detaljni pregled učenja ili ponavljanja s vizualizacijom intenziteta aktivnosti

Profesoru su, preko poveznice „Group activity visualization“, dostupne vizualizacije intenziteta aktivnosti na razini grupe ili na razini jednog studenta. Na slici 55 prikazana je vizualizacija intenziteta aktivnosti cijele grupe (ispod koje se prikazuje kumulativna vizualizacija intenziteta aktivnosti po studentima grupe).

Vizualizacija podataka o aktivnostima jednog studenta omogućava njegovo praćenje kroz cijeli period dostupnosti sustava, otkrivanje dinamike savladavanja domene znanja u odnosu na ostatak grupe, distribuciju aktivnosti po pojmovima domene, te druge korisne informacije. Na slici 56 vidljiv je pregled vizualizacija aktivnosti jednog studenta: a) pregled intenziteta aktivnosti po danima, b) krivulja učenja po postavljenim pitanjima, te c) pregled aktivnosti učenja i ponavljanja po svakom pojmu u domeni znanja.

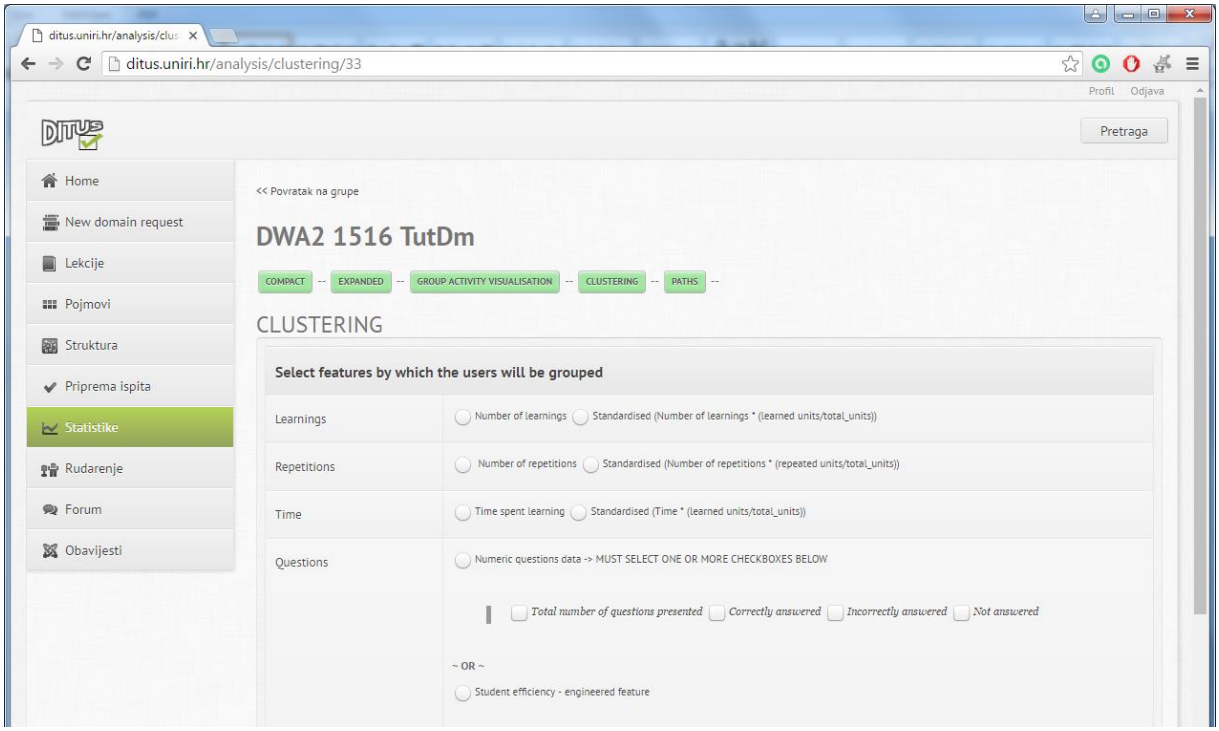


Slika 55. Sučelje za vizualizaciju aktivnosti grupe u vremenu



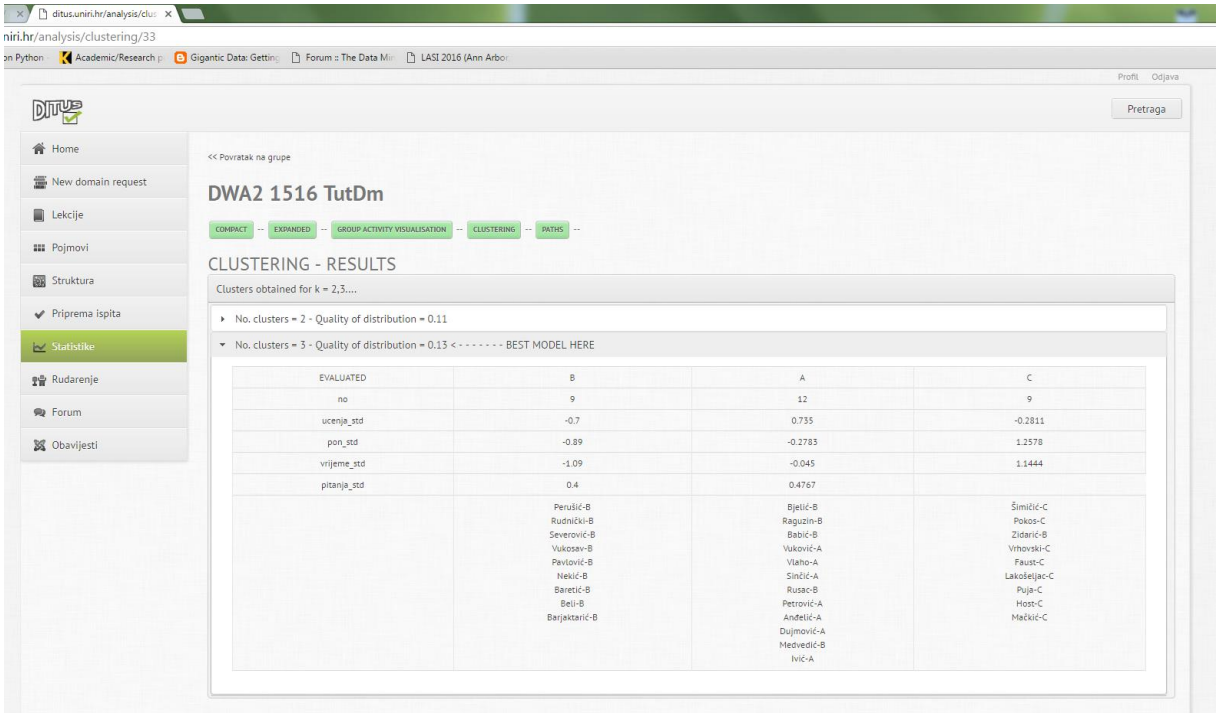
Slika 56. Sučelje za prikaz vizualizaciju podataka jednog studenta

Sučelje prikazano na slici 57 omogućava profesoru pokretanje dubinske analize podataka – klasteriranja studenata, opisane u poglavlju 3.2. Na početnom ekranu profesor odabire značajke koje želi uključiti u analizu.



Slika 57. Sučelje za pokretanje postupka klasteriranja

Nakon pokretanja postupka (Slika 58), prikazuju se svi kreirani modeli (sa rastućim brojem skupina - klastera).



Slika 58. Prikaz rezultata procesa klasteriranja

Profesor može analizirati sve prikazane modele, vrijednosti centroida svake skupine, članove dodjeljene skupini, itd. Sustav daje informaciju o tome koji je model odabran kao najbolji. Modeli se listaju pomoću tzv. *Accordion* kontrole razvijene primjenom JQuery JavaScript platforme.

Odabirom akcije „Paths“ profesor pokreće dubinsku analizu putanja korisnika opisanu u poglavlju 3.3., nakon čega dobiva kratki pregled rezultata (statistike provedenih dubinskih analiza putanja).

5 Verifikacija rada sustava

U ovom poglavlju analiziraju se funkcionalnosti sustava čiji je formalni opis dan u poglavlju 3., a detalji implementacije opisani u poglavlju 4. Ključne funkcionalnosti sustava nužne za poboljšanje njegove prilagodljivosti i učinkovitosti, pored onih razvijenih za pripremu domene znanja i provedbu procesa učenja, su:

a) klasteriranje studenata,
b) otkrivanje čestih i učinkovitih putanja
c) odabir i prezentacija sugestija.

Jedna od ključnih razlika ovog sustava u odnosu na sustave spomenute u poglavlju 2.2, je provedba dubinske analize nad rastućim skupom zapisa o interakcijama korisnika sa sustavom, što podrazumijeva da se spomenute funkcionalnosti izvode istovremeno s korištenjem sustava odnosno provedbom procesa učenja. Funkcionalnosti klasteriranja i otkrivanja čestih putanja izvode se periodički, u vremenskim intervalima čija duljina je određena intenzitetom aktivnosti korisnika. Sustav obavještava profesora o potrebi pokretanja dubinske analize nakon svakih 400 novih zapisa o aktivnostima korisnika u eksperimentalnoj skupini. Ovaj broj je odabran na temelju preliminarnih istraživanja, tijekom kojih je uočeno da provedba analiza podataka kod manjeg broja zapisa (npr. svakih 200 ili 300) ne generira (veće) promjene u ranije dobivenim modelima, dok bi veći broj (npr. svakih 500, 600) uzrokovao preveliko kašnjenje ažuriranja modela u odnosu na aktivnosti korisnika. Ovaj uvjet se automatski provjerava skriptom postavljenom u *crontab* operativnog sustava Linux svakih 15 minuta, čime se smanjuje kašnjenje sustava za promjenama u bazi podataka. Postupak klasteriranja studenata izvodi se prvi, nakon čega se pokreće postupak otkrivanja čestih putanja za svaki klaster korisnika u odabranom modelu klasteriranja. Treća, ključna, funkcionalnost izvodi se kontinuirano (u realnom vremenu) kao dio poboljšanog tutorskog modela sustava, odnosno kao dio procesa učenja i ponavljanja. Postupak odabira i prezentacije sugestija oslanja se na podatke koje kreiraju i ažuriraju prethodne dvije funkcionalnosti. Ova funkcionalnost je jedina koja je vidljiva korisnicima sustava, te se manifestira kroz sučelja prikazana na Slikama 47. i 51. u poglavlju 5. Sugestije ostvaruju osnovni cilj ovog istraživanja – nude korisniku podršku u kreiranju vlastite strategije za savladavanje domene znanja.

Sve analize prikazane u ovom poglavlju odnose na se eksperimentalnu skupinu korisnika (za koju su provođene dubinske analize podataka, te nuđene sugestije). Uz svaku analizu istaknuto je na koju domenu znanja se podaci odnose. Na kraju analize rada svake od tri

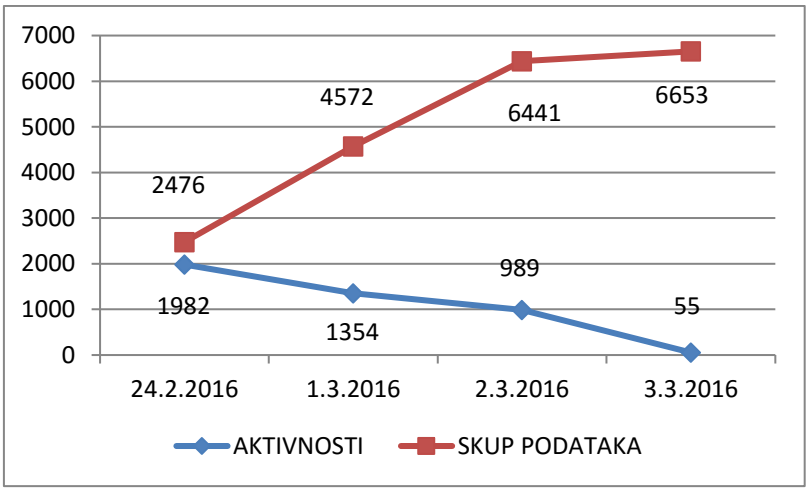
ključne funkcionalnosti sustava daju se zaključci o uočenim prednostima i nedostacima, te ističu pravci budućeg razvoja.

5.1 Verifikacija rada podsustava za klasteriranje studenata

Podsustav za klasteriranje studenata oslanja se na modul za komunikaciju s alatima za dubinsku analizu podataka, te se sastoji od tri faze:

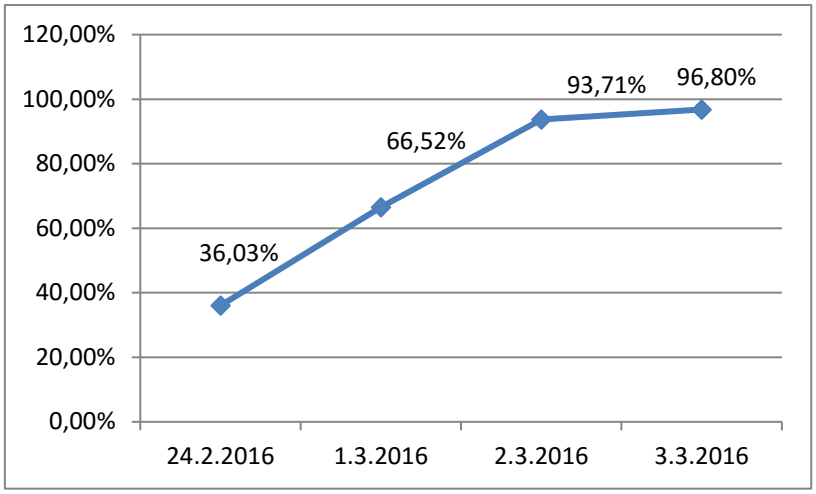
a) kreiranje skupa modela klasteriranja (sa rastućim brojem skupina k=2,3...n)
b) evaluacije kreiranih modela i odabira najbolje ocjenjenog modela
c) interpretacije načina učenja u odabranom modelu s ciljem procjene kvalitete procesa učenja svake skupine, odnosno kreiranja poretka skupina kao preduvjeta izvođenja sljedeće ključne funkcionalnosti – otkrivanja čestih putanja učenja.

Kao što je ranije spomenuto, analize podataka provode se na rastućem skupu podataka, dok korisnici savladavanju domenu znanja. Slika 59. prikazuje rast skupa podataka o interakcijama korisnika sa sustavom, te intenzitet aktivnosti korisnika po danima u domeni znanja „Osnovne web tehnologije“.

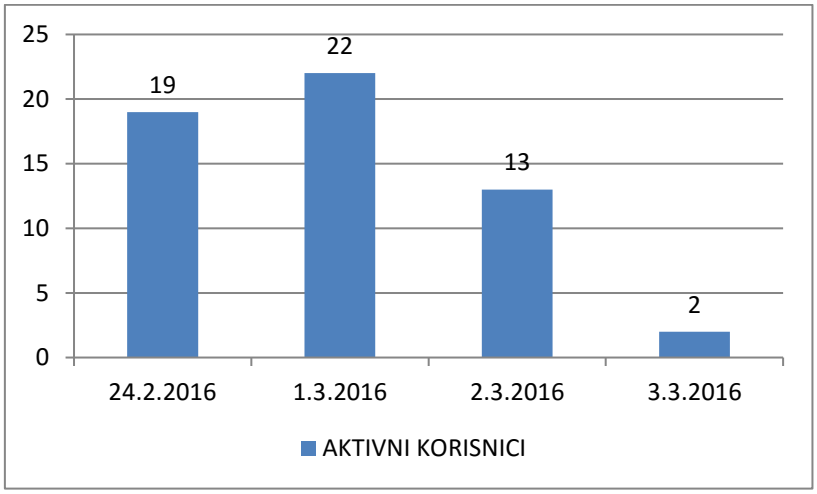


Slika 59. Rast skupa podataka i aktivnosti korisnika eksperimentalne skupine po danima

Rast skupa podataka korelira s rastom postotka ukupne savladanosti domene cijele skupine (Slika 60) i s brojem aktivnih korisnika (Slika 61) iz čega je vidljivo da su sve aktivnosti posljednjeg dana korištenja sustava kreirala dva studenta.



Slika 60. Savladanost domene eksperimentalne skupine po danima



Slika 61. Broj aktivnih studenata eksperimentalne skupine po danima

Intenzitet aktivnosti korisnika po danima prikazuje Tablica 7, iz koje je vidljivo da su korisnici domenu znanja savladavali u prosjeku dva dana. uz najveći intenzitet u prva dva dana otvorenosti sustava korisnicima. Intenzitet boja izražava postotak savladavanja domene svakog studenta po danima.

Važno je još jednom istaknuti da se dubinske analize podataka s ciljem kreiranja sve kvalitetnijih sugestija provode sukladno intenzitetu aktivnosti korisnika, što znači da sustav nema većeg kašnjenja, odnosno nije moguća situacija u kojoj će se samo korisnicima koji koriste sustav zadnjeg dana prezentirati sugestije. Drugim riječima, ukoliko intenzitet aktivnosti bude vrlo visok tijekom jednog dana, sustav će tog dana provesti onoliko analiza koliko puta bude dosegnut uvjet spomenut u uvodnom dijelu ovog poglavlja, te nuditi sugestije otkrivene na temelju nanovijeg skupa podataka.

Tablica 7. Savladanost domene studenata eksperimentalne skupine po danima

Studenti	DANA AKT	PO DANIMA			
		1	2	3	4
Student 1	2	0%	81%	19%	0,0%
Student 2	1	0%	0%	100%	0,0%
Student 3	2	31%	74%	0%	0,0%
Student 4	3	34%	15%	62%	0,0%
Student 5	1	100%	0%	0%	0,0%
Student 6	2	96%	4%	0%	0,0%
Student 7	3	4%	79%	22%	0,0%
Student 8	2	25%	85%	0%	0,0%
Student 9	4	1%	31%	48%	32,9%
Student 10	2	58%	46%	0%	0,0%
Student 11	1	100%	0%	0%	0,0%
Student 12	1	100%	0%	0%	0,0%
Student 13	2	51%	54%	0%	0,0%
Student 14	2	0%	0%	100%	0,0%
Student 15	3	5%	26%	73%	0,0%
Student 16	3	30%	8%	0%	72,6%
Student 17	3	8%	61%	37%	0,0%
Student 18	1	100%	0%	0%	0,0%
Student 19	1	100%	0%	0%	0,0%
Student 20	2	0%	5%	100%	0,0%
Student 21	2	0%	4%	97%	0,0%
Student 22	2	54%	46%	0%	0,0%
Student 23	1	0%	7%	0%	0,0%
Student 24	2	0%	43%	65%	0,0%
Student 25	2	0%	2%	100%	0,0%
Student 26	2	0%	91%	11%	0,0%
Student 27	2	47%	55%	0%	0,0%
Student 28	1	100%	0%	0%	0,0%
Student 29	1	0%	100%	0%	0,0%
PROSJEK DANA	1,93	19	22	13	2

Podaci prikazani u ovom dijelu pripadaju domeni „Osnovne web tehnologije“.

5.1.1 Verifikacija postupka kreiranja skupa modela grupiranja

Tijekom korištenja sustava provedeno je 9, odnosno 13 postupaka grupiranja studenata, kao što je prikazano u Tablici 8. Prosječno vrijeme kreiranja skupa modela nije prelazilo pet sekundi, zahvaljujući relativno malom broju značajki u skupu podataka i broju studenata u skupini. Svaka analiza kreira jednu datoteku na poslužitelju koja sadrži ulazni skup podataka za alat za dubinsku analizu podataka u kojem se provodi analiza (Weka), dok se rezultat

preuzima sa standardnog izlaza odnosno direktno s izlaza procesa pokrenutog u ljusci operativnog sustava Linux.

Tablica 8. Broj analiza grupiranja po danima za obje domene

Osnovne web tehnologije	
24.2.2016	2
26.2.2016	1
1.3.2016	4
2.3.2016	2
3.3.2016	0
UKUPNO	9

Uzorci dizajna	
23.3.2016	1
24.3.2016	1
26.3.2016	1
28.3.2016	1
29.3.2016	2
30.3.2016	5
31.3.2016	2
UKUPNO	13

Rezultati se pohranjuju u bazu podataka (model prikazan na slici 20 u poglavlju 4.1.5).

5.1.2 Verifikacija postupka odabira modela

Po okončanju postupka kreiranja skupa modela grupiranja, metodom siluete opisane u Poglavlju 3.2.3 (Algoritam 2), automatski se odabire najbolji model (model s najvišom vrijednošću siluete). U sljedećoj tablici prikazani su svi kreirani modeli i njihove vrijednosti siluete, a posebno su istaknuti odabrani modeli odnosno broj skupina u odabranom modelu.

Tablica 9. Pregled kreiranih modela grupiranja u domeni „Osnovne web tehnologije“

RBR	DATUM I VRIJEME	BROJ SKUPINA I KVALITETA MODELA						
1	24.02.2016 16:18:08	2	**3**	4	5			
		0,41	**0,56**	0,55	0,4			
2	24.02.2016 17:01:31	2	3	4	5	6	**7**	
		0,45	0,26	0,33	0,36	0,24	**0,51**	
3	26.02.2016 15:59:48	**2**	3	4	5	6	7	8
		0,57	0,20	0,22	0,13	0,23	0,21	0,26
4	01.03.2016 00:20:13	2	3	**4**				
		0,27	0,38	**0,48**				
5	01.03.2016 16:03:49	**2**	3	4	5	6		
		0,68	0,44	0,45	0,43	0,42		
6	01.03.2016 18:40:55	2	3	**4**	5	6		
		0,21	0,49	**0,55**	0,36	0,29		
7	01.03.2016 23:41:00	2	3	**4**	5	6	7	8
		0,21	0,35	**0,50**	0,35	0,39	0,25	0,16
8	02.03.2016 16:26:46	**2**	3	4				
		0,48	0,40	0,23				
9	02.03.2016 21:05:59	**2**	3	4	5	6		
		0,51	0,33	0,24	0,16	0,35		

Najviša postignuta vrijednost siluete bila je 0,68 odnosno 0,6, kao što je prikazano u Tablicama 9 i 10.

Kod ranijih analiza uočava se vrlo niska vrijednost siluete, što se može pripisati utjecaju većeg broja neaktivnih studenata, te nekolicini studenata koji su imali vrlo velike razlike u vrijednostima značajki u modelu podataka (neki studenti su završili domenu i to s vrlo velikim brojem učenja, dok su drugi tek počinjali), dakle postojanju iznimnih vrijednosti (engl. *outlier*) u skupu podataka što je poznati nedostatak algoritma kMeans. Sa količinom podataka, odnosno s povećanjem broja aktivnih korisnika postižu se sve više vrijednosti siluete.

Tablica 10. Pregled kreirani modela grupiranja u domeni "Uzorci dizajna"

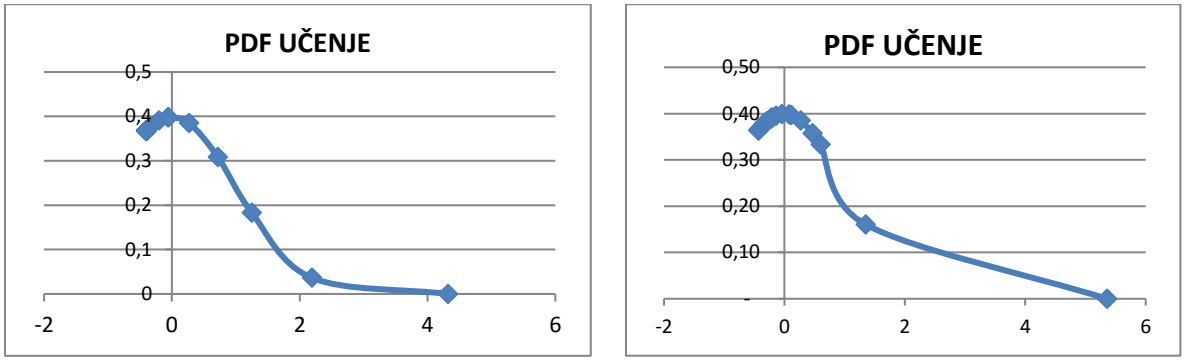
RBR	DATUM I VRIJEME	BROJ SKUPINA I KVALITETA MODELA							
1	23.03.2016 21:16:47	2 0,16	**3** **0,17**						
2	24.03.2016 14:03:53	2 0,1	3 0,13	4 0,16	**5** **0,21**				
3	26.03.2016 18:41:13	2 0,11	**3** **0,17**						
4	28.03.2016 19:44:25	2 0,15	**3** **0,22**						
5	29.03.2016 18:22:41	2 0,11	**3** **0,13**						
6	29.03.2016 20:13:36	2 0,39	3 0,32	**4** **0,41**	5 0,31	6 0,31	7 0,31	8 0,27	
7	30.03.2016 09:18:32	**2** **0,38**	3 0,38	4 0,36	5 0,27	6 0,35	7 0,32		
8	30.03.2016 14:50:03	2 0,36	3 0,33	**4** **0,42**	5 0,42				
9	30.03.2016 17:02:44	2 0,2	**3** **0,32**	4 0,19	5 0,26	6 0,25	7 0,19		
10	30.03.2016 18:48:50	**2** **0,60**	3 0,14	4 0,2	5 0,42				
11	30.03.2016 21:36:26	**2** **0,45**	3 0,18	4 0,13	5 0,16	6 0,39	7 0,18	8 0,18	
12	31.03.2016 00:01:54	**2** **0,3**	3 0,15	4 0,09	5 0,05	6 0,14	7 0,13	8 0,1	9 0,23
13	31.03.2016 09:03:55	2 0,3	**3** **0,53**	4 0,12	5 0,3	6 0,34	7 0,1		

Cilj ovog istraživanja bilo je ispitivanje mogućnosti automatizacije postupaka dubinske analize podataka. S obzirom da je metoda siluete inicijalno zamišljena kao metoda vizualizacije kvalitete provedenog postupka grupiranja, njeno iscrtavanje pored svakog kreiranog modela, moglo bi biti od koristi profesoru za samostalni odabir željenog modela. Potreba za uključivanjem profesora u postupak odabira dobivenih modela grupiranja ističe se i pri verifikaciji sljedeće faze – interpretacije kvalitete procesa učenja po skupinama odabranog modela.

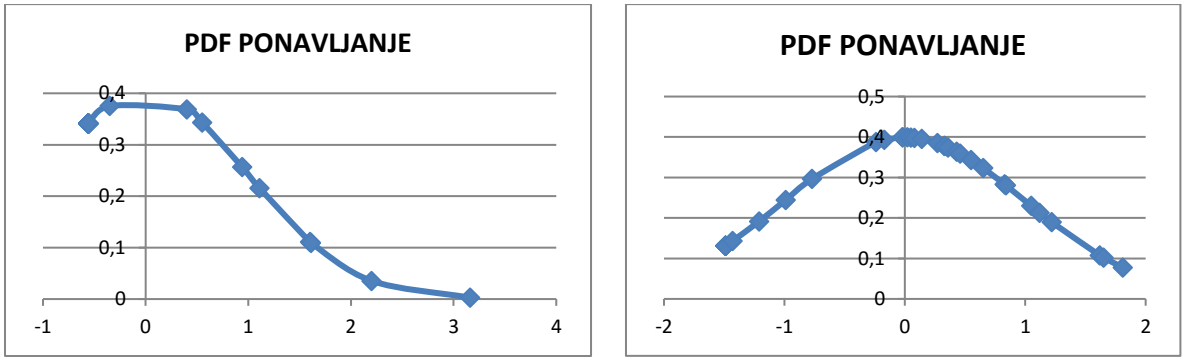
5.1.3 Verifikacija postupka interpretacije kvalitete procesa učenja

Postupak interpretacije kvalitete procesa učenja klastera u odabranom modelu opisan je u Poglavlju 3.2.4. Temelji se na uočenoj sličnosti distribuciji vrijednosti značajki učenja, ponavljanja i vremena učenja normalnoj distribuciji. Na Slikama 62, 63 i 64 uspoređuju se krivulje razdiobe vjerojatnosti značajki (u Domeni „Uzorci dizajna) koje sačinjavaju (pored značajke: učinkovitost) ulazni skup podataka za algoritam klasteriranja. Riječ je o značajkama koje se odnose na broj učenja L_{std}, broj ponavljanja R_{std} i vrijeme učenja T_{std}. Uspoređen je oblik krivulje kod prve (na najmanjem skupu podataka) i posljednje analize (na konačnom, najvećem skupu podataka). Kod značajki ponavljanja i vremena učenja uočava se približavanje oblika krivulje normalnoj distribuciji, dok se kod značajke učenja manifestira problem iznimaka u skupu podataka (dva studenta s iznimno visokim brojem aktivnosti učenja uzrokuju izobličenje krivulje). Na X osi prikazanih grafikona nalaze se standardne devijacije (jer su vrijednosti spomenutih značajki standardizirane), iz čega je vidljivo da su vrijednosti značajki većine studenata (rombovi) grupirani oko vrijednosti 0 odnosno najmanje udaljeni od prosjeka grupe.

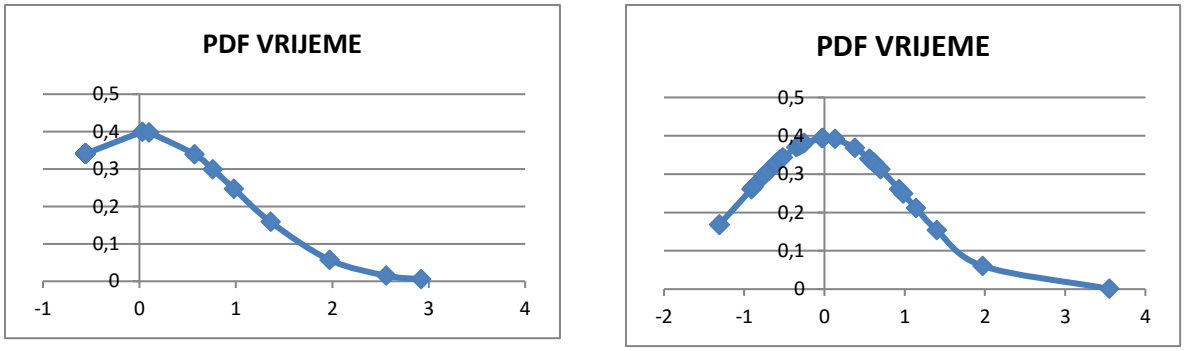
Kod ranih analiza najveća (najviša) vrijednost funkcije distribucije rezultata kod spomenutih značajki često bude kod vrijednosti nula, što neaktivne korisnike najčešće stavlja u središnji klaster (ispod aktivnih korisnika koji napreduju brže i aktivnih korisnika koji napreduju sporije, odnosno imaju visok broj aktivnosti učenja). Taj nedostatak može se jednostavno riješiti izostavljanjem neaktivnih studenata iz skupa podataka pripremljenog za dubinsku analizu podataka, odnosno automatskim kreiranjem dodatnog, najlošije ocjenjenog klastera korisnika. Ovaj problem je nastao zbog pretpostavke proizašle iz preliminarnih testova prototipa sustava u kojem su svi studenti bili aktivni (barem djelomično savladali domenu), pa nije dolazilo do ove anomalije sustava.



Slika 62. Usporedba pdf(L_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna"



Slika 63. Usporedba pdf(R_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna"



Slika 64. Usporedba pdf(T_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna"

Usporedbom prve i posljednje analize lako se uočava realnija distribucija vrijednosti rezultata pojedinih značajki koja omogućava sve bolje razultate dubinske analize podataka.

Problem iznimnih vrijednosti u skupu podataka uočava se u distribuciji vrijednosti značajke učenja, kod koje je vidljiv utjecaj dvoje studenata s iznimno visokim brojem aktivnosti učenja (524, 660) koji snažno utječe na prosječnu vrijednost aktivnosti učenja nužnu za izračun funkcije distribucije (bez dvoje studenata prosjek je 90, s njima 122). Iako je predloženi algoritam procjene kvalitete procesa učenja prilagodljiv „širini“ distribucije rezultata (čime se

umanjuje utjecaj iznimaka), iznimke utječu na rezultate samog *kMeans* algoritma koji se izvodi u povezanom alatu za dubinsku analizu podataka. Ponovno se nameće zaključak o potrebi prepuštanja (dijela) kontrole profesoru, odnosno razvoja dodatnih funkcionalnosti kojima bi profesor mogao odabrati studente koje neće ući u konačan skup podataka, već u neki „ručno“ kreirani klaster ili usporediti rezultate sa ili bez neaktivnih ili iznimnih studenata. Spomenute nadogradnje ugraditi će se i testirati u sljedećim verzijama prototipa sustava.

5.2 Verifikacija rada podsustava za okrivanje čestih i učinkovitih putanja

Podsustav za otkrivanje čestih i učinkovitih putanja izvršava se nakon provedbe postupka grupiranja studenata. Za svaku skupinu odabranog modela grupiranja provodi se postupak otkrivanja čestih putanja za svaki od pojmova u domeni znanja, zasebno za putanje koje vode do pojma („prefix“ putanje) i za putanje koje vode od pojma („sufix“ putanje). Ovaj postupak oslanja se na algoritam USPAN opisan u poglavlju 3.3, implementiran u alatu SPMF. Glavni preduvjet za pokretanje postupka otkrivanja čestih i učinkovitih putanja za neki pojam domene, u nekoj skupini studenata, je da je pojam učilo barem pet studenata čime se osigurava da otkrivene putanje (uzorci) zaista imaju karakteristiku učestalosti odnosno da je iste putanje imalo više različitih studenata. Taj preduvjet osigurava i postojanje minimalno pet putanja/zapisa (iako ih može biti više) u ulaznoj datoteci algoritma. Na ovaj način algoritam se kod ranijih analiza poziva rjeđe, dok se u kasnijim analizama poziva za sve više (ili sve) pojmova u domeni, za sve klastere studenata. Opterećenje poslužitelja na kojem je sustav pokrenut time raste s ukupnom količinom podataka.

Ključan utjecaj na broj otkrivenih putanja ima broj studenata u klasteru, koji se može promijeniti kod svake nove analize klasteriranja studenata. Ukoliko je broj studenata u klasteru malen (npr. 1, 2 studenta), to će smanjiti potencijalni skup podataka, te nužno smanjiti i broj otkrivenih putanja. Međutim, taj nedostatak umanjuje algoritam za odabir sugestija (opisan u poglavlju 3.4), koji pri pretrazi čestih putanja koristiti otkrivene putanje klastera koja je ocjenjena jednom ocjenom više od trenutnog klastera.

Ukoliko veliki broj studenata ima veliki broj aktivnosti na istom pojmu, za taj pojam će vjerojatno biti otkriven najveći broj putanja, od kojih će one najučinkovitije biti prezentirane korisnicima u obliku sugestija. Također će se koristiti putanje za taj pojam otkrivene u skupini ocjenjenoj jednu razinu bolje od trenutne skupine. Time se ostvaruje jedan od osnovnih

ciljeva sustava – pružanje podrške studentima kod savladavanja složenijih pojmova, odnosno pronalaska učinkovitijeg puta kroz domenju znanja.

Broj provedenih analiza čestih putanja jednak je onom prikazanom u Tablici 8 jer se analiza čestih putanja pokreće nakon klasteriranja korisnika.

Općeniti pregled kretanja broja otkrivenih putanja po tipu, te broj pojmova za koji su otkrivene putanje dan je u Tablici 11 za domenu „Osnovne web tehnologije“ i Tablici 12 za domenu „Uzorci dizajna“.

Tablica 11. Broj provedenih analiza putanja u domeni „Osnovne web tehnologije“

RBR	PUTANJA	PREFIX	SUFIX	POJMOVA (31)
1	21	5	16	7
2	35	20	15	15
3	115	65	50	22
4	121	79	42	18
5	89	52	37	21
6	418	227	191	31
7	538	289	249	31

Tablica 12. Broj provedenih analiza putanja u domeni „Uzorci dizajna“

RBR	PUTANJA	PREFIX	SUFIX	POJMOVA(36)
1	47	37	10	7
2	50	38	12	15
3	147	89	58	20
4	298	169	129	27
5	409	222	187	28
6	637	348	289	32
7	918	449	469	34
8	737	382	355	35
9	1352	634	718	36
10	1858	880	978	36
11	2041	966	1075	36
12	2478	1206	1272	36
13	2541	1233	1308	36

Iz tablica je vidljivo da je u domeni „Uzorci dizajna“ nešto ranije postignuta potpuna „pokrivenost“ pojmova putanjama (otkrivene su česte putanje za sve pojmove u domeni), što se može objasniti nešto većim brojem aktivnih studenata, te složenošću domene, koja obrađuje studentima nepoznato područje (što uzrokuje više aktivnosti učenja istog pojma, odnosno generiranje većeg skupa podataka), za razliku od domene „Osnovne web tehnologije“ koja se sastoji od pojmova koje studenti obrađuju na više kolegija tijekom

dosadašnjeg studija, te im je prolazak kroz tu domenu bio jednostavniji, što se odražava na broj učenja i ponavljanja, a zatim i na broj pojmova za koje su pronađene česte putanje.

Tablica 13 prikazuje broj otkrivenih putanja po klasteru i tipu (P = prefix, S = sufix). Evaluirani klasteri dobivaju ocjenu u obliku slova pri čemu je A najviša ocjena. U tablici se može uočiti izostanak otkrivenih putanja za klaster (najčešće pod slovom B) neaktivnih korisnika. Retci prikazuju broj otkrivenih putanja, sumu otkrivenih putanja po tipu, te ukupnu sumu putanja po analizi.

Tablica 13. Pregled broja otkrivenih putanja po analizi (po klasteru i tipu) u domeni „Osnovne web tehnologije“

RBR	KLASTER	TIP	UKUPNO PO TIPU
1	A	P	35
		S	10
			45
	B	P	2
			47
2	B	P	5
		S	5
			10
	C	P	3
		S	5
			8
	D	P	27
		S	2
			29
	E	P	3
			3
			50
3	A	P	5
		S	9
			14
	C	P	84
		S	49
			133
			147
4	A	P	8
		S	10
			18
	C	P	161
		S	119
			280
			298
5	A	P	8
		S	10
			18
	C	P	161
		S	119
			280
			298
6		P	61
		S	63

	A		124
	C	P	161
		S	124
			285
			409
7	A	P	29
		S	27
			56
	B	P	243
		S	188
			431
	D	P	76
		S	74
			150
			637

Tijekom korištenja sustava za domenu znanja „Osnovne web tehnologije“ sustav je kreirao ukupno 463 datoteke s ulaznim i izlaznim podacima, dok je za domenu „Uzorci dizajna“ kreirano 1995 datoteka. Kao što je ranije definirano, najmanji skup podataka može se sastojati od pet putanja odnosno redaka u ulaznoj datoteci, dok je, kod posljednje analize, najveći broj redaka u ulaznom skupu podataka za jedan pojam bio 139.

5.3 Verifikacija rada unaprijeđenog tutorskog modula

Unaprijeđeni model tutorskog sustava oslanja se na unaprijeđeni model studenta (koji je, izvođenjem postupka otkrivanja grupa, te automatskim odabirom i evaluacijom modela, proširen informacijom kojoj skupini student pripada, te ocjenom skupine), te na skup čestih i učinkovitih putanja (otkrivenih izvođenjem postupka otkrivanja i automatske evaluacije učinkovosti putanja) za svaku grupu. Svrha je unaprijeđenog tutorskog modela da, kroz izvođenje algoritma 6., opisanog u poglavlju 3.4, korisnicima prezentira sugestije koje bi, prema objektivnim podacima, trebale pozitivno utjecati na proces učenja u smislu poboljšanja njegove učinkovitosti (skraćivanje broja potrebnih aktivnosti i vremena potrebnog za savladavanje domene, uz istovremeno poboljšanje korisničkog iskustva (eng. *user experience*). Sugestije se prikazuju:

- na početku procesa UČENJA - sugestije iz skupine „prefix“ putanja, sugeriraju korisniku koji pojam učiti PRIJE trenutno odabranog pojma,
- na kraju procesa UČENJA - sugestije iz skupine „sufix“ putanja, sugeriraju korisniku koji pojam učiti NAKON trenutno odabranog pojma
- na kraju procesa PONAVLJANJA – sugestije iz skupine „sufix“ putanja, sugeriraju korisniku koji pojam učiti NAKON trenutno odabranog pojma

Studenti koji među prvima savladaju domenu znanja, u trenucima kada sustav još nema dovoljno podataka o putanjama, neće vidjeti sugestije. Također, studenti kojima su sugestije prikazane mogu ih ignorirati.

U Tablicama 14 i 15 prikazan je ukupan broj prikazanih sugestija po svakoj provedenoj dubinskoj analizi podataka za domene „Osnovne web tehnologije“ i „Uzorci dizajna“. Broj prikazanih sugestija ovisi o intenzitetu aktivnosti korisnika, pa je taj broj vrlo malen posljednjeg dana korištenja sustava zbog malog broja aktivnih korisnika. Prvi stupac u skupini PREFIX predstavlja broj prikazanih sugestija tog tipa, zatim za koliko pojmova su nuđene sugestije (JP = jedinstvenih pojmova), koliko različitih korisnika je vidjelo sugestije (JK = jedinstvenih korisnika), te koliko puta su sugestije korištene, odnosno koliko puta je korisnik kliknuo na sugeriranu poveznicu. Iz tablice je vidljivo da studenti nisu koristili sugestije tipa PREFIX, dok su sugestije tipa SUFIX korištene nešto više, ali sveukupno u relativno malom omjeru u odnosu na broj prikazanih sugestija.

Tablica 14. Prikazane i korištene sugestije u domeni „Osnovne web tehnologije“

		PREFIX				SUFIX			
ANALIZA	∑ PRIKAZANIH	∑ PREFIX	JP	JK	P	∑ SUFIX	JP	JK	P
1	12	6	2	3	0	6	3	3	2
2	32	18	7	5	0	14	7	5	6
3	186	70	11	6	0	116	14	6	8
4	130	46	5	4	0	84	10	4	4
5	32	8	5	3	0	24	9	3	4
6	436	184	14	5	0	252	22	5	6
7	313	127	15	4	0	186	21	4	5

Prikazani rezultati ističu potrebu za boljim informiranjem studenata o svrsi sugestija, načinu njihova kreiranja i odabira, te mogućem pozitivnom utjecaju na proces učenja. Također se ističe potreba za boljom vizualnom prezentacijom sugestija. Studenti su logičnijim slijedom aktivnosti smatrali sugestije tipa SUFIX, koje su prikazane kao odgovor na pitanje „Što dalje?“, od sugestija tipa PREFIX, koje su prikazane kao informacija „Prema rezultatima drugih korisnika, prije učenja ovog pojma bilo bi korisno učiti pojam *{{X}}*“.

Taj tip sugestija ostavlja dojam prekidanja tijeka učenja, za razliku od SUFIX sugestija koje ga (prirodno) nastavljaju, što može biti od razloga ignoriranja ponuđenih sugestija.

Razvojem prototipa sustava, prema ideji predloženoj u poglavlju 2.3, te modeliranom u poglavlju 3, ostvaren je jedan od ciljeva ovog istraživanja. Prototip demonstrira izvodivost ideje o integraciji alata za dubinsku analizu podataka u sustav za e-učenje.

Tablica 15. Prikazane i korištene sugestije u domeni „Uzorci dizajna“

		PREFIX				SUFIX			
ANALIZA	∑ PRIKAZANIH	∑ PREFIX	JP	JK	P	∑ SUFIX	JP	JK	P
1	7	2	1	1	0	5	2	2	5
2	4	3	2	1	0	1	1	1	2
3	75	30	9	4	0	45	10	5	6
4	98	67	8	6	0	31	5	4	1
5	186	74	12	5	0	112	17	6	7
6	254	151	24	7	0	103	25	9	4
7	364	192	23	5	0	172	25	6	2
8	337	190	17	7	0	147	23	8	5
9	479	197	20	5	0	282	35	8	12
10	570	282	24	7	0	288	27	8	1
11	454	210	18	5	0	244	29	7	35
12	454	192	19	4	0	262	33	4	12
13	68	15	11	3	0	53	20	3	19

Arhitekturu sustava karakterizira fleksibilnost u smislu odabira alata i odabira algoritama koji se žele izvoditi. Razvijena arhitektura omogućava da se koristi bilo koji alat koji ima razvijene biblioteke funkcija (API) za komunikaciju putem komandne linije ili web servisa, te da se izvodi bilo koji od algoritama koje nude ti alati. Za izvođenje novih algoritama potrebno je definirati mapiranje ulaznih parametara željenog algoritma u ispravan API upit prema alatu, te po potrebi osigurati ispravno oblikovanje ulaznog skupa podataka prije zapisivanja u datoteku na poslužitelju. Spomenute izmjene su, u odnosu na implementaciju jednog po jednog algoritma u kôd same aplikacije, tehnički vrlo jednostavne i vremenski nezahtjevne.

Na temelju provedene verifikacije rada sustava prva postavljena hipoteza ovog istraživanja:

Hipoteza 1 – dubinska analiza putanja tutorskog sustava može se koristiti za povećanje prilagodljivosti tutorskog sustava smatra se potvrđenom.

6 Opis istraživanja

Razvijeni prototip sustava DITUS, opisan u ranijim poglavljima testiran je u nastavi dvaju kolegija koji se izvode na preddiplomskom studiju Odjela za informatiku. U ovom poglavlju dan je opis istraživanja te su opisane subjektivne i objektivne metode vrednovanja sustava s aspekta djelotvornosti (provjere razine usvojenosti znanja), zadovoljstva korisnika (anketa provedena među studentima koji su koristili sustav), te učinkovitosti procesa učenja).

6.1 Sudionici

U istraživanju su sudjelovali studenti jednopredmetnog studija informatike (3. godina), te studenti dvopredmetnog studija informatike (3. godina). Sudionici istraživanja koristili su prototip sustava u sklopu kolegija Dinamičke web aplikacije 1 (domena: „Osnovne web tehnologije“) i Dinamičke web aplikacije 2 (domena: „Uzorci dizajna“).

6.1.1 Podjela sudionika u grupe

Kao osnova za podjelu studenata upisanih na kolegij Dinamičke web aplikacije 1 korišten je trenutni broj prikupljenih bodova na kolegiju. Kako je eksperiment (korištenje DITUS sustava) započet nakon svih drugih predviđenih aktivnosti kolegija, broj bodova dobro je odražavao moguće završne ocjene na kolegiju za svakog studenta. Na ovom kolegiju studenti tijekom semestra mogu prikupiti svih 100 ocjenskih bodova i to kroz pet aktivnosti (2 kolokvija * 30 bodova, 2 teorijska testa * 10 bodova i korištenje DITUS sustava * 20 bodova). Aktivni studenti na kolegiju svrstani su od najvećeg prema najmanjem broju bodova i zatim je svaki drugi student pridružen eksperimentalnoj skupini. Nakon kreiranja skupina izvršen je t-test (Zimmerman, 1997) na dva uzorka s podrazumijevanom jednakom varijancom. Domenu 1 koristilo je ukupno 54 studenata, od čega 48 studenata jednopredmetnog i 6 studenata dvopredmetnog studija, odnosno 27 studenata po grupi.

Rezultat t-testa, prikazan u Tablici 16, je pokazao da su skupine podjednake, odnosno da ne postoji statistički značajna razlika između članova kontrolne i eksperimentalne skupine. Konkretnije, nepostojanje statistički značajne razlike potvrđeno je činjenicom da je vrijednost „t Stat“ retka manja od kritične vrijednosti (redak „t Critical two-tail) potrebne da bi se razlika između dvaju skupina smatrala statistički značajnom. Također, vrijednost retka „P(t<=t) two tail“ je veća od 0,05 što je uobičajena mjera statističke značajnosti (odnosno, ukoliko je dobivena vrijednost veća od 0,05 skupine nisu statistički značajno različite).

Tablica 16. Rezultati t-testa skupina u domeni „Osnovne web tehnologije“

	Kontrolna skupina	*Eksperimentalna skupina*
Mean	34,91	33,28
Variance	172,72	154,32
Observations	27,00	27,00
Pooled Variance	163,52	
Hypothesized Mean Difference	0	
df	52	
t Stat	0,47	
P(T<=t) one-tail	0,32	
t Critical one-tail	1,67	
P(T<=t) two-tail	0,64	
t Critical two-tail	2,01	

Drugi eksperiment proveden je u sklopu kolegija Dinamičke web aplikacije 2. Kako je eksperiment provođen prije ostalih aktivnosti za prikupljanje ocjenskih bodova na kolegiju, a kolegij upisuju isti studenti koji su na prethodnom semestru slušali kolegij Dinamičke web aplikacije 1, odlučeno je da se broj bodova s tog kolegija uzme kao mjerilo kreiranja kontrolne i eksperimentalne skupine. Ovom skupu studenata priključen je i manji broj studenata koji ponavljaju kolegij Dinamičke web aplikacije 2, pri čemu je korišten isti pristup. Domenu 2 koristilo je ukupno 64 studenata, od čega 58 studenata jednopredmetnog i 6 studenata dvopredmetnog studija, odnosno 32 studenata po grupi.

Tablica 17. Rezultati t-testa skupina u domeni „Uzorci dizajna“

	Kontrolna skupina	*Eksperimentalna skupina*
Mean	33,03	30,90
Variance	177,00	177,44
Observations	32,00	32,00
Pooled Variance	177,22	
Hypothesized Mean Difference	0,00	
df	62,00	
t Stat	0,64	
P(T<=t) one-tail	0,26	
t Critical one-tail	1,67	
P(T<=t) two-tail	0,53	
t Critical two-tail	2,00	

Aktivni studenti na kolegiju svrstani su od najvećeg prema najmanjem broju bodova i zatim je svaki drugi student pridružen eksperimentalnoj skupini. Nakon kreiranja skupina izvršen je t-test na dva uzorka s podrazumijevanom jednakom varijancom. Rezultat t-testa (Tablica 17) je

također pokazao da su skupine podjednake, odnosno da ne postoji statistički značajna razlika između skupina.

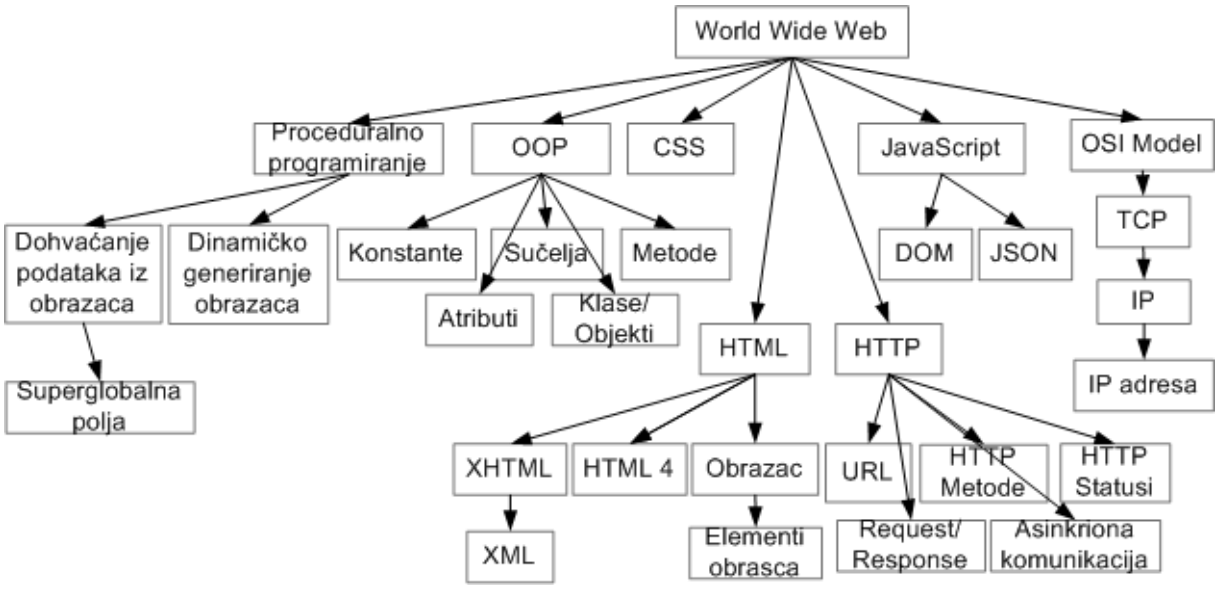
Nakon kreiranja skupina definirano je vremensko trajanje eksperimenta. Domenu znanja „Osnovne web tehnologije“ korisnici su mogli savladati u periodu od pet dana, dok je za savladavanje domene „Uzorci dizajna“ bilo predviđeno najviše sedam dana.

6.2 Domene znanja

Domene znanja nastale su na temelju sadržaja koji se obrađuju na predavanjima i vježbama gore spomenutih kolegija. Domena „Osnovne web tehnologije“ sadrži 31 pojam, koji obrađuju temeljne web standarde i skriptne jezike, te dio sadržaja obrađenih na vježbama iz kolegija. Domena „Uzorci dizajna“ sastoji se od 36 pojmova, od kojih velika većina otpada na uzorke dizajna opisane u (Vlissides, et al., 1995). Ova domena dijeli dio pojmova (5) s domenom „Osnovne web tehnologije“ ali uz prilagođene pragove, nastavne sadržaje i pitanja. Riječ je o pojmovima koji daju pregled sintakse PHP jezika za objektno orjentirano programiranje. U prvoj domeni cilj je savladavanje osnovnih elemenata sintakse PHP jezika za objektno orjentirano programiranje, dok navedeni pojmovi u drugoj domeni imaju ulogu podsjetnika, te prezentiraju složenije primjere i dodatne elemente ključne za implementaciju uzoraka dizajna.

6.2.1 Domena „Osnovne web tehnologije“

Domena znanja prikazana na slici 65 ima klasičnu hijerahijsku strukturu koja započinje pojmom „World Wide Web“ te se grana prema prikazanim skupinama pojmova. Dio prikazanih sadržaja/pojmova studenti obrađuju na kolegiju Dinamičke web aplikacije, dok su se s dijelom sadržaja mogli upoznati i na kolegijima koje su slušali ranije tijekom studija.



Slika 65. Struktura domene znanja „Osnovne web tehnologije“

Za svaki pojam (KU) u domeni definira se početna (s_{KUi}) i konačna vrijednost (h_{KUi}) koju je potrebno dosegnuti kako bi se pojam smatrao savladanim. Početna vrijednost uvijek je postavljana na nulu, dok je konačna vrijednost („prag“) određena sukladno procjenjenoj složenosti pojedinog pojma. Također se unose pitanja za provjeru i incijalna pitanja koja omogućavaju izvođenje tutorskog modela sustava. Za proces učenja ključni su nastavni materijali. U Tablici 18 su, pored pragova i broja pitanja (za ponavljanje i inicijalna), navedeni i sadržaji podjeljeni po tipu (definicija, tekst, primjeri (kôda), poveznice na vanjska web sjedišta i video materijale (Youtube), uneseni za svaki pojam domene.

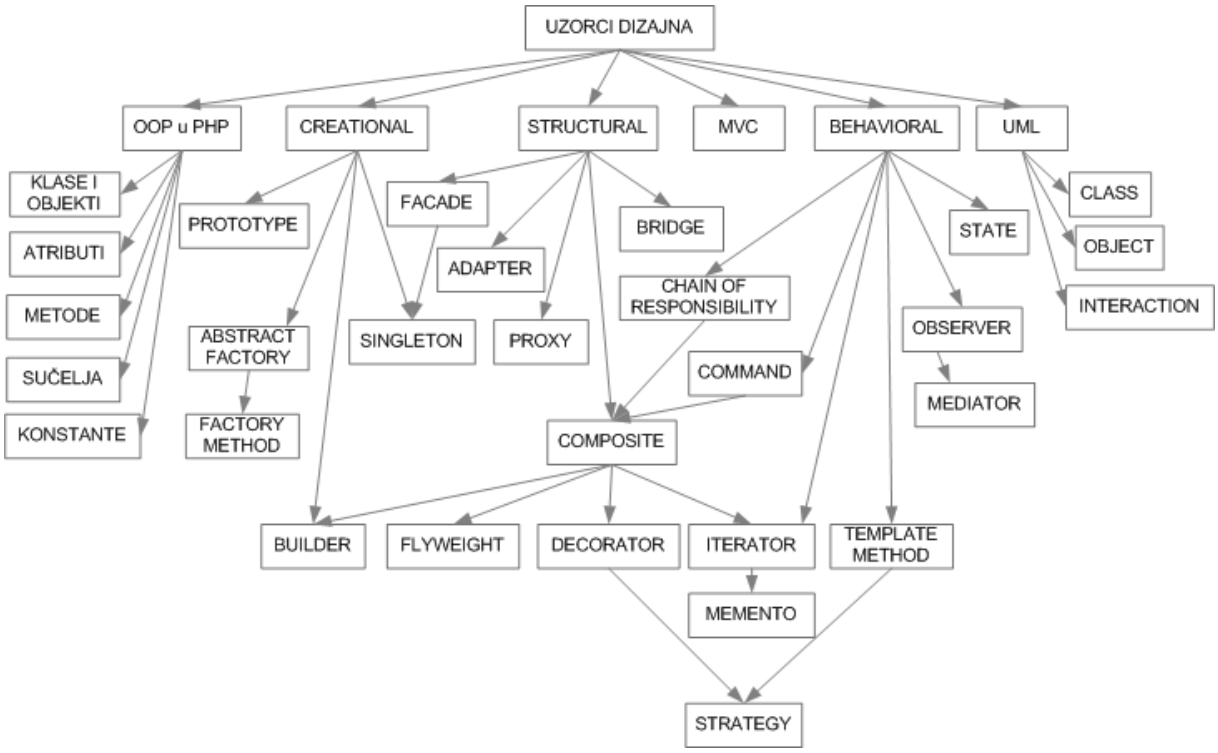
Tablica 18. Pojmovi, pragovi, pitanja i nastavni materijali za domenu „Osnovne web tehnologije“

				PITANJA			SADRŽAJI					
i	POJAM (ku_i)	s_{ku_i}	h_{ku_i}	PON.	INIC.	UKUPNO	DEFINICIJA	TEKST	PRIMJERI	LINKOVI	VIDEO	UKUPNO
1	Asinkrona komunikacija	0	4	5	3	8	1	1	0	2	0	4
2	Atributi	0	4	6	3	9	1	1	3	0	0	5
3	CSS	0	10	15	8	23	1	1	0	2	4	8
4	Dinamičko generiranje obrazaca	0	8	4	4	8	1	1	0	0	0	2
5	Dohvaćanje podataka iz obrazaca	0	6	5	4	9	1	1	0	0	0	2
6	DOM	0	10	7	4	11	1	1	2	3	1	8
7	Elementi obrasca	0	10	12	7	19	1	1	0	1	3	6
8	HTML	0	10	9	11	20	1	1	0	1	0	3
9	HTML 4.0	0	10	10	3	13	1	1	0	3	0	5
10	HTML Obrazac	0	10	6	4	10	1	1	0	2	0	4
11	HTTP	0	10	7	6	13	1	2	0	2	0	5
12	HTTP Metode	0	10	7	3	10	1	1	0	2	0	4
13	HTTP statusi	0	10	6	4	10	1	1	0	1	0	3
14	Internet protokol (IP)	0	10	6	4	10	1	1	0	2	0	4
15	IP adresa	0	10	8	6	14	1	1	0	1	1	4
16	JavaScript	0	10	5	5	10	1	1	0	4	0	6
17	JSON	0	6	5	5	10	1	1	0	3	0	5
18	Klase i objekti	0	4	6	5	11	1	1	7	0	0	9
19	Konstante	0	4	4	5	9	1	1	1	0	0	3
20	Metode	0	4	10	4	14	1	1	5	0	0	7
21	OOP u PHP	0	4	6	5	11	1	1	0	1	0	3
22	OSI model	0	10	9	5	14	1	1	0	2	0	4
23	Proceduralno programiranje	0	5	4	5	9	1	1	0	0	0	2
24	Request-Response	0	4	3	2	5	1	1	0	0	0	2
25	Sučelja	0	4	6	5	11	1	1	2	0	0	4
26	Superglobalna polja	0	6	4	6	10	1	1	0	1	0	3
27	TCP	0	6	8	3	11	1	1	0	2	0	4
28	URL	0	8	6	7	13	1	1	0	2	0	4
29	World Wide Web	0	10	5	6	11	1	1	0	2	0	4
30	XHTML	0	10	6	5	11	1	1	0	3	0	5
31	XML	0	10	12	4	16	1	1	0	0	0	2
	PROSJEČNE VRIJEDNOSTI			6,8	4,9	11,7	1,0	1,0	0,6	1,4	0,3	4,3

Iz Tablice 18 je vidljivo da je prosječno uneseno nešto manje od 12 pitanja po pojmu, te ponuđeno nešto više od četiri različite jedinice sadržaja.

6.2.2 Domena „Uzorci dizajna“

Struktura pojmova prikazana na slici 66 izrađena je po uzoru na strukturu uzoraka prikazanu u knjizi „Design patterns: elements of reusable object-oriented software“ (Vlissides, et al., 1995). Iako je naziv domene „Uzorci dizajna“, sadržaji povezani s pojmovima domene orjentirani su na implementaciju uzoraka dizajna pri razvoju web aplikacija, što često zahtjeva specifične dorade u kôdu uzorka. Zbog složenosti implementacije uzoraka nisu korišteni primjeri kôda kao vrsta sadržaja domene, već su korištene poveznice prema vanjskim sadržajima (web sjedištima) specijaliziranima za poučavanje o primjeni uzoraka dizajna u razvoju web aplikacija korištenjem skriptnog jezika PHP kojim se studenti služe na kolegijima Dinamičke web aplikacije 1 i 2.



Slika 66. Struktura domene znanja „Uzorci dizajna“

Kao i kod prethodne domene, za svaki pojam definirana je početna i konačna vrijednost (prag) koji je potrebno dosegnuti kako bi se pojam smatrao savladanim. Pojmovi koji se pojavljuju i u prethodnoj domeni imaju niže postavljen prag, te izmjenjene sadržaje i pitanja. Kao što je ranije navedeno, njihova svrha u ovoj domeni je da služe kao podsjetnici na elementarne i nešto naprednije elemente sintakse jezika PHP za pisanje objektno-orjentiranih aplikacija.

Tablica 19. Pojmovi, pragovi, pitanja i nastavni materijali za domenu „Uzorci dizajna“

				PITANJA			SADRŽAJI					
i	POJAM (ku_i)	s_{ku_i}	h_{ku_i}	PON.	INIC.	UKUPNO	DEFINICIJA	TEKST	PRIMJERI	LINKOVI	VIDEO	UKUPNO
1	Abstract Factory	0	10	8	5	13	1	1	0	3	1	6
2	Adapter	0	10	9	5	14	1	1	0	2	1	5
3	Atributi	0	4	8	4	12	1	1	3	2	0	7
4	Behavioral patterns	0	4	5	4	9	1	1	0	0	0	2
5	Bridge	0	10	9	4	13	1	1	0	2	1	5
6	Builder	0	10	11	5	16	1	1	0	2	1	5
7	Chain of Responsibility	0	10	10	3	13	1	1	0	2	1	5
8	Command	0	10	9	5	14	1	1	0	2	1	5
9	Composite	0	10	9	4	13	1	1	0	2	1	5
10	Creational patterns	0	4	3	4	7	1	1	0	1	0	3
11	Decorator	0	10	9	5	14	1	1	0	2	1	5
12	Dijagram INTERAKCIJE	0	5	5	5	10	1	1	0	1	2	5
13	Dijagram KLASA	0	6	8	5	13	1	1	0	3	0	5
14	Dijagram OBJEKATA	0	4	4	4	8	1	1	0	2	3	7
15	Facade	0	10	9	4	13	1	1	0	2	1	5
16	Factory Method	0	10	9	3	12	1	1	0	2	1	5
17	Flyweight	0	10	10	4	14	1	1	0	1	1	4
18	Iterator	0	10	9	4	13	1	1	0	2	1	5
19	Klase i objekti	0	4	7	6	13	1	1	7	2	0	11
20	Konstante	0	4	5	5	10	1	1	1	2	0	5
21	Mediator	0	10	10	5	15	1	1	0	2	1	5
22	Memento	0	10	10	4	14	1	1	0	2	1	5
23	Metode	0	4	10	6	16	1	1	5	2	0	9
24	MVC	0	10	6	6	12	1	1	0	1	2	5
25	Observer	0	10	10	5	15	1	1	0	2	1	5
26	OOP u PHP	0	4	6	5	11	1	1	0	1	1	4
27	Prototype	0	10	9	4	13	1	1	0	2	0	4
28	Proxy	0	10	9	4	13	1	1	0	2	1	5
29	Singleton	0	10	8	3	11	1	1	1	2	1	6
30	State	0	10	9	4	13	1	1	0	2	1	5
31	Strategy	0	10	9	4	13	1	1	0	2	1	5
32	Structural patterns	0	4	4	4	8	1	1	0	0	0	2
33	Sučelja	0	4	6	5	11	1	1	2	0	0	4
34	Template Method	0	10	6	4	10	1	1	0	2	1	5
35	UML	0	5	5	5	10	1	1	0	5	1	8
36	Uzorci dizajna	0	10	6	7	13	1	1	0	2	1	5
	PROSJEČNE VRIJEDNOSTI			7,8	4,5	12,3	1,0	1,0	0,5	1,8	0,8	5,2

Iz Tablice 19 je vidljivo da je prosječno uneseno nešto više od 12 pitanja po pojmu, te ponuđeno nešto više od pet različitih jedinica sadržaja.

Vrijeme potrebno za razvoj domene znanja i ostalih potrebnih sadržaja (pitanja) potrebnih za funkcioniranje sustava često se ističe kao najveći nedostatak tutorskih sustava. Na razvoj testnih domena opisanih u ovom radu utrošeno je oko 40 radnih sati po domeni iako je moguće utrošiti i znatno više vremena ukoliko se želi još kvalitetnije prezentirati sadržaje i osmisliti još veći broj pitanja. Ovdje prezentirane domene predstavljaju testne uzorke konačnih domena, koje bi u konačnici trebale sadržavati između 100 i 150 pojmova. Statična pitanja također predstavljaju problem u domenama znanja koje ne podliježu strogim formalnim pravilima, odnosno gdje je otežan ili onemogućen razvoj parametriziranih pitanja (Brusilovsky, et al., 2004) (kakav je primjerice relativno jednostavno razviti kod učenja osnovnih matematičkih operacija). Statična pitanja (odnosno tekst pitanja i ponuđenih odgovora) ostavljaju mogućnost korisniku da do točnih odgovora dođe pogađanjem, iako je u sustav DITUS ugrađena zaštita od osvježavanja sučelja za postavljanje pitanja i prikaz rezultata, kao i sustav nasumičnog odabira poretka ponuđenih odgovora (engl. *shuffle*). Sustav za kreiranje parametriziranih pitanja, kao i proširenje skupa vrsta pitanja odnosno načina davanja odgovora biti će značajno proširen u sljedećim iteracijama razvoja prototipa sustava.

6.3 Provjera znanja

Konačna provjera znanja za obje domene sastojala se od 30 pitanja višestrukog odabira. Vrijeme trajanja provjere znanja bilo je ograničeno na 25 minuta. Pitanja su obuhvaćala više pojmova odjednom, odnosno nisu bila fokusirana na pojedine pojmove. Kao što je prikazano u poglavlju 4.3. funkcionalnost provedbe testa znanja je implementirana u sustav DITUS. Nastavnik priprema test i odabire (ispitna) pitanja koja želi uvrstiti u test, nakon čega test može učiniti dostupan korisnicima. Cjeloviti testovi za obje domene prikazani su u Prilogu 1, a rezultati u poglavlju 7.1.

6.4 Provjera zadovoljstva korisnika

Zadovoljstvo korisnika sustavom analizirano je temeljem anketnog upitnika. Anketni upitnik sastojao se od 61 pitanja podjeljenih u 9 skupina. Anketni upitnik za članove kontrolne skupine imao je jednu skupinu (7 pitanja) manje – skupinu kojom se provjeravalo zadovoljstvo korisnika ponuđenim sugestijama. Anketni upitnik je prilagođen iz (Slavuj, 2013) odnosno iz originalnog DeLone/McLean modela (Delone, 2003).

Isti anketni upitnik je korišten u obje domene. Cjeloviti anketni upitnik prikazan je u Prilogu 2., dok su cjeloviti rezultati anketa za obje domene nalaze u Prilozima 3 i 4.

6.5 Analiza učinkovitosti sustava analizom prikupljenih podataka

Ova kategorija vrednuje učinkovitost učenja u sustavu podržanom dubinskom analizom podataka statističkom analizom prikupljenih podataka iz obje skupine, za obje domene. Postavljene su sljedeće skupine kriterija usporedbe rezultata kontrolne i eksperimentalne skupine:

- Aktivnosti
    - broj učenja
    - broj ponavljanja
- Vrijeme učenja
    - prosječno vrijeme učenja.
- Krivulje učenja
- Analiza utjecaja korištenja sugestija na kvalitetu procesa učenja

Krivulje učenja odnosno performansi učenja su dvodimenzionalna vizualizacija napretka studenta tijekom učenja nekog pojma, nastavne jedinice ili vještine. Pritom se na X osi pojavljuju prilike (pitanja) dok se na Y osi pojavljuje napredak u znanju. Krivulje učenja u stvarnosti predstavljaju krivulje performansi odnosno učinkovitosti studenta. „Prave“ krivulje učenja, nastale su nakon što se taj termin ustalio u području i nazivaju se „*moment-by-moment learning graphs*“ (MMLG) (Baker, et al., 2011). S obzirom da ne bi bilo korisno niti pregledno vizualizirati MMLG za svaki pojam u domeni za svakog studenta odabrane su krivulje učenja kao metoda vizualne analize kvalitete odnosno dinamike procesa učenja kod kontrolne i eksperimentalne skupine.

Kao metoda konačnog dokazivanja ili odbacivanja druge hipoteze postavljene u ovom istraživanju korištena je analiza korištenja sugestija koje je korisnicima prikazivao poboljšani tutorski model sustava. Cilj je bio dokazati povezanost (korelaciju) korištenja sugestija i kvalitete procesa učenja, odnosno broja aktivnosti potrebnih za savladavanje domene znanja.

7 Vrednovanje sustava

7.1 Vrednovanje djelotvornosti sustava provjerama znanja

S obzirom da je cilj ovog sustava poboljšanje učinkovitosti procesa učenja, ne očekuje se značajna razlika u rezultatima testova znanja kontrolne i eksperimentalne skupine. Naime, sustav prezentira jednake nastavne materijale svim studentima (članovima kontrolne i eksperimentalne skupine), kao i jednaka pitanja za ponavljanje i jednaka inicijalna pitanja, dok je jedina razlika u funkcionalnosti sustava prikaz sugestija studentima u eksperimentalnoj skupini.

Kao što je navedeno u poglavlju 6.5, studenti su, po istjeku vremena predviđenog za savladavanje domene, pristupili testu znanja koji se sastojao od 30 pitanja višestrukog odabira, u trajanju od 25 minuta. Pitanja nisu bila vezana uz pojedine pojmove u domeni, već su pokrivala više pojmova odjednom, odnosno testirala razumijevanje cjelokupne domene i odnosa između pojedinih pojmova.

Rezultati testova znanja su prikazani u sljedeća dva podpoglavlja.

7.1.1 Domena „Osnovne web tehnologije“

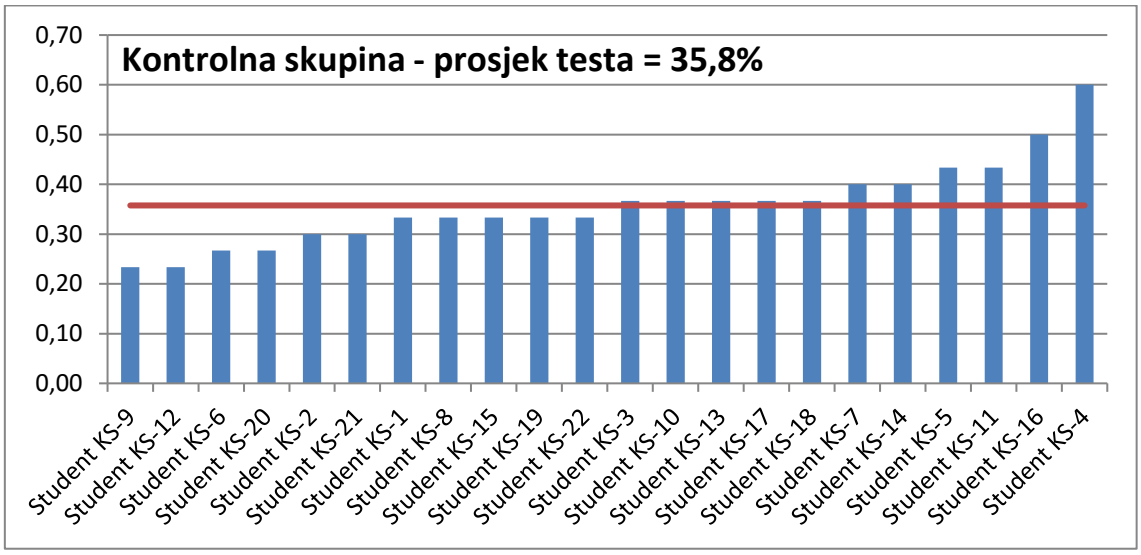
Test znanja u kontrolnoj skupini ispunilo je 22 studenata od 31 odnosno 70,9%, dok je u eksperimentalnoj skupini test ispunilo 27 od 30 odnosno 90%. Rezultati pojedinih studenata prikazani su u Tablici 20. Na kraju tablice prikazana je aritmetička sredina (AS) i standardna devijacija (SD) dobivenih rezultata.

Tablica 20. Rezultati testa znanja za KS i ES u domeni „Osnovne web tehnologije“

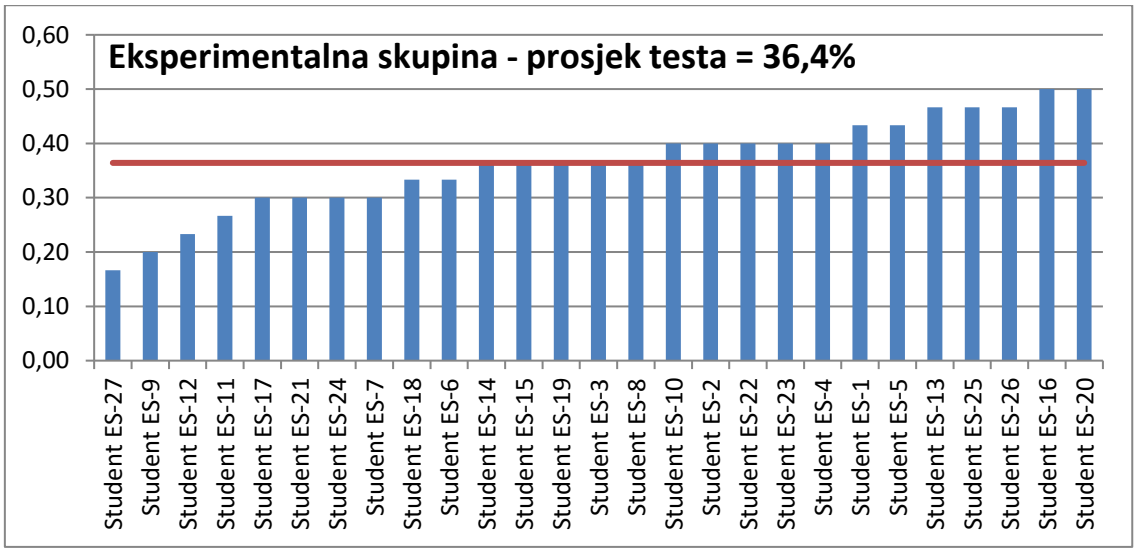
	KONTROLNA SKUPINA		EKSPERIMENTALNA SKUPINA	
1	Student KS-1	0,33	Student ES-1	0,43
2	Student KS-2	0,30	Student ES-2	0,40
3	Student KS-3	0,37	Student ES-3	0,37
4	Student KS-4	0,60	Student ES-4	0,40
5	Student KS-5	0,43	Student ES-5	0,43
6	Student KS-6	0,27	Student ES-6	0,33
7	Student KS-7	0,40	Student ES-7	0,30
8	Student KS-8	0,33	Student ES-8	0,37
9	Student KS-9	0,23	Student ES-9	0,20
10	Student KS-10	0,37	Student ES-10	0,40
11	Student KS-11	0,43	Student ES-11	0,27
12	Student KS-12	0,23	Student ES-12	0,23
13	Student KS-13	0,37	Student ES-13	0,47
14	Student KS-14	0,40	Student ES-14	0,37
15	Student KS-15	0,33	Student ES-15	0,37
16	Student KS-16	0,50	Student ES-16	0,50
17	Student KS-17	0,37	Student ES-17	0,30
18	Student KS-18	0,37	Student ES-18	0,33
19	Student KS-19	0,33	Student ES-19	0,37
20	Student KS-20	0,27	Student ES-20	0,50
21	Student KS-21	0,30	Student ES-21	0,30
22	Student KS-22	0,33	Student ES-22	0,40
23			Student ES-23	0,40
24			Student ES-24	0,30
25			Student ES-25	0,47
26			Student ES-26	0,47
27			Student ES-27	0,17
	AS	**0,358**	**AS**	**0,364**
	STD	0,083	STD	0,085

Iz rezultata je vidljiva malo viša prosječna vrijednost rezultata eksperimentalne skupine.

Rezultati su (poredani prema uspješnosti) grafički prikazani na Slikama 67 i 68



Slika 67. Rezultati testa znanja KS za domenu „Osnovne web tehnologije“



Slika 68. Rezultati testa znanja ES za domenu „Osnovne web tehnologije“

Iz Slika 67 i 68 je vidljivo da je veći broj studenata eksperimentalne skupine imalo rezultat jednak ili viši od prosjeka (17 od 27 = 62,9%), u odnosu na studente iz kontrolne skupine (11 od 22 = 50%), gdje su rezultati dva studenta podigli prosjek za 7%.

Eksperimentalna skupina u konačnici je postigla <1% bolji prosječni rezultat testa znanja.

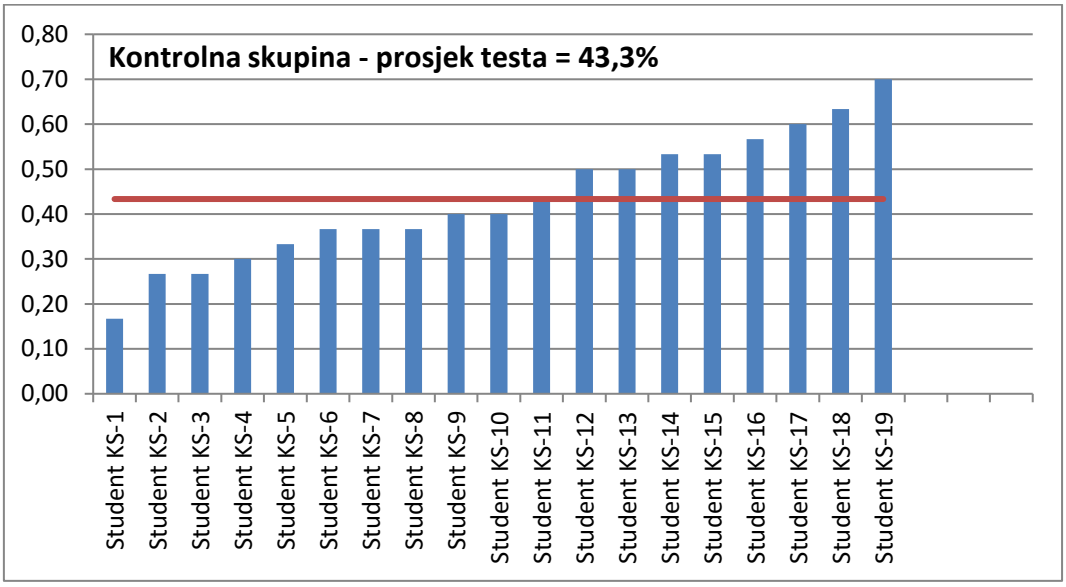
7.1.2 Domena „Uzorci dizajna“

Test znanja u kontrolnoj skupini ispunilo je 19 studenata od 33 odnosno 57,5%, dok je u eksperimentalnoj skupini test ispunilo 26 od 34 odnosno 76,5%. Rezultati pojedinih studenata prikazani su u Tablici 21.

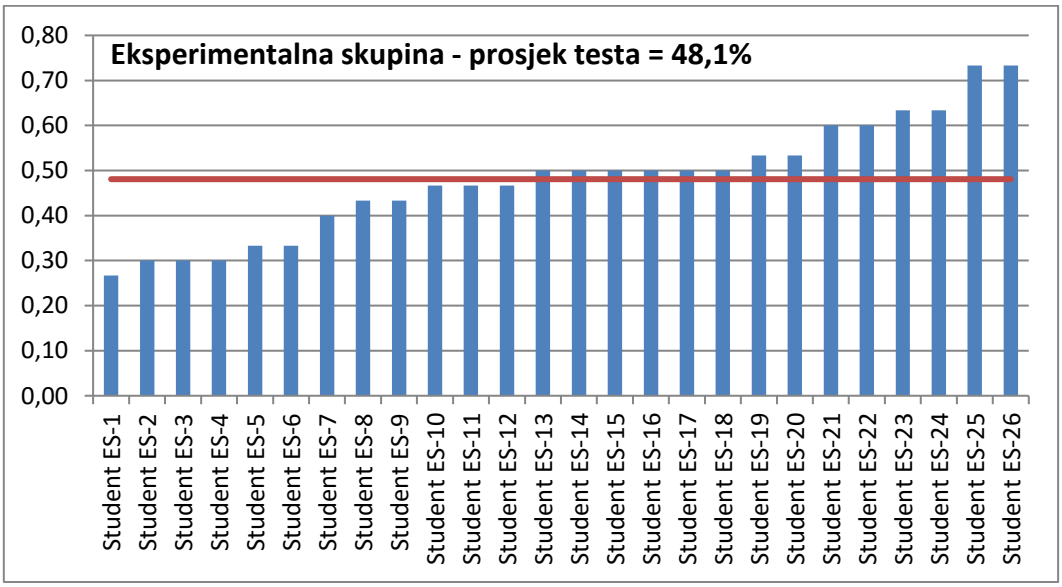
Tablica 21. Rezultati testa znanja za KS i ES u domeni „Uzorci dizajna“

	KONTROLNA SKUPINA		EKSPERIMENTALNA SKUPINA	
1	Student KS-1	0,17	Student ES-1	0,27
2	Student KS-2	0,27	Student ES-2	0,30
3	Student KS-3	0,27	Student ES-3	0,30
4	Student KS-4	0,30	Student ES-4	0,30
5	Student KS-5	0,33	Student ES-5	0,33
6	Student KS-6	0,37	Student ES-6	0,33
7	Student KS-7	0,37	Student ES-7	0,40
8	Student KS-8	0,37	Student ES-8	0,43
9	Student KS-9	0,40	Student ES-9	0,43
10	Student KS-10	0,40	Student ES-10	0,47
11	Student KS-11	0,43	Student ES-11	0,47
12	Student KS-12	0,50	Student ES-12	0,47
13	Student KS-13	0,50	Student ES-13	0,50
14	Student KS-14	0,53	Student ES-14	0,50
15	Student KS-15	0,53	Student ES-15	0,50
16	Student KS-16	0,57	Student ES-16	0,50
17	Student KS-17	0,60	Student ES-17	0,50
18	Student KS-18	0,63	Student ES-18	0,50
19	Student KS-19	0,70	Student ES-19	0,53
20			Student ES-20	0,53
21			Student ES-21	0,60
22			Student ES-22	0,60
23			Student ES-23	0,63
24			Student ES-24	0,63
25			Student ES-25	0,73
26			Student ES-26	0,73
	AS	**0,433**	**AS**	**0,481**
	STD	0,141	STD	0,125

Iz rezultata je vidljiva nešto viša prosječna vrijednost rezultata eksperimentalne skupine. Rezultati su (poredani prema uspješnosti) grafički prikazani na Slikama 69 i 70



Slika 69. Rezultati testa znanja KS za domenu „Uzorci dizajna“



Slika 70. Rezultati testa znanja ES za domenu „Uzorci dizajna“

Rezultati prikazani na Slikama 69 i 70 pokazuju nešto ujednačenije rezultate (iako teže usporedive zbog gotovo 30% manjeg broja rezultata) u odnosu na eksperimentalnu skupinu. U eksperimentalnoj skupini se mogu uočiti skupine sa slabijim, prosječnim i nadprosječnim rezultatima testa znanja. Također je vidljivo da je veći broj studenata eksperimentalne skupine

imalo rezultat jednak ili viši od prosjeka (14 od 26 = 53,8%), u odnosu na studente iz kontrolne skupine (9 od 19 = 47,4%%).

Eksperimentalna skupina u konačnici je postigla 4,7% bolji prosječni rezultat testa znanja.

Članovi eksperimentalne skupine u obje domene znanja ostvarili su bolji rezultat od studenata kontrolne skupine. Međutim, test statističke značajnosti (t-test) nije pokazao statistički značajnu razliku između skupina. Završnom testu znanja u obje domene pristupio je puno veći postotak studenata iz eksperimentalne skupine. Taj podatak nije moguće jednoznačno interpretirati, ali može imati značenje veće motiviranosti ili angažmana studenata, proizašlog iz većeg angažmana (prilagodljivosti) sustava prema korisnicima – nuđenje sugestija.

7.2 Vrednovanje sustava ispitivanjem zadovoljstva korisnika

Kao što je navedeno u poglavlju 6.6, zadovoljstvo korisnika evaluirano je anketom koja se sastojala od 61 pitanja. Osim uvodnog dijela ankete u kojem su studenti unosili opće informacije (dob, spol, smjer studija), te posljednjeg dijela, koji se sastojao od jednog pitanja otvorenog tipa (komentari, prijedlozi i ideje za poboljšanje sustava), većina pitanja nudila je studentu odabir između vrijednosti od 1 (nikako se ne slažem) do 5 (u potpunosti se slažem). U ovom poglavlju prezentirani su samo ključni rezultati anketa odnosno različitosti između kontrolne i eksperimentalne skupine dok su cjeloviti rezultati anketa dani u Prilogu 3 (domena „Osnovne web tehnologije“) i Prilogu 5 (domena „Uzorci dizajna“). Studentima kontrolne skupine nisu prezentirane sugestije, pa im ta skupina pitanja nije prikazana u anketi. Predzadnji stupac tablice (osim za kategoriju „Opće informacije“) prikazuje aritmetičku sredinu (AS) odnosno prosječnu ocjenu pitanja u skupini. Zadnji stupac prikazuje standardnu devijaciju (SD) ocjena u skupini.

7.2.1 Domena „Osnovne web tehnologije“

U kontrolnoj skupini anketu je ispunilo 22 od 31 studenata, odnosno 70,9%, dok je u eksperimentalnoj skupini anketu ispunilo 28 od 30 studenata, odnosno 93,3%.

Tablica 22. Ankete – OPĆE INFORMACIJE (domena „Osnovne web tehnologije“)

	KONTROLNA SKUPINA						EKSPERIMENTALNA SKUPINA					
Broj anketa	22						28					
OPĆE INFORMACIJE												
Spol	M	9	41%	Ž	13	59%	M	12	43%	Ž	16	57%
Starost	M	0		Ž	0		M	0		Ž	0	
Razina studija (preddiplomski/diplomski)	PD	22	100%	D	0	0%	PD	28	100%	D	0	
Studijska grupa	1P	16	73%	2P	6	27%	1P	24	86%	2P	4	14%

Iz Tablice 22 vidljivo je da je u obje skupine podjednak omjer studenata muškog i ženskog spola, te broj studenata dvopredmetnog studija (od onih koji su ispunili anketu).

U gotovo svim kategorijama studenti kontrolne skupine imali su nešto više prosječne ocjene sustava od studenata eksperimentalne skupine. Zbog malog uzorka, ocjene jednog studenta mogu uzrokovati promjene u prosječnoj ocjeni grupe. Kontrolna skupina imala je višu prosječnu ocjenu i za pitanja o prilagodljivosti sustava korisniku (Tablica 23) što se može interpretirati nedovoljno dobrom prezentacijom sugestija, odnosno nedovoljno dobrom objašnjenju njihove svrhe. Pri razvoju sučelja, odnosno planiranju prezentacije sugestija odabran je minimalistički pristup uz pretpostavku da će kratka poruka i poveznica na sugerirani pojam biti samorazumljivi, te minimalno utjecati na tijek učenja. Na temelju ovih rezultata može se zaključiti da je sugestije trebalo vizualno više istaknuti te bolje obrazložiti prednosti korištenja sugestija studentima eksperimentalne skupine.

Tablica 23. Ankete - FLEKSIBILNOST (domena „Osnovne web tehnologije“)

	KONTROLNA SKUPINA							EKSPERIMENTALNA SKUPINA						
FLEKSIBILNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
Sustav omogućuje jednake prilike za korisnike s različitim razinama predznanja:	0%	0%	18%	50%	32%	4,14	0,69	4%	4%	18%	36%	39%	4,04	1,02
Sustav omogućuje poveznice i rad s materijalima izvan sustava:	0%	0%	0%	50%	50%	4,50	0,50	4%	4%	14%	36%	43%	4,11	1,01
Sustav je prilagođen i naprednim korisnicima jer ima i materijale za naprednije:	0%	5%	23%	41%	32%	4,00	0,85	4%	7%	43%	25%	21%	3,54	1,02
Sadržaj u sustavu prilagođava se mojim potrebama:	0%	9%	18%	55%	18%	3,82	0,83	0%	11%	29%	36%	25%	3,75	0,95
Sustav mi predlaže tijek učenja ovisno o mojem znanju:	5%	0%	27%	45%	23%	3,82	0,94	0%	11%	32%	36%	21%	3,68	0,93

Studenti kontrolne skupine su također bolje ocjenili svoje iskustvo korišenja sustava. Visoku ocjenu potrebi korištenja ovog sustava i u drugim kolegijima dali su studenti obje skupine, što govori o pozitivnom općem stavu studenata prema primjeni novih tehnologija u nastavi, kao i prihvaćanju tutorskog modela e-učenja (Tablica 24).

Tablica 24. Anketa – ISKUSTVO (domena „Osnovne web tehnologije“)

	KONTROLNA SKUPINA							EKSPERIMENTALNA SKUPINA						
ISKUSTVO	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
Sviđa mi se koristiti sustav:	0%	5%	5%	45%	45%	4,32	0,76	7%	7%	7%	32%	46%	4,04	1,21
Moja iskustva u radu sa sustavom su pozitivna:	0%	0%	5%	50%	45%	4,41	0,58	0%	11%	18%	32%	39%	4,00	1,00
Rad sa sustavom je koristan za usvajanje znanja iz kolegija:	0%	0%	23%	23%	55%	4,32	0,82	0%	7%	29%	25%	39%	3,96	0,98
Sustav bi se trebao koristiti i u sklopu drugih kolegija:	0%	5%	5%	9%	82%	4,68	0,76	0%	4%	29%	18%	50%	4,14	0,95
Koristeći sustav brže usvajam gradivo:	0%	0%	27%	23%	50%	4,23	0,85	4%	0%	36%	25%	36%	3,89	1,01
Lakše učim koristeći sustav nego učeći na klasičan način:	5%	9%	5%	32%	50%	4,14	1,14	4%	7%	18%	43%	29%	3,86	1,03

Studenti eksperimentalne skupine odgovarali su na skup od 7 pitanja o sugestijama i njihovoj implementaciji u sustav. Iz rezultata prikazanih u Tablici 25, vidljiva je potreba daljnjeg rada na prezentaciji sugestija, ali studenti ih smatraju korisnim mehanizmom, iako ne utječe presudno na odluku o tijeku procesa učenja.

Značajna je prosječna ocjena smislenosti ponuđenih pojmova što potvrđuje ideju primjene dubinske analize podataka, a posebno otkrivanja čestih i učinkovitih putanja za kreiranje sugestija. Studenti su sugestije „što učiti PRIJE“ odabranog pojma ocjenili nešto višom ocjenom, od sugestija „što učiti POSLIJE“, što govori o utjecaju sugestija na strategiju studenta za savladavanja domene odnosno pojma.

Ocjena nije u skladu s objektivnim podacima dobivenim analizom zapisa u bazi podataka o korištenju sugestija (Tablice 15 i 16) iz kojih je vidljivo da studenti nisu koristili sugestije tipa PREFIX.

Studenti kontrolne skupine unijeli su 7, dok su studenti eksperimentalne skupine unijeli 12 komentara/prijedloga i ideja za poboljšanje sustava. Svi komentari bili su pozitivno intonirani, uz isticanje potrebe primjene sustava u drugim kolegijima, za pripremu kolokvija, te otklanjanje manjih nedostataka.

Tablica 25. Anketa – SUGESTIJE (domena „Osnovne web tehnologije“)

SUGESTIJE	1	2	3	4	5	AS	SD
Zadovoljni ste načinom na koji su prezentirane sugestije (vizualno):	0%	4%	36%	46%	14%	3,71	0,75
Općenito govoreći, sugestije su koristan mehanizam za pomoć pri odabiru pojmova za učenje/ponavljanje:	0%	0%	39%	50%	11%	3,71	0,65
Sugestije su imale pozitivan utjecaj na Vaš proces učenja (vrijeme/brzina savladavanja pojmova):	4%	18%	54%	25%	0%	3,00	0,76
Sugestije su utjecale na moju odluku o odabiru sljedećeg pojma (koristili ste ih/niste ih koristili, tj. klikali):	11%	14%	32%	36%	7%	3,14	1,09
Pojmovi ponuđeni u sugestijama činili su vam se smislenima/logičnima u odnosnu na pojam na kojem ste se trenutno nalazili i trenutnu savladanost domene:	0%	4%	32%	57%	7%	3,68	0,66
Sugestije "Što učiti PRIJE" odabranog pojma bile su korisne:	0%	11%	43%	39%	7%	3,43	0,78
Sugestije "Što učiti POSLIJE" odabranog pojma bile su korisne:	4%	11%	46%	29%	11%	3,32	0,93

Ukupni rezultati ankete za domenu „Osnovne web tehnologije“ prikazani su u Prilogu 3.

7.2.2 Domena „Uzorci dizajna“

U kontrolnoj skupini anketu je ispunilo 19 od 34 studenata odnosno 57,5%, dok je u eksperimentalnoj skupini test ispunilo 28 od 33 odnosno 82,3%. Kao što je vidljivo iz Tablice 26, u obje skupine je podjednak omjer studenata muškog i ženskog spola, te broj studenata dvopredmetnog studija (od onih koji su ispunili anketu).

Tablica 26. Anketa – OPĆE INFORMACIJE (domena „Uzorci dizajna“)

	KONTROLNA SKUPINA						EKSPERIMENTALNA SKUPINA					
Broj anketa	19						28					
OPĆE INFORMACIJE												
Spol	M	9	47%	Ž	10	53%	M	13	46%	Ž	15	54%
Starost	M	21		Ž	21		M	22		Ž	21	
Razina studija (preddiplomski/diplomski)	P	19		D	0		P	28		D	0	
Studijska grupa	1P	15		2P	4		1P	25		2P	3	

U rezultatima ankete za ovu domenu primjetna je puno veća uravnoteženost u prosječnim ocjenama po kategorijama i pitanjima kod kontrolne i eksperimentalne skupine, za razliku od prethodne domene.

Tablica 27. Anketa – FLEKSIBILNOST (domena „Uzorci dizajna“)

	KONTROLNA SKUPINA							EKSPERIMENTALNA SKUPINA						
FLEKSIBILNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
Sustav omogućuje jednake prilike za korisnike s različitim razinama predznanja:	0%	5%	16%	32%	47%	4,17	0,90	4%	7%	14%	36%	39%	4,11	1,10
Sustav omogućuje poveznice i rad s materijalima izvan sustava:	5%	0%	16%	26%	53%	4,17	1,07	0%	18%	7%	29%	46%	4,00	1,11
Sustav je prilagođen i naprednim korisnicima jer ima i materijale za naprednije:	0%	5%	16%	26%	53%	4,28	0,93	0%	4%	25%	36%	36%	4,00	0,88
Sadržaj u sustavu prilagođava se mojim potrebama:	0%	0%	21%	47%	32%	4,11	0,74	0%	11%	21%	39%	29%	4,00	0,94
Sustav mi predlaže tijek učenja ovisno o mojem znanju:	0%	16%	11%	42%	32%	3,89	1,05	4%	18%	18%	29%	32%	3,89	1,15

U kategorijama FLEKSIBILNOST (Tablica 27) i ISKUSTVO studenti kontrolne skupine imali su ili nešto više ili jednake prosječne ocjene što učvršćuje zaključak da je potreban daljnji rad na prezentaciji i razumijevanju svrhe sugestija

Tablica 28. Ankete – SUGESTIJE (domena „Uzorci dizajna“)

SUGESTIJE	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
Zadovoljni ste načinom na koji su prezentirane sugestije (vizualno):	4%	4%	36%	29%	29%	3,72	1,10
Općenito govoreći, sugestije su koristan mehanizam za pomoć pri odabiru pojmova za učenje/ponavljanje:	0%	7%	32%	36%	25%	4,06	0,78
Sugestije su imale pozitivan utjecaj na Vaš proces učenja (vrijeme/brzina savladavanja pojmova):	7%	18%	36%	21%	18%	3,61	1,11
Sugestije su utjecale na moju odluku o odabiru sljedećeg pojma (koristili ste ih/niste ih koristili, tj. klikali):	11%	32%	29%	7%	21%	3,22	1,18
Pojmovi ponuđeni u sugestijama činili su vam se smislenima/logičnima u odnosnu na pojam na kojem ste se trenutno nalazili i trenutnu savladanost domene:	0%	14%	39%	29%	18%	3,67	0,88
Sugestije "Što učiti PRIJE" odabranog pojma bile su korisne:	0%	14%	46%	11%	29%	3,83	1,01
Sugestije "Što učiti POSLIJE" odabranog pojma bile su korisne:	0%	7%	54%	21%	18%	3,61	0,89

. Rezultati eksperimentalne skupine na pitanja o sugestijama prikazani u Tablici 28 pokazuju nešto više prosječne ocjene u odnosu na prethodnu domenu, što je značajno jer domena „Uzorci dizajna“ sadrži pojmove koje studenti ne obrađuju na niti jednom drugom kolegiju, što je čini složenijom za savladavanje.
Studenti su ponovno procjenili sugestije tipa „što učiti PRIJE“ korisnijima od sugestija tipa „što učiti POSLIJE“ što također nije u skladu s objektivno prikupljenim podacima iz baze podataka iz kojih je vidljivo da sugestije tipa PREFIX studenti nisu koristili.

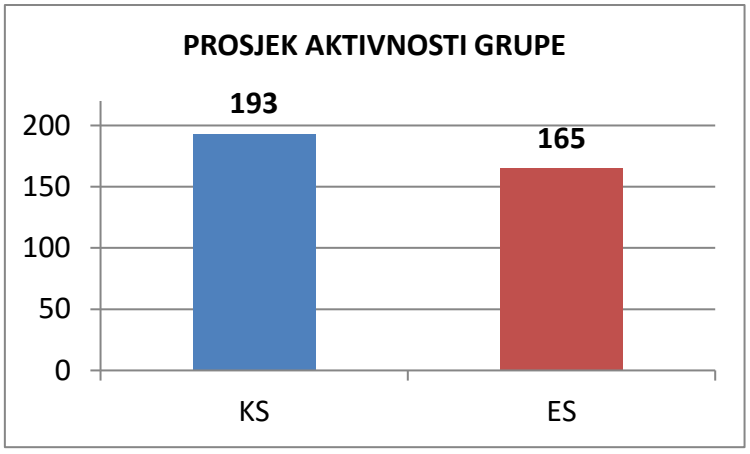
Studenti kontrolne skupine unijeli su 7, dok su studenti eksperimentalne skupine unijeli 8 komentara/prijedloga i ideja za poboljšanje sustava. Svi komentari bili su pozitivno intonirani, uz isticanje potrebe primjene sustava u drugim kolegijima, te otklanjanje manjih nedostataka.
Ukupni rezultati ankete za domenu „Osnovne web tehnologije“ prikazani su u Prilogu 5.

7.3 Vrednovanje učinkovitosti sustava

Iako je evaluacija sustava na temelju testova znanja dala pozitivne rezultate kod eksperimentalnih skupina za obje domene, s obzirom na mali broj ispitanika, nije ih moguće potvrditi kao statistički značajne. S obzirom na primarni cilj istraživanja koji ističe poboljšanje učinkovitosti sustava odnosno procesa učenja, sljedeći dio evaluacije rada sustava je od najveće važnosti za potvrdu druge hipoteze. U ovom dijelu analiziraju se objektivni podaci o aktivnostima korisnika prikupljani tijekom njihove interakcije sa sustavom, odnosno učenja/savladavanja domena znanja.

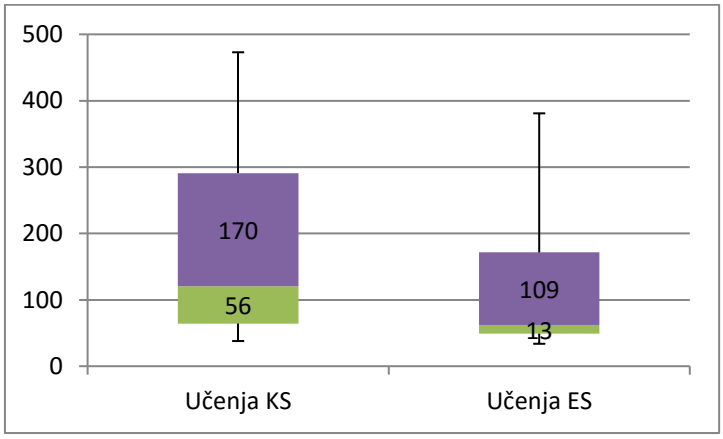
7.3.1 Domena „Osnovne web tehnologije“

Primjenom metoda deskriptivne statistike na konačan skup podataka studenata obje skupine dobiveni su rezultati prikazani na Slikama 71, 73 i 74. Kako je broj aktivnih studenata, odnosno studenata koji su u potpunosti savladali domenu znanja različith u kontrolnoj i eksperimentalnoj skupini, za te studente vrijednosti su transponirane u odnosu na prosjek skupine i postotak savladanosti domene znanja. Na temelju tih podataka kreiran je grafikon prikazan na slici 71, koji prikazuje prosječan broj aktivnosti (učenja ili provjere) potrebnih za savladavanje domene u kontrolnoj (KS) i eksperimentalnoj skupini (ES). Iz grafikona je vidljivo da je studentima u eksperimentalnoj skupini trebalo manje aktivnosti (14,7%) za savladavanje domene.



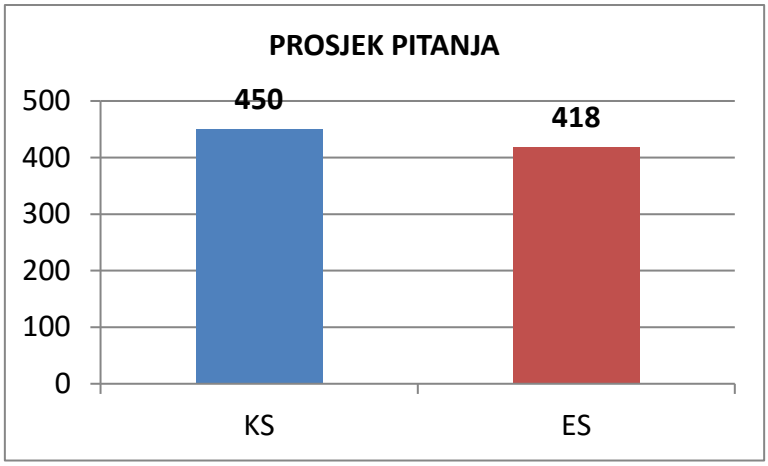
Slika 71. Prosjek aktivnosti grupe u domeni "Osnovne web tehnologije"

Na slici 72 prikazan je „*box and whisker*“ grafikon iz kojeg je vidljivo da su članovi eksperimentalne skupine imali manji ukupni raspon broja učenja, te puno manji broj učenja za 75% skupine.



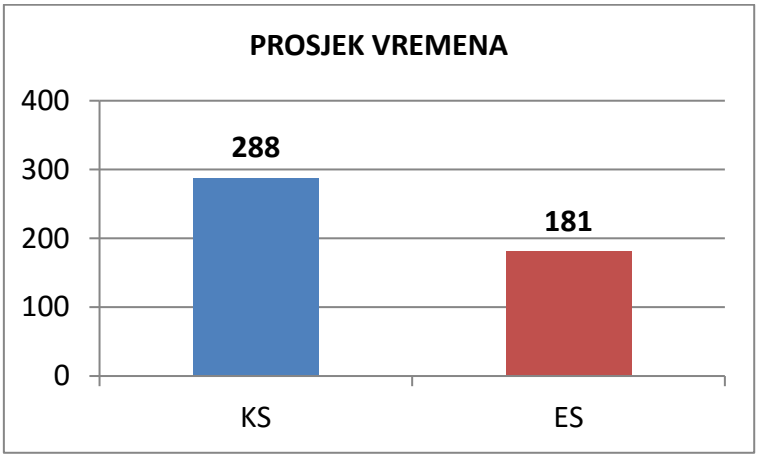
Slika 72. Usporedba aktivnosti grupe u domeni „Osnovne web tehnologije“

Sljedeća karakteristika kojom su uspoređene kontrolna i eksperimentalna skupina je prosječan broj postavljenih pitanja. Vidljivo je da su studenti u eksperimentalnoj skupini imali za 7% niži prosječan broj postavljenih pitanja. Iako bi, prema prethodnom grafikonu, bilo moguće očekivati veću razliku, podaci pokazuju da su studenti eksperimentalne skupine imali veći broj pitanja postavljenih tijekom ponavljanja, dok su studenti kontrolne skupine veći broj pitanja imali tijekom učenja (inicijalna pitanja), što je proces koji znatno sporije dopridonosi ukupnom savladavanju domene znanja.

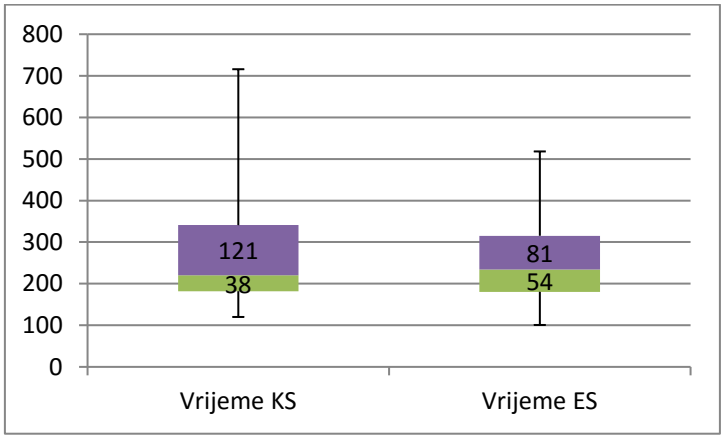


Slika 73. Prosjek broja postavljenih pitanja grupe u domeni "Osnovne web tehnologije"

Konačno, skupine su uspoređene prema prosječnom vremenu potrebnom za savladavanje domene znanja. Na slici 75 vidljivo je da su studenti eksperimentalne skupine trebali manje vremena (18,9%) za savladavanje domene, što se bilo moguće očekivati na temelju podataka sa slika 71 i 73.



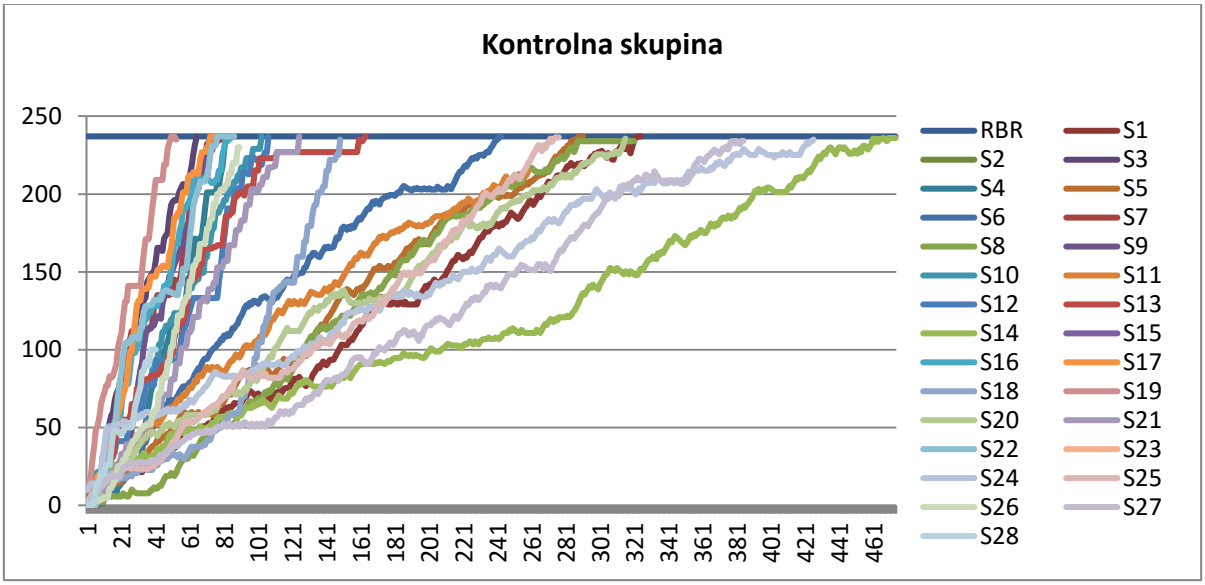
Slika 74. Prosjek vremena učenja grupe u domeni "Osnovne web tehnologije"



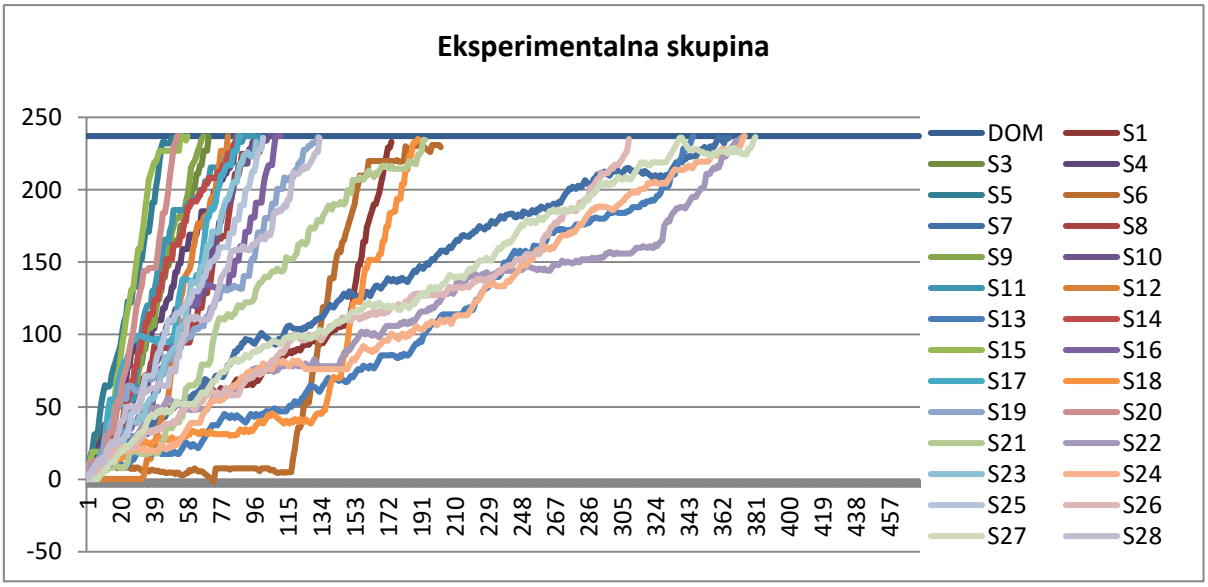
Slika 75. Usporedba vremena učenja grupe u domeni „Osnovne web tehnologije“

Studenti eksperimentalne skupine uravnoteženije su koristili funkcionalnosti učenja i ponavljanja, čime su, uz gotovo jednak broj postavljenih pitanja, brže dolazili do potpune savladanosti domene što je vidljivo iz Slike 75.

Sljedeća analiza prikazuje krivulje učenja kontrolne (Slika 76) i eksperimentalne (Slika 77) skupine za domenu „Osnovne web tehnologije“. Plava vodoravna linija pri vrhu grafikona označava sumu pragova svih pojmova u domeni, odnosno ukupan broj bodova koje studenti moraju sakupiti da bi domena bila u potpunosti savladana. Širina grafikona određena je maksimalnim brojem postavljenih pitanja iz obje skupine (kreiran od strane studenta u kontrolnoj skupini) koji iznosi 457.



Slika 76. Krivulje učenja kontrolne skupine u domeni "Osnovne web tehnologije"



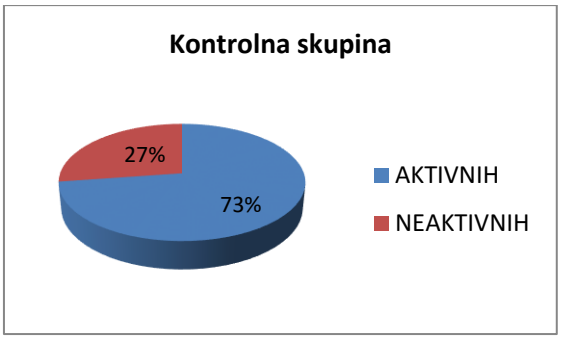
Slika 77. Krivulje učenja eksperimentalne skupine u domeni "Osnovne web tehnologije"

Vizualnom usporedbom grafikona uočeno je sljedeće:

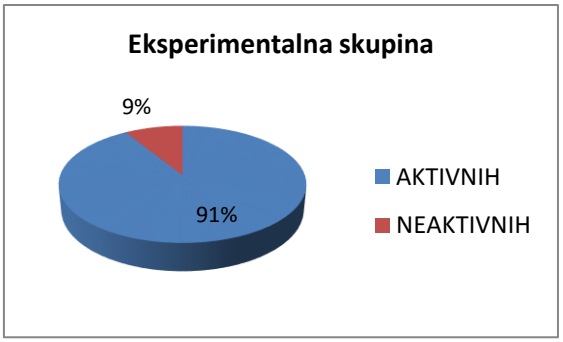
- najveći broj postavljenih pitanja (promjena u matrici znanja) kod studenata kontrolne skupine je 475, dok je kod eksperimentalne skupine 381, odnosno manji je za 94 pitanja (aktivnosti),
- kod eksperimentalne skupine uočavaju se tri podskupine – a) studenti s bržim rastom stanja u matrici znanja (tu skupinu čini više studenata nego kod kontrolne skupine), b) studenti sa sporim rastom stanja u matrici znanja (studenti koji su se gotovo isključivo fokusirali na korištenje funkcionalnosti Učenja), te c) skupinu koja je imala vrlo spor rast stanja, koji je nakon određenog perioda prešao u nagli rast.
- kod kontrolne skupine vidljiva je kontinuirana distribucija krivulja učenja od studenata s najbržim rastom do onih s najsporijim, ali je moguće uočiti dvije skupine studenata (brži i sporiji).
- nedostatak ove vizualizacije je izostanak vremenske linije koja bi olakšala praćenje promjene stanja u vremenu, te identifikaciju studenata koji su i iznimno kratkom periodu došli do potpune savladanosti domene.

7.3.2 Domena „Uzorci dizajna“

Domena „Uzorci dizajna“ predstavlja nove sadržaje s kojima se studenti nisu imali prilike susretati prilikom dosadašnjeg studija, te je očekivano sporije napredovanje studenata kroz domenu u odnosu na prethodnu. Kod kontrolne skupine koja se sastojala od 33 studenta, bilo je gotovo 30% neaktivnih (9 studenata) u odnosu na eksperimentalnu skupinu koja se sastojala od 34 studenta, od kojih je bilo samo troje neaktivnih, što je vizualno prikazano na Slikama 78 i 79.

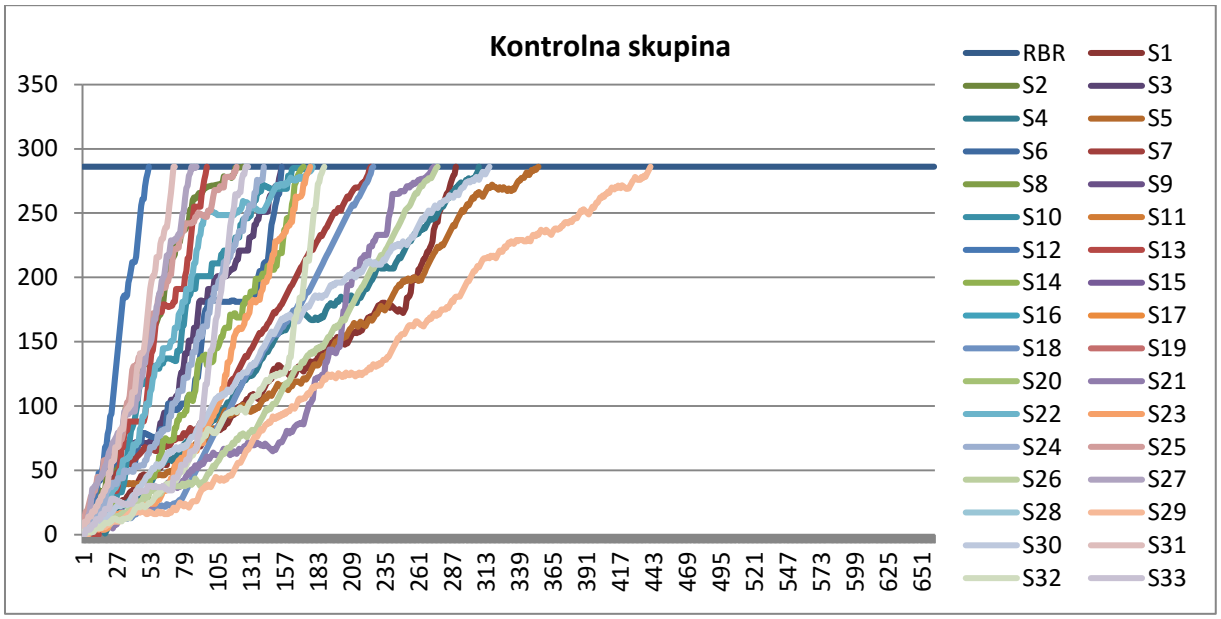


Slika 78. Odnos aktivnih i neaktivnih studenata kontrolne skupine

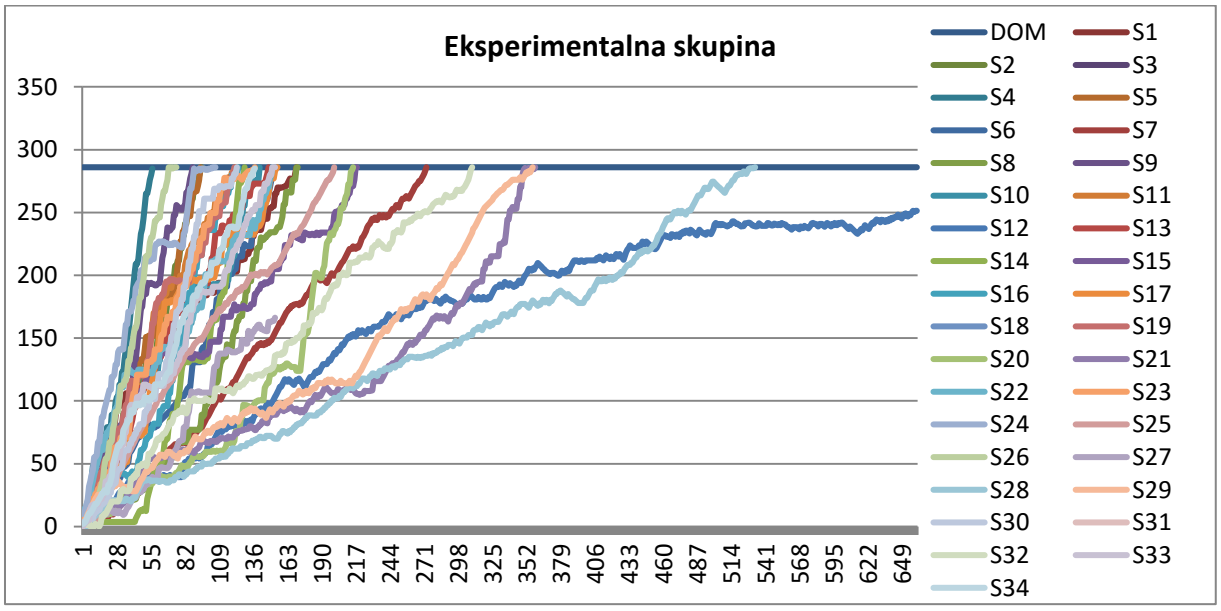


Slika 79. Odnos aktivnih i neaktivnih studenata eksperimentalne skupine

Sljedeća analiza prikazuje krivulje učenja kontrolne i eksperimentalne skupine (Slike 80 i 81).



Slika 80. Krivulje učenja kontrolne skupine u domeni „Uzorci dizajna“



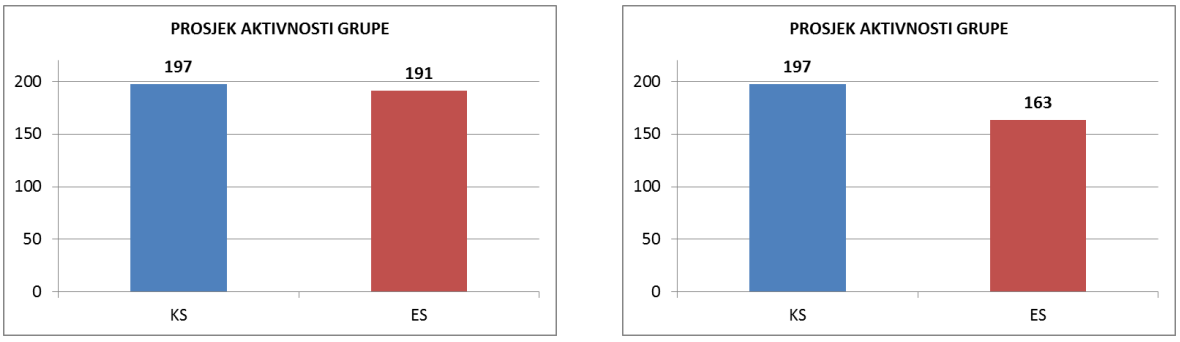
Slika 81. Krivulje učenja eksperimentalne skupine u domeni „Uzorci dizajna“

Plava vodoravna linija pri vrhu grafikona označava sumu pragova svih pojmova u domeni, odnosno ukupan broj bodova koje studenti moraju sakupiti da bi domena bila u potpunosti savladana (286). Širina grafikona određena je maksimalnim brojem postavljenih pitanja iz obje skupine (kreiran od strane studenta u eksperimentalnoj skupini) koji iznosi 651.

Vizualnom usporedbom grafikona uočeno je sljedeće:

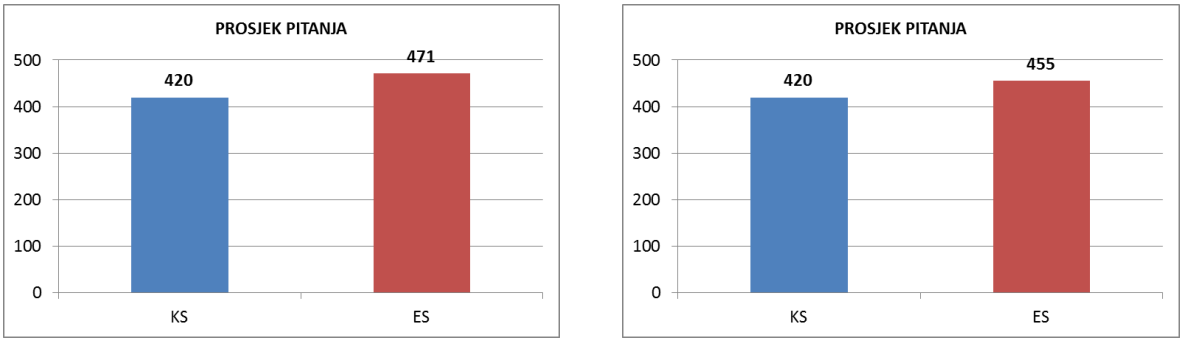
- najveći broj postavljenih pitanja (promjena u matrici znanja) kod studenata kontrolne skupine je 443, dok je kod eksperimentalne skupine 651.
- dva studenta eksperimentalne skupine predstavljaju iznimke (eng. *outlier*) koji značajno podižu prosječne vrijednosti aktivnosti čitave skupine. Broj aktivnosti koje su generirali je gotovo dvostruko veći od svih drugih članova skupine, zbog čega su u analizama koje slijede podaci analizirani sa i bez podataka ovih dvaju studenata.
- kod kontrolne skupine vidljiva je kontinuirana distribucija krivulja učenja od studenata s najbržim rastom do onih s najsporijim, te nije moguće uočiti neke jasno odvojene skupine studenata (bržih ili sporijih).
- kod eksperimentalne skupine uočava se vrlo gusta skupina studenata čije su krivulje „strmije“ od zamišljenog prosjeka kontrolne skupine,
- kod eksperimentalne skupine mogu se uočiti tri podskupine – a) studenti s bržim rastom stanja u matrici znanja (tu skupinu čini većina studenata), b) studenti s nešto sporijim rastom stanja u matrici znanja, te c) dva studenta koja predstavljaju iznimne vrijednosti – ovi studenti koristili su pristup pogađanja točnih odgovora, što je rezultiralo vrlo visokim brojem aktivnosti odnosno postavljenih pitanja. Dakle, osnovna strategija nije bilo usvajanje znanja, već prikupljanje bodova za savladavanje pojmova u što kraćem roku. Prosječno vrijeme odgovora na pitanja kod ovih studenata bilo je vrlo kratko (sekunde), što potvrđuje pretpostavku o pogađanju odgovora.

Slijedi analiza prosječnih vrijednosti aktivnosti učenja, ponavljanja i vremena učenja, te usporedbe cjelovitih raspona vrijednosti između kontrolne i eksperimentalne skupine. U grafikonima prikazanima na lijevoj strani nalaze se prosječne vrijednosti koje uključuju podatke dvaju spomenutih studenata, dok se u grafikonima na desnoj strani nalaze prosječne vrijednosti bez njihovih podataka.



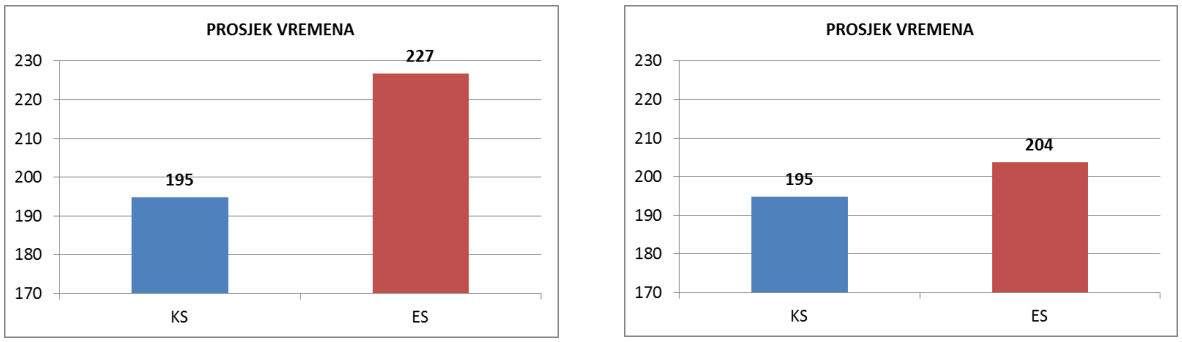
Slika 82. Prosjek aktivnosti učenja sa i bez „outlier“-a

Usporedba prosječnih vrijednosti kontrolne i eksperimentalne skupine pokazuje manji prosječan broj aktivnosti učenja čak i sa uključenim podacima dvaju *outlier*-a (Slika 82).



Slika 83. Prosjek broja pitanja sa i bez „outlier“-a

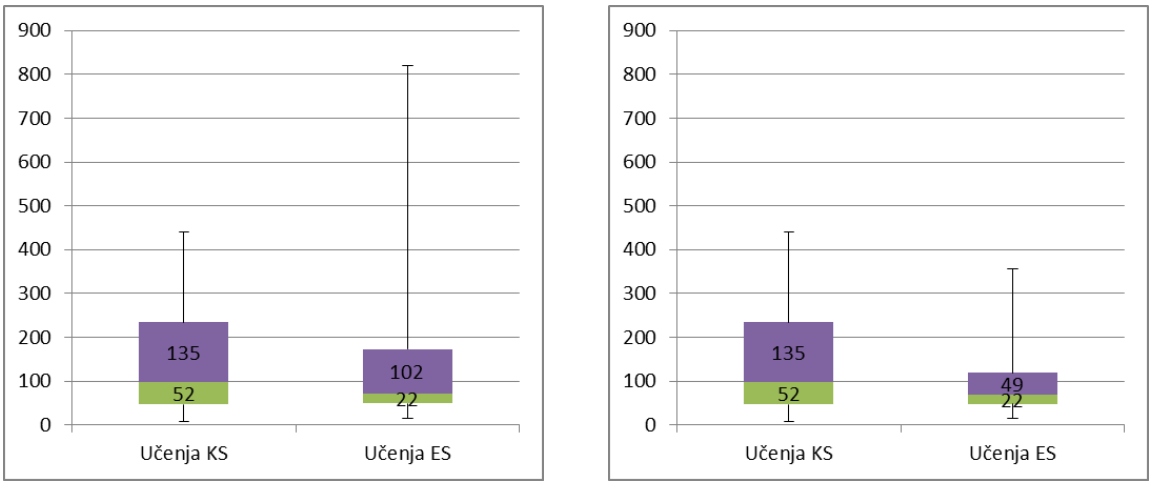
Usporedba prosjeka broja pitanja (na slici 83) i prosjeka vremena učenja (Slika 84) pokazuje nešto veći prosjek kod eksperimentalne skupine u oba slučaja. Na ovaj prosjek utjecao je i veći broj neaktivnih studenata u kontrolnoj skupini.



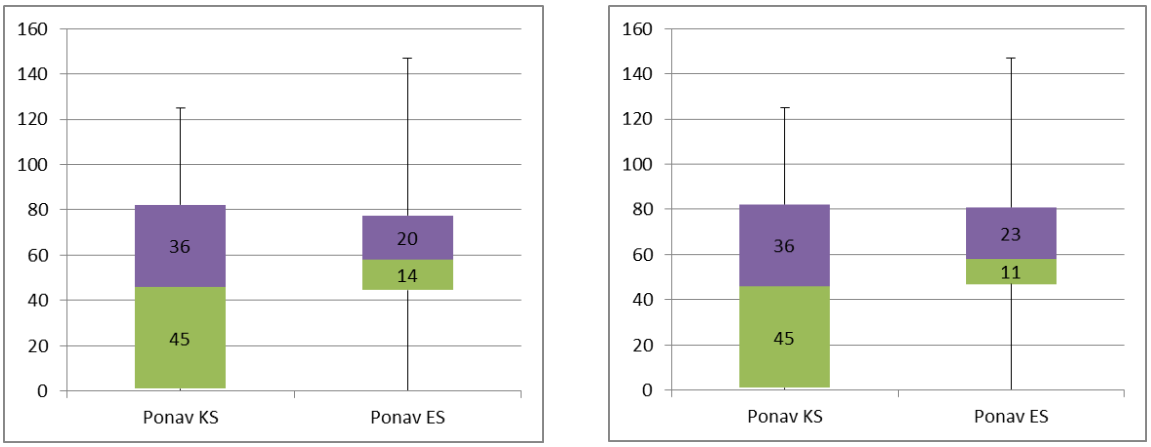
Slika 84. Prosjek vremena učenja sa i bez „outlier“-a

Na sljedećem skupu grafikona (Slike 85, 86 i 87) analizirani su rasponi vrijednosti učenja, ponavljanja i vremena učenja kontrolne i eksperimentalne skupine, sa i bez spomenutih „*outlier*“ studenata. Eksperimentalna skupina više je koristila funkcionalnost ponavljanja, što je očekivano uslijed sugestija koje relativno često nude ponavljanje upravo učenog pojma. Međutim, vrijednosti 75% članova skupine još uvijek je manja od kontrolne skupine. Može se zaključiti da su studenti kontrolne skupine trebali manje aktivnosti za savladavanje domene,

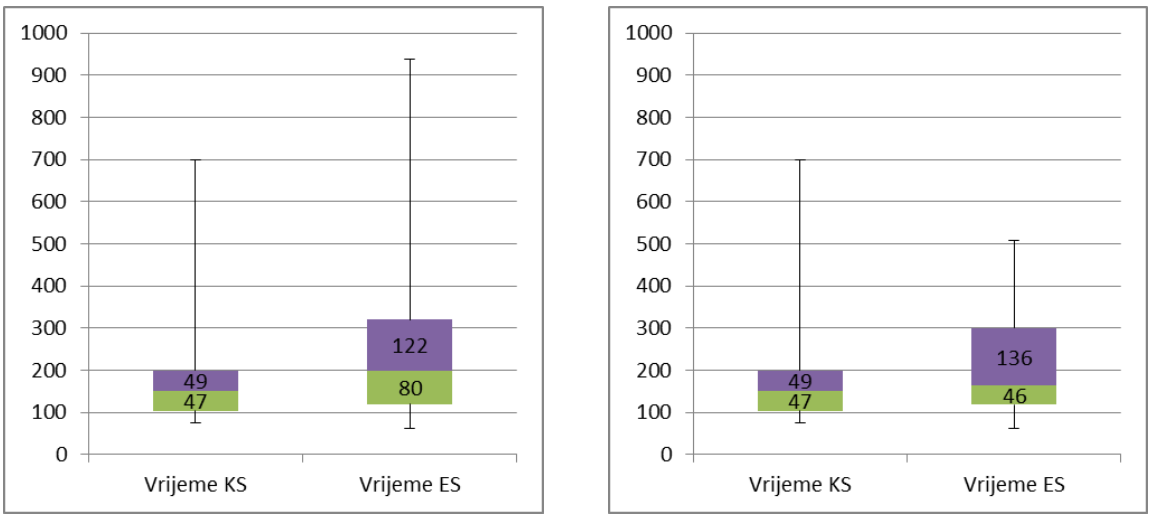
korištenjem sustava na optimalniji način (ravnomjerno korištenje obje funkcionalnosti sustava) što je put prema kojem vode i ponuđene sugestije.



Slika 85. Usporedba aktivnosti učenja sa i bez „outlier“-a



Slika 86. Usporedba aktivnosti ponavljanja sa i bez „outlier“-a



Slika 87. Usporedba vremena učenja sa i bez „outlier“-a

Kod vremena učenja, ponovno je uočljiva značajna promjena bez dvaju *outlier*-a, ali je utrošeno vrijeme eksperimentalne skupine (75%) još uvijek veće od onog kontrolne skupine. Duže vrijeme učenja ne treba se smatrati nužno lošom karakteristikom, posebno kada algoritam procjene kvalitete procesa učenja koji ocjenjuje dobivene klastere daje više bodova onoj skupini koja ima duže vrijeme učenja od ostalih skupina. Razlog tome je nagrađivanje dužeg vremena posvećenog učenju, umjesto direktnog prelaska na provjeru znanja.

Iako spomenute karakteristike govore u prilog hipotezi 2, iz prikazanih podataka nije moguće sa sigurnošću utvrditi povezanost između učinkovitosti procesa učenja između dvije skupine (sa ili bez sugestija temeljenih na dubinskoj analizi podataka). Sljedeći dio analize konačnog skupa podataka utvrđuje korelaciju korištenja, odnosno praćenja ponuđenih sugestija i broja aktivnosti odnosno vremena potrebnog za savladavanje domene znanja.

7.3.3 Korelacija broja praćenih sugestija i aktivnosti učenja

Nakon analize procesa učenja vizualnom analizom podataka, te primjenom metoda deskriptivne statistike, uspoređen je odnos korištenih sugestija (omjer broja prikazanih sugestija i broja praćenih sugestija, odnosno sugestija na koje je student kliknuo) i broja aktivnosti učenja. U prikazanoj tablici, također je navedena informacija o pripadnosti studenta klasteru.

Iz Tablice 29 moguće je zaključiti sljedeće:

- studenti najbolje ocjenjenog klastera imaju gotovo 6 puta manji prosječan broj aktivnosti učenja (prosjek nije uključivao neaktivne studente) u odnosnu na klaster B iz čega slijedi i manji broj prikazanih sugestija
- u omjeru broja prikazanih i korištenih sugestija vidljivo je da su studenti klastera A koristili sugestije gotovo dvostruko više od studenata klastera B.

Na temelju podataka o broju praćenih sugestija i broju aktivnosti učenja izračunata je vrijednost Pearson korelacijskog koeficijenta (stupci „PRA/PRIK“ i „UČENJA“) za oba klastera.

Iznos Pearson koeficijenta korelacije za klaster A = **0,39**.

Iznos Pearson koeficijenta korelacije za klaster B = **-0,17**.

Prema dobivenim rezultatima, vidljivo je da je klaster A imao pozitivnu korelaciju između omjera korištenja sugestija i aktivnosti učenja, dok ista korelacija ne postoji, odnosno ima vrijednost vrlo blizu nuli uz negativan predznak.

Usporedba vrijednosti eksperimentalne skupine u domeni „Uzorci dizajna“ prikazana je u Tablici 30. Vrijednosti ponovno prikazuju značajno manju prosječnu vrijednost aktivnosti učenja (78 u odnosu na 389) u najboljem klasteru, te daleko viši postotak omjera prikazanih i praćenih (kliknutih) sugestija u odnosu na klaster C.

Tablica 29. Povezanost korištenja sugestija i aktivnosti učenja (domena „Osnovne web tehnologije“)

STUDENT	CLUSTER	PRIKAZANIH	PRAĆENIH	PRA/PRIK	UČENJA
369	A	102	22	21,57%	101
379	A	93	26	27,96%	152
327	A	40	0	0,00%	50
320	A	35	7	20,00%	49
346	A	28	7	25,00%	54
307	A	28	6	21,43%	35
368	A	26	6	23,08%	31
263	A	22	11	50,00%	51
251	A	18	3	16,67%	36
233	A	18	0	0,00%	38
311	A	14	0	0,00%	8
261	A	7	0	0,00%	14
250	A	7	0	0,00%	54
319	A	0	0	0,00%	18
284	A	0	0	0,00%	23
304	A	0	0	0,00%	33
372	A	0	0	0,00%	151
267	A	0	0	0,00%	12
257	A	0	0	0,00%	43
244	A	0	0	0,00%	44
315	A	0	0	0,00%	157
PROSJEK		33,69	6,77	16%	51,77
STUDENT	**CLUSTER**	**PRIKAZANIH**	**PRAĆENIH**	**PRA/PRIK**	**UČENJA**
303	B	221	1	0,45%	381
259	B	217	12	5,53%	354
285	B	32	9	28,13%	327
264	B	13	0	0,00%	296
254	B	0	0	0,00%	366
383	B	0	0	0,00%	375
PROSJEK		120,75	5,50	9%	339,50

Tablica 30. Povezanost korištenja sugestija i aktivnosti učenja (domena „Uzorci dizajna“)

STUDENT	CLUSTER	PRIKAZANIH	PRACENIH	PRA/PRIK	UČENJA
264	A	67	47	70,15%	262
263	A	64	38	59,38%	100
285	A	7	4	57,14%	113
315	A	4	2	50,00%	51
202	A	49	14	28,57%	147
250	A	4	1	25,00%	54
372	A	66	9	13,64%	71
346	A	9	1	11,11%	61
267	A	39	4	10,26%	14
311	A	162	12	7,41%	115
369	A	28	2	7,14%	79
303	A	33	2	6,06%	68
319	A	18	1	5,56%	48
320	A	79	4	5,06%	65
368	A	40	2	5,00%	48
280	A	90	4	4,44%	105
259	A	33	1	3,03%	19
384	A	67	1	1,49%	60
261	A	68	0	0,00%	46
328	A	40	0	0,00%	46
284	A	28	0	0,00%	40
251	A	16	0	0,00%	119
257	A	2	0	0,00%	68
307	A	0	0	0,00%	38
327	A	0	0	0,00%	76
PROSJEK		44,04	6,48	16%	78,22
233	B	0	0	0,00%	0
236	B	0	0	0,00%	0
244	B	0	0	0,00%	0
				s	
179	C	494	29	5,87%	325
194	C	931	12	1,29%	660
193	C	647	7	1,08%	524
379	C	278	2	0,72%	198
254	C	246	0	0,00%	273
383	C	221	0	0,00%	357
PROSJEK		469,50	8,33	1%	389,50

Na temelju prikazanih podataka izračunata je vrijednost Pearsonovog koeficijenta korelacije za klastere A i C. Klaster B su sačinjavali neaktivni korisnici.

Iznos Person korelacijskog koeficijenta za klaster A: **0,60**.
Iznos Person korelacijskog koeficijenta za klaster C: **-0,01**.

Kao i kod prethodne domene, pokazuje se jaka korelacija između omjera korištenja sugestija i (manjeg) broja aktivnosti učenja kod studenata u klasteru A. Kod studenata u klasteru C takva korelacija ne postoji (odnosno ima vrijednost vrlo blisku nuli).

Na temelju provedene analize, odnosno vrednovanja rada sustava na temelju prikupljenih podataka, zaključeno je da se **Hipoteza 2 : Povećanje prilagodljivosti tutorskog modela povećava učinkovitost učenja pomoću inteligentnog tutorskog sustava** smatra potvrđenom.

8 Zaključak

U ovom doktorskom radu opisan je model poboljšanja prilagodljivosti tutorskog sustava, implementiranog kao prototip web aplikacije nazvane DITUS (*Department of Informatics TUtoring System*). Sustav DITUS omogućuje cjelokupnu pripremu i provedbu procesa e-učenja, te prilagođava strukture pojmova za učenje trenutnoj razini znanja korisnika. Kako bi se povećala učinkovitost procesa učenja pomoću sustava, njegov tutorski model je proširen funkcionalnošću prikaza sugestija koje pomažu korisniku pri kreiranju vlastite strategije savladavanja domene znanja predstavljene u sustavu. Odabir i prikaz sugestija temelje se na prikupljanju i dubinskoj analizi podataka o interakcijama korisnika sa sustavom.

Dubinska analiza podataka je općeniti naziv za primjenu metoda iz područja strojnog učenja, umjetne inteligencije i statistike na različite skupove podataka s ciljem otkrivanja korisnih informacija odnosno znanja, na temelju kojeg se mogu donositi poslovne odluke ili poboljšavati postojeći poslovni i drugi sustavi. Primjena tih metoda u domeni obrazovanja najčešće zahtijeva prilagodbu ulaznih podataka, te uključivanje dobivenih rezultata u druge obrazovne metode ili procese. Zbog svoje složenosti i potrebe za angažmanom više eksperata, dubinske analize podataka u obrazovanju se provode rijetko i s dužim vremenskim odmakom između pokretanja procesa i implementacije dobivenih rezultata. Analiza i interpretacija dobivenih rezultata većinom predstavlja jedinu svrhu i završnu fazu tog procesa, dok su mogućnosti primjene tih rezultata za poboljšanje procesa učenja nedovoljno iskorištene.

Cilj razvoja ovog sustava bila je verifikacija ideje o mogućnosti povezivanja sustava za e-učenje s alatima za dubinsku analizu podataka, te razvoja arhitekture za provedbu cjelokupnog procesa dubinske analize podataka o učenju, automatske interpretacije rezultata, te njihove implementacije u sustav s ciljem poboljšanja njegove učinkovitosti. Takav sustav uklanja spomenute nedostatke i omogućuje kontinuiranu primjenu dubinske analize podataka bez vremenskog odmaka.

Novi model arhitekture opisanog sustava sastoji se od podsustava za prikupljanje, obradu i pripremu podataka o učenju, podsustava za komunikaciju s alatima za dubinsku analizu podataka, zatim podsustava za klasteriranje korisnika i podsustava za otkrivanje čestih i učinkovitih putanja korisnika. Spomenuti podsustavi čine osnovu novog tutorskog modela sustava.

Osnovni preduvjet dubinske analize podataka je prikupljanje što veće količine podataka o interakcijama korisnika sa sustavom DITUS. U tu svrhu razvijen je novi model baze

podataka, te dodane funkcionalnosti za pohranu podataka o svim aktivnostima korisnika, korištenim nastavnim materijalima i provjerama znanja.

Podsustav za klasteriranje korisnika predstavlja prvu fazu analize podataka i ima svrhu ostvarivanja bolje pedagoške/tutorske funkcije sustava. Kako bi sustav mogao korisnicima nuditi sugestije prilagođene njihovoj razini znanja, ali i načinu učenja, podsustav za klasteriranje otkriva skupine korisnika koji sustav koriste na sličan način i napreduju kroz domenu znanja sličnom brzinom. Primjenom vlastitog algoritma automatski se interpretira kvaliteta procesa učenja svake skupine, te se one na temelju te interpretacije rangiraju. Na temelju rezultata rada ovog podsustava, izvršava se podsustav za otkrivanje čestih i učinkovitih putanja korisnika. Kako bi se korisnicima pomoglo u savladavanju domene znanja, provodi se dubinska analiza niza pojmova koje su korisnici učili prije dolaska do trenutnog pojma, te niza pojmova koje su korisnici učili nakon završetka učenja trenutnog pojma. Cilj je u skupu svih nizeva (putanja) otkriti česte i učinkovite nizeve koje novi tutorski model sustava može koristiti za odabir i prezentaciju sugestija. Bez podsustava za klasteriranje korisnika, ne bi postojala mogućnost provedbe otkrivanja čestih i učinkovitih putanja za svaki klaster, što bi značilo prikazivanje istih sugestija svim korisnicima neovisno o njihovom načinu rada i učinkovitosti.

Koncept sugestija temelji na pretpostavci da na temelju objektivne analize podataka o učenju možemo, umjesto da korisnika prepustimo pogađanju slijeda pojmova za učenje, u trenutku odabira pojma, sugerirati jedan pojam koji je korisno učiti *prije* tog pojma, odnosno, po završetku učenja, sugerirati jedan pojam koji je korisno učiti *nakon* tog pojma. Ukoliko se analizom putanja otkriju one najučinkovitije za svaki pojam u domeni, očekuje se da bi praćenjem ponuđenih sugestija korisnik trebao ostvariti optimalniji „put“ kroz domenu znanja, odnosno poboljšati kvalitetu procesa učenja.

Spomenute analize podataka provode se u skladu s intenzitetom rada korisnika u sustavu čime se vremenski odmak između prikupljanja podataka svodi s više mjeseci (trajanje semestra) na nekoliko minuta.

Oba spomenuta podsustava podržana su slojem za komunikaciju s alatima za dubinsku analizi podataka koji omogućava pokretanje dubinske analize podataka u bilo kojem trenutku.

Karakteristika sustava DITUS je pokretanje dubinske analize podataka nad rastućim skupom podataka o interakcijama korisnika sa sustavom, čime se omogućava poboljšanje prilagodljivosti sustava već unutar nekoliko sati od početka korištenja sustava. Preduvjet takvog načina rada bio je razvoj vlastitih algoritama za pripremu ulaznih skupova podataka,

automatizaciju provedbe dubinske analize podataka, te korištenje dobivenih rezultata u poboljšanom tutorskom modelu sustava.

Verifikacija ideje te razvijenog modela i prototipa sustava je provedena istraživanjem na dvije domene znanja, koje su savladavali studenti podjeljeni u kontrolnu i eksperimentalnu skupinu, pri čemu je kontrolna skupina koristila nepromjenjenu verziju sustava, dok je eksperimentalna skupina koristila verziju sustava s proširenom arhitekturom i poboljšanim tutorskim modelom sustava za odabir i prezentaciju sugestija.

Vrednovanje rada sustava provedeno je usporedbom rezultata testova znanja, anketom o zadovoljstvu korisnika sustavom, te analizom prikupljenih podataka o procesima učenja obje skupine, pri čemu su korištene metode vizualizacije podataka, te deskriptivne statistike. Dobiveni rezultati ukazuju na poboljšanje učinkovitosti procesa učenja kod eksperimentalne skupine u obje domene znanja. Analizom podataka utvrđena je pozitivan odnos između praćenja ponuđenih sugestija i (smanjenja) broja aktivnosti potrebnih za savladavanje domene znanja.

U budućem radu provoditi će se eksperimenti nad različitim verzijama algoritama za klasteriranje korisnika i otkrivanje čestih i učinkovitih putanja korisnika. Kod algoritma za procjenu kvalitete procesa učenja uočeni su nedostatci kod rangiranja skupine neaktivnih studenata. Kod algoritma za otkrivanje čestih i učinkovitih putanja korisnika, moguća je implementacija različitih pristupa pripremi ulaznog skupa podataka, odnosno transformacije niza aktivnosti u oblik prilagođen za dubinsku analizu putanja.

Nakon razvoja cjelovitih domena znanja (koje će sadržavati dvostruko veći broj pojmova) istraživanje će se ponoviti. Samim povećanjem skupa pojmova, kao i vremena u kojem će sustav biti dostupan studentima, očekuje se značajno povećanje ulaznih skupova podataka, a samim time i kvalitete dobivenih rezultata.

Također će se nastaviti razvijati modul za vizualnu analitiku podataka o učenju koji je namjenjen nastavnicima, a koji im omogućuje da prate aktivnosti korisnika u sustavu, te pokreću dubinske analize podataka, prilagođene razini tehničkog znanja nastavnika.

Literatura

Ahmed, C. F. (2010). A novel approach for mining high-utility sequential patterns in sequence databases. *ETRI journal, 32*(5), 676-686.

AIED. (2016). Preuzeto 20. 4 2016 iz http://iaied.org/about/

Aleven, V., Ashely, K., Lynch, C., & Pinkwart, N. (2006). Ill-Defined Domains. *Proceedings of the International Conference on Intelligent Tutoring Systems.*

Ali, L., Asadi, M., Gašević, D., Jovanović, J., & Hatala, M. (2013). Factors influencing beliefs for adoption of a learning analytics tool: An empirical study. *Computers & Education, 62*(1), 130-148.

Ali, L., Hatala, M., Gašević, D., & Jovanović, J. (2012). A qualitative evaluation of evolution of a learning analytics tool. *Computers & Education, 58*(1), 470-489.

Amari, S. I. (1992). Four types of learning curves. *Neural Computation, 4*(4), 605-618.

Apache. (2016). *HTTP server project*. Preuzeto 20. 4 2016 iz https://httpd.apache.org/

ARFF Data format. (2016). Preuzeto 20. 4 2016 iz https://weka.wikispaces.com/ARFF

Azevedo, A. I. (2008). KDD, SEMMA and CRISP-DM: a parallel overview. *IADS-DM.*

Baker, R. (2010). Data mining for education. U *International encyclopedia of education 7* (str. 112-118).

Baker, R., & Siemens, G. (2014). Educational data mining and learning analytics. U *Cambridge Handbook of the Learning Sciences.* Cambridge.

Baker, R., & Yacef, K. (2009). The State of Educational Data Mining in 2009: A review and Future Visions. *Journal of Educational Data Mining, 1*(1), 3-17.

Baker, R., Goldstein, A., & Heffernan, N. (2011). Detecting learning moment-by-moment. *International Journal of Artificial Intelligence in Education, 21*(1), 5-25.

Beck, J. M. (2008). How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students. *Proceedings of the 9th International Conference on Intelligent Tutoring Systems*, (str. 353-362).

BlackBoard.com. (2016). Preuzeto 20. 4 2016 iz http://uki.blackboard.com/about-us

Bloom, B. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to- one tutoring. *Educational Researcher, 13*, 4-16.

Bosch, N. D. (2015). Automatic detection of learning-centered affective states in the wild. *Proceedings of the 20th International Conference on Intelligent User Interfaces*, (str. 379-388).

Bouchet, F. H. (2013). Clustering and Profiling Students According to their Interactions with an Intelligent Tutoring System Fostering Self-Regulated Learning. *Journal of Educational Data Mining, 5*(1), 104-146.

Brain, B. (2016). *The Teachable Agents Group*. Preuzeto 20. 4 2016 iz http://www.teachableagents.org/research/bettysbrain.php

Bray, T. P.-M. (2016). *Extensible markup language (XML)*. (World Wide Web Consortium Recommendation REC-xml-19980210) Preuzeto 20. 4 2016 iz http://www. w3. org/TR/1998/REC-xml-19980210

Brusilovsky, P., Sosnovsky, S., & Shcherbinina, O. (2004). QuizGuide: Increasing the educational value of individualized self-assessment quizzes with adaptive navigation support. *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, (str. 1806-1813).

Brusilowsky, P., & Peylo, C. (2003). Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education, 13*, 159-172.

Chapman, P. C. (2000). *CRISP-DM 1.0 Step-by-step data mining guide.*

Chen, P. P.-S. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems, 1*(1), 9-36.

Cohen, P. K. (1982). Educational outcomes of tutoring: A metaanalysis. *American Educational Research Journal, 19*, 237-248.

Corbett, A. T. (1997). Intelligent tutoring systems . U *Handbook of humancomputer interaction* (str. 849-874).

CSS3. (2016). Preuzeto 20. 4 2016 iz https://www.w3.org/standards/techs/css#w3c_all

Delone, W. H. (2003). The DeLone and McLean model of information systems success: a ten-year update. *Journal of management information systems, 19*(4), 9-30.

EDM, S. (2016). *International Educational Data Mining Society*. Preuzeto 20. 4 2016 iz http://www.educationaldatamining.org/

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From Data Mining to Knowledge Discovery in Databases. *American Association for Artificial Intelligence*, 37-54.

Fini, A. (2009). The Technological Dimension of a Massive Open Online Course: The Case of the CCK08 Course Tools. *International Review of Research in Open and Distance Learning, 10*(5).

Fournier-Viger, P., Gomariz, A., Gueniche, T., Soltani, A., Wu, C., & Tseng, V. (2014). SPMF: a Java Open-Source Pattern Mining Library. *Journal of Machine Learning Research (JMLR), 15*, 3389-3393.

Fournier-Viger, P., Nkambou, R., & Nguifo, E. M. (2008). A knowledge discovery framework for learning task models from user interactions in intelligent tutoring systems. *MICAI 2008: Advances in Artificial Intelligence.*

Fournier-Viger, P., Nkambou, R., & Nguifo, E. M. (2010). Building intelligent tutoring systems for ill-defined domains. *Advances in intelligent tutoring systems*, 81-101.

GitSCM. (2016). Preuzeto 20. 4 2016 iz https://git-scm.com/

Gordon, A. D. (1999). *Classification, 2nd ed.* Boca Raton: Chapman & Hall/CRC.

Guazzelli, A. Z. (2009). PMML: An open standard for sharing models. *The R Journal, 1*(1), 60-65.

Guerra, J. S. (2014). The Problem Solving Genome: Analyzing Sequential Patterns of Student Work with Parameterized Exercises. *Proceedings of the 7th International Conference on Educational Data Mining*, (str. 153-160).

Hornick, M. F. (2010). Java data mining: strategy, standard, and practice: a practical guide for architecture, design, and implementation. *Morgan Kaufmann.*

HTML. (2016). Preuzeto 20. 4 2016 iz https://www.w3.org/TR/html5/

HTTP. (2016). *Hypertext Transfer Protocol -- HTTP/1.1*. (w3.org) Preuzeto 20. 4 2016 iz https://www.w3.org/Protocols/rfc2616/rfc2616.html

JQuery. (2016). Preuzeto 20. 4 2016 iz https://jquery.com/

JQuery. (2016). *Mobile*. Preuzeto 20. 4 2016 iz https://jquerymobile.com/

Kappel, G. (2005). *Web engineering.* John Wiley.

Kaufman, L. &. (2009). *Finding groups in data: an introduction to cluster analysis.* John Wiley & Sons.

Killaby, E. (2016). *Using closure tables to manage hierarchical relations in MySQL*. Preuzeto 20. 4 2016 iz https://gist.github.com/ekillaby/2377806

Kinnebrew, J. S. (2013). A contextualized, differential sequence mining method to derive students' learning behavior patterns. *JEDM-Journal of Educational Data Mining, 5*(1), 190-219.

Klašnja-Milićević, A. V. (2011). E-Learning personalization based on hybrid recommendation strategy and learning style identification. *Computers & Education, 56*(3), 885-899.

Koutri, M. A. (2005). A survey on web usage mining techniques for web-based adaptive hypermedia systems. U *Adaptable and Adaptive Hypermedia Systems.* Hershey.: Idea Publishing Inc.

Kovačić, B. (2002). *Sustav učenja na daljinu temeljen na dijalogu.* Zagreb: Fakultet elektrotehnike i računarstva.

Krasner, G. E. (1988). A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of object oriented programming, 1*(3), 26-49.

Krüger, A. M. (2010). A Data Model to Ease Analysis and Mining of Educational Data. *Proceedings of the International Conference on Educational Data Mining.*

Lloyd., S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory , 28*(2), 129–137.

Lynch, C. A. (2006). Defining ill-defined domains; a literature survey. *Proceedings of the workshop on intelligent tutoring systems for ill-defined domains at the 8th international conference on intelligent tutoring systems*, (str. 1-10).

McGarry, L., Packer, A., Fino, E., Nikolenko, V., Sippy, T., & Yuste, R. (2010). Quantitative classification of somatostatin-positive neocortical interneurons identifies three interneuron subtypes. *Neural Circuits, 4*(12).

Milligan, G. W. (1989). A validation study of a variable weighting algorithm for cluster analysis. *Journal of Classification, 6*, 53-71.

Mitrović, A. (2003). An intelligent SQL tutor on the web. *International Journal of Artificial Intelligence in Education, 13*(2-4), 173-197.

Moodle. (2016). *Moodle org*. Preuzeto 20. 4 2016 iz https://moodle.org

Mostow, J. &. (2006). Some useful tactics to modify, map and mine data from intelligent tutors. *Natural Language Engineering, 12*(02), 195-208.

MySQL. (2016). Preuzeto 20. 4 2016 iz https://www.mysql.com/

Nkambou, R. F.-V. (2011). Learning task models in ill-defined domain using an hybrid knowledge discovery framework. *Knowledge-Based Systems, 24*(1), 176-185.

Pei, J. H.-A. (2004). Mining sequential patterns by pattern-growth: The Prefixspan approach. *IEEE Transactions on Knowledge and Data Engineering, 16*(11), 1424-1440.

Pentaho. (2016). *Data Mining Algorithms and Tools in Weka*. Preuzeto 20. 4 2016 iz http://wiki.pentaho.com/display/DATAMINING/Data+Mining+Algorithms+and+Tools+in+Weka

Perera, D. K. (2009). Clustering and sequential pattern mining of online collaborative learning data. *Knowledge and Data Engineering, 21*(6), 759-772.

Petz, B. (2004). *Osnovne statističke metode za nematematičare.* Jastrebarsko: Naklada Slap.

Phalcon. (2016). *PHP*. Preuzeto 20. 4 2016 iz https://phalconphp.com/en/

PHP. (2016). Preuzeto 20. 4 2016 iz http://www.php.net

Piatetsky-Shapiro, G. (2014). *KDnuggets Methodology Poll*. Preuzeto 20. 4 2016 iz http://www.kdnuggets.com/polls/2014/analytics-data-mining-data-science-methodology.html

Pinto, H. H. (2001). Multi-dimensional sequential pattern mining. *Proceedings of the tenth international conference on Information and knowledge management* (str. 81-88). ACM.

Quizjet. (2016). *QuizJet Overview*. Preuzeto 20. 4 2016 iz http://www.pitt.edu/~paws/system_quizjet.htm

Romero, C., & Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert systems with applications, 33*(1), 135-146.

Romero, C., & Ventura, S. (2010). Educational Data Mining: A Review of the State-of-the-Art. *Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, 40*(6), 601-618.

Romero, C., & Ventura, S. (2013). Data mining in education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 3*(1), 12-27.

Romero, C., Castro, C., & Ventura, S. (2013). A Moodle Block for Selecting, Visualizing and Mining Students' Usage Data. *Proceedings of the 6th International Conference on Educational Data Mining.* Memphis.

Romero, C., Ventura, S., Delgado, J. A., & DeBra, P. (2007). Personalized links recommendation based on data mining in adaptive educational hypermedia systems. *Creating New Learning Experiences on a Global Scale, 2*, 292-306.

Rosenberg, M. J. (2001). *E-learning: Strategies for delivering knowledge in the digital age* (Svez. 3). New York: McGraw-Hill.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53-65.

Slavuj, V. K. (2013). Assessing user satisfaction with a system for e-learning based on dialogue. *Digital Document and Society*.

SRI, C. f. (2012). *Enhancing Teaching and Learning Through Educational Data Mining and Learning Analytics: An Issue Brief.* USA: U.S. Department of Education.

Srikant, R. &. (1996). *Mining sequential patterns: Generalizations and performance improvements.* Berlin Heidelberg: Springer.

Symons, M. J. (1981). Clustering criteria and multivariate normal mixtures. *Biometrics, 37*, 35-43.

Tibshirani, R. W. (2001). Estimating the number of data clusters via the gap statistic. *Journal of the Royal Statistical Society B, 63*, 411-423.

Vlissides, J., Helm, R., Johnson, R., & Gamma, E. (1995). *Design patterns: Elements of reusable object-oriented software.* Reading: Addison-Wesley.

Weka3. (2016). *Weka 3: Data mining software in Java*. Preuzeto 20. 4 2016 iz http://www.cs.waikato.ac.nz/ml/weka/

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques (3 ed.).* Elsevier.

Xu, R., & Wunsch, D. I. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks, 16*(3), 645-678.

Yin, J. Z. (2012). Uspan: an efficient algorithm for mining high utility sequential patterns. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, (str. 660-668).

Zaïane, O. R. (2002). Building A Recommender Agent for e-Learning Systems. *Proceedings of the International Conference in Education*, (str. 55-59). Auckland.

Zaiane, O. R., & Luo, J. (2001). Towards evaluating learners' behaviour in a web-based distance learning environment. *IEEE International Conference on Advanced Learning Technologies*, (str. 357-360).

Zimmerman, D. W. (1997). A Note on Interpretation of the Paired-Samples t Test". *Journal of Educational and Behavioral Statistics, 22*(3), 349–360.

Zorilla, M., & Garcia-Saiz, D. (2013). A service oriented architecture to provide data mining services for non-expert data miners. *Decision Support Systems, 55*(1), 399-411.

Zorrila, M., Garcia, D., & Alvarez, E. (2010). A Decision Support System to improve e-Learning Environments. *Proceedings of the 2010 EDBT/ICDT Workshops*, (str. 11-18). New York.

Popis slika

Slika 1. Životni ciklus dubinske analize podataka prema CRISP-DM modelu (prilagođeno prema (Chapman, 2000)) 8
Slika 2. Popis općenitih zadataka i rezultata prema CRISP-DM modelu (prilagođeno prema (Chapman, 2000)) 8
Slika 3. Venn dijagram položaja područja istraživanja EDM i LA (Romero & Ventura, 2013) 10
Slika 4. Općeniti ciljevi primjene DAP u obrazovanju (Romero & Ventura, 2007) 11
Slika 5. Općeniti model tijeka provedbe DAP u obrazovanju, prilagođeno prema (Romero & Ventura, 2013) 13
Slika 6. Model učenja na daljinu zasnovan na dijalogu (Kovačić, 2002) 19
Slika 7. Primjer strukture pojmova u domeni znanja (Kovačić, 2002) 20
Slika 8. Proširena arhitektura sustava 23
Slika 9. Funkcija podsustava za klasteriranje korisnika 29
Slika 10. Primjer vizualizacije kvalitete modela klasteiranja metodom siluete (McGarry, i dr., 2010) 37
Slika 11. Koraci procjene kvalitete procesa učenja 40
Slika 12. Primjeri nastanka različitih putanja kroz zamišljenu domenu znanja 44
Slika 13. Primjer početnog i transformiranog skupa putanja korisnika 48
Slika 14. Primjer poruka korisniku u sučelju sustava 53
Slika 15. Arhitektura 3-slojnih i MVC aplikacija 56
Slika 16. Dio modela baze podataka za upravljanje korisnicima 57
Slika 17. Dio modela baze podataka za upravljanje domenom znanja 58
Slika 18. Dio modela baze podataka za provjere znanja 59
Slika 19. Dio modela baze podataka za aktivnosti učenja i model studenta 61
Slika 20. Dio modela baze podataka za provedene DAP 63
Slika 21. Pregled arhitekture sustava DITUS 64
Slika 22. Klasa za provedbu analiza DAP 66
Slika 23. Klasa za komunikaciju sa alatima za DAP 67
Slika 24. Klase uključene u proces grupiranja korisnika 68
Slika 25. Dijagram sekvenci za provedbu grupiranja korisnika (Algoritmi 1, 2 i 3) 68
Slika 26. Klase uključene u proces otkrivanja čestih i učinkovitih putanja korisnika 69
Slika 27. Dijagram sekvenci za provedbu analize čestih i učinkovitih putanja 70

Slika 28. UML dijagram aktivnosti procesa „Učenje“ 72
Slika 29. UML dijagram aktivnosti procesa „Ponavljanje“ 73
Slika 30. UML dijagram aktivnosti algoritma „Učenje“ 74
Slika 31. Klase uključene u provedbu procesa Učenja i Ponavljanja, te odabira sugestija 75
Slika 32. Početna stranica DITUS sustava 76
Slika 33. Sučelje za rad s domenama znanja 77
Slika 34. Sučelje za organizaciju grupa, domena i studenata 77
Slika 35. Početni ekran profesora nakon prijave – odabir domene 78
Slika 36. Pregled pojmova s akcijama unosa pitanja, sadržaja, izmjenu i brisanje pojma 78
Slika 37. Prikaz pitanja za odabrani pojam 79
Slika 38. Sučelje za unos pitanja za odabrani pojam 80
Slika 39. Sučelje za rad s nastavnim materijalima 81
Slika 40. Sučelje za izradu strukture pojmova 82
Slika 41. Sučelje za kreiranje ispita i odabir pitanja za ispit 83
Slika 42. Pregled domena nakon prijave studenta 83
Slika 43. Početna stranica funkcionalnosti učenja 84
Slika 44. Prikaz materijala za učenje i sugestije 84
Slika 45. Provjera znanja nakon prikaza materijala za učenje 85
Slika 46. Incijalno pitanje za sve podpojmove u strukturi učenja 85
Slika 47. Završni ekran učenja s prikazanom sugestijom 86
Slika 48. Početak procesa ponavljanja nakon odabira pojma 86
Slika 49. Prikaz pitanja u procesu ponavljanja 87
Slika 50. Prikaz rezultata odgovora na pitanje u procesu ponavljanja 87
Slika 51. Završni ekran procesa ponavljanja s prikazanom sugestijom 88
Slika 52. Početni ekran modula za analitiku učenja i rad s modulima za DAP 88
Slika 53. Pregled podataka o učenju odabrane grupe 89
Slika 54. Detaljni pregled učenja ili ponavljanja s vizualizacijom intenziteta aktivnosti 90
Slika 55. Sučelje za vizualizaciju aktivnosti grupe u vremenu 91
Slika 56. Sučelje za prikaz vizualizaciju podataka jednog studenta 91
Slika 57. Sučelje za pokretanje postupka klasteriranja 92
Slika 58. Prikaz rezultata procesa klasteriranja 92
Slika 59. Rast skupa podataka i aktivnosti korisnika eksperimentalne skupine po danima 96
Slika 60. Savladanost domene eksperimentalne skupine po danima 97

Slika 61. Broj aktivnih studenata eksperimentalne skupine po danima 97
Slika 62. Usporedba pdf(L_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna" 102
Slika 63. Usporedba pdf(R_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna" 102
Slika 64. Usporedba pdf(T_{std}) kod prve i posljednje analize u domeni "Uzorci dizajna" 102
Slika 65. Struktura domene znanja „Osnovne web tehnologije“ 112
Slika 66. Struktura domene znanja „Uzorci dizajna“ 114
Slika 67. Rezultati testa znanja KS za domenu „Osnovne web tehnologije“ 121
Slika 68. Rezultati testa znanja ES za domenu „Osnovne web tehnologije“ 121
Slika 69. Rezultati testa znanja KS za domenu „Uzorci dizajna“ 123
Slika 70. Rezultati testa znanja ES za domenu „Uzorci dizajna“ 123
Slika 71. Prosjek aktivnosti grupe u domeni "Osnovne web tehnologije" 130
Slika 72. Usporedba aktivnosti grupe u domeni „Osnovne web tehnologije“ 130
Slika 73. Prosjek broja postavljenih pitanja grupe u domeni "Osnovne web tehnologije" 131
Slika 74. Prosjek vremena učenja grupe u domeni "Osnovne web tehnologije" 131
Slika 75. Usporedba vremena učenja grupe u domeni „Osnovne web tehnologije“ 131
Slika 76. Krivulje učenja kontrolne skupine u domeni "Osnovne web tehnologije" 132
Slika 77. Krivulje učenja eksperimentalne skupine u domeni "Osnovne web tehnologije" .. 132
Slika 78. Odnos aktivnih i neaktivnih studenata kontrolne skupine 133
Slika 79. Odnos aktivnih i neaktivnih studenata eksperimentalne skupine 134
Slika 80. Krivulje učenja kontrolne skupine u domeni „Uzorci dizajna“ 134
Slika 81. Krivulje učenja eksperimentalne skupine u domeni „Uzorci dizajna“ 134
Slika 82. Prosjek aktivnosti učenja sa i bez „outlier“-a 136
Slika 83. Prosjek broja pitanja sa i bez „outlier“-a 136
Slika 84. Prosjek vremena učenja sa i bez „outlier“-a 136
Slika 85. Usporedba aktivnosti učenja sa i bez „outlier“-a 137
Slika 86. Usporedba aktivnosti ponavljanja sa i bez „outlier“-a 137
Slika 87. Usporedba vremena učenja sa i bez „outlier“-a 137

Popis tablica

Tablica 1. Mapiranje broja bodova na intervale 39
Tablica 2. Primjer dodjele broja bodova klasterima prema pripadnosti intervalu 42
Tablica 3. Konačno bodovanje procesa učenja po klasterima 42
Tablica 4. Mogući primjeri učinkovitosti i dužine putanja prikazanih na slici 12. 45
Tablica 5. Primjer ulaznog skupa podataka – baze sekvenci za USPAN algoritam 50
Tablica 6. Pregled poruka korisniku za odabir sugestije 53
Tablica 7. Savladanost domene studenata eksperimentalne skupine po danima 98
Tablica 8. Broj analiza grupiranja po danima za obje domene 99
Tablica 9. Pregled kreiranih modela grupiranja u domeni „Osnovne web tehnologije“ 99
Tablica 10. Pregled kreirani modela grupiranja u domeni "Uzorci dizajna" 100
Tablica 11. Broj provedenih analiza putanja u domeni „Osnovne web tehnologije“ 104
Tablica 12. Broj provedenih analiza putanja u domeni „Uzorci dizajna“ 104
Tablica 13. Pregled broja otkrivenih putanja po analizi (po klasteru i tipu) u domeni „Osnovne web tehnologije“ 105
Tablica 14. Prikazane i korištene sugestije u domeni „Osnovne web tehnologije“ 107
Tablica 15. Prikazane i korištene sugestije u domeni „Uzorci dizajna“ 108
Tablica 16. Rezultati t-testa skupina u domeni „Osnovne web tehnologije“ 110
Tablica 17. Rezultati t-testa skupina u domeni „Uzorci dizajna“ 110
Tablica 18. Pojmovi, pragovi, pitanja i nastavni materijali za domenu „Osnovne web tehnologije“ 113
Tablica 19. Pojmovi, pragovi, pitanja i nastavni materijali za domenu „Uzorci dizajna“ 115
Tablica 20. Rezultati testa znanja za KS i ES u domeni „Osnovne web tehnologije“ 120
Tablica 21. Rezultati testa znanja za KS i ES u domeni „Uzorci dizajna“ 122
Tablica 22. Ankete – OPĆE INFORMACIJE (domena „Osnovne web tehnologije“) 125
Tablica 23. Ankete - FLEKSIBILNOST (domena „Osnovne web tehnologije“) 125
Tablica 24. Anketa – ISKUSTVO (domena „Osnovne web tehnologije“) 126
Tablica 25. Anketa – SUGESTIJE (domena „Osnovne web tehnologije“) 127
Tablica 26. Anketa – OPĆE INFORMACIJE (domena „Uzorci dizajna“) 127
Tablica 27. Anketa – FLEKSIBILNOST (domena „Uzorci dizajna“) 128
Tablica 28. Ankete – SUGESTIJE (domena „Uzorci dizajna“) 128
Tablica 29. Povezanost korištenja sugestija i aktivnosti učenja (domena „Osnovne web tehnologije“) 139

Tablica 30. Povezanost korištenja sugestija i aktivnosti učenja (domena „Uzorci dizajna“) 140

Popis privitaka

Privitak 1. Test znanja za domenu „Osnovne web tehnologije“

1. Osnovno pravilo pisanja HTML-a je da svaki element koji je ugrađen u vanjski element mora završiti prije nego završi vanjski element. To pravilo se engleski naziva:

- plain model
- square model
- box model
- block model

2. Prednost OOP u odnosu na proceduralno programiranje može se uočiti na primjeru slanja objekta kao argumenta nekoj funkciji. Kod proceduralnog programiranja nekoj funkciji mogu poslati samo podatak (varijablu) ili polje podataka (array). Međutim, kod OOP, kada šaljemo objekt, funkciji koja je primila objekt dostupni su ne samo podaci objekta nego i :

- sve navedene opcije su točne
- drugi objekti koji možda sačinjavaju taj objekt
- metode, odnosno funkcionalnosti definirane za taj objekt
- metode, odnosno funkcionalnosti definirane za objekte koji možda sačinjavaju poslani objekt

3. Ukoliko želimo izbrisati obrazac sa 5 polja sa web stranice korištenjem JavaScripta, moramo napisati izraze koji će manipulirati Document Object Model-om (stablom strukture) te web stranice. Koji je najmanji broj izraza potreban za ukloniti obrazac (općenito gledajući, bez obzira na konkretan broj linija koda)?

- Dva
- Pet
- Tri
- Jedan

4. Kada zatražimo stranicu u web pregledniku, nećemo dobiti nikakav sadržaj ukoliko _____ protokol ne radi ispravno.

- IP
- DNS
- TCP
- HTTP

5. Što NIJE napisano u ovom URL-u: http://ditus.uniri.hr/dokumenti/arhiva/2014/14_10.pdf a poprimiti će defaultnu vrijednost 80?

- fragment
- port
- protokol
- querystring

6. Kada želimo kreirati AJAX (XmlHttpRequest) poziv putem JavaScripta koristimo se kojim dijelom JavaScript arhitekture:

- Core
- Navigation
- BOM
- DOM

7. Sadržaj ove XML datoteke u JSON obliku sastojao bi se od _____ objekta sa _____ elemenata u polju od koji bi se svaki sastojao od _____ podatka.

- Jednog, četiri, tri
- Jednog, četiri, dva
- Četiri, jednim, tri
- Četiri, jednim, dva

8. Jedinstvenost naziva polja u obrascima je ključna za dobivanje svih informacija od korisnika. Ukoliko u obrascu imamo tri polja istog naziva (a nisu dio "radio" grupe) i korisnik popuni sva tri polja, koju ćemo vrijednosti dobiti kada korisnik pošalje obrazac?

- sve tri
- samo zadnju
- samo prvu
- niti jednu

9. Ukoliko ispunimo anonimnu anketu na nekom portalu, jedini način da se vlasnik portala pokuša zaštititi od višestrukog glasanja iste osobe jer da uz glas zapiše i IP adresu računala sa kojeg ste glasali, koju može dohvatiti iz _____ .

- $_SESSION
- $_ENV
- $_SERVER
- $_REQUEST

10. Ako se u konstruktoru klase pojavljuje ključna riječ PARENT, onda se u deklaraciji te klase sigurno pojavljuje i ključna riječ:

- implement

- this
- extends
- overloads

11. Od koje verzije HTMLa je omogućeno korištenje okvia (frames)?

- 4
- 3
- 5
- 2

12. Ukoliko oblikujemo padajuću listu kao popis koji omogućuje višestruki odabir koji znak moramo dodati nazivu tog polja?

- uglate zagrade
- navodnike
- okrugle zagrade
- vitičaste zagrade

13. Ukoliko s nekom osobom obavimo Skype razgovor, a zatim posjetimo njihovu web stranicu i pošaljemo e-mail, koliko puta smo koristili asinkronu, a koliko sinkronu komunikaciju?

- Asinkronu dva puta, a sinkronu jednom
- Asinkronu jednom, a sinkronu dva puta
- Asinkronu tri puta
- Sinkronu tri puta

14. Što je od navedenog točno kada je riječ o HTML-u?

- HTML je skriptni jezik
- HTML je jezik za modeliranje
- HTML je jezik za formatiranje
- HTML je jezik za strukturiranje

15. Ako želimo napraviti konstantu u funkciji koristimo se ključnom riječi:

- define
- final
- var
- const

16. Ukoliko na obrascu moramo izgenerirati dva TEXT polja za svaki prikazani redak u tablici (dakle, imati ćemo dvije odvojene skupine text polja) koji postupak će osigurati da možemo razlikovati između skupina, ali i redaka? (navodnici i zagrade su namjerno izostavljene)

- input name=a_.$id value="" i input name=b_$id value=""
- input name=a_b_.$id value="" i input name=c_d_.$id value=""
- input name=a value="" i inaput name=b value=""
- input name=a value=b i input name=c value=d

17. Ako želimo naglasiti da konstruktor neke klase prima dva objekta, od kojih prvi mora biti instanca klase A, a drugi instanca klase B u deklaraciji konstruktora moramo napisati:

- public function _construct(A, B)
- public function _construct(instanceof(a) $a, instanceof(b) $b)
- public function _construct(A $a, B $b)
- public function _construct(a $a, b $b)

18. U koju skupinu pripadaju ovi elementi CSS-a - :hover, :lang, :empty ?

- pseudo-klase
- klase
- pesudo-elementi
- definicije

19. Ukoliko pošaljemo POST zahtjev na poslužitelj, a dobijemo odgovor koji počinje brojkom 3, vjerojatno se dogodilo sljedeće:

- Kliknuli smo na link, poslužitelj je obradio zahtjev bez greške
- Pokušali smo poslati obrazac, ali smo dobili informaciju da će naš zahtjev biti preusmjeren
- Kliknuli smo na link, ali se dogodila greška na poslužitelju
- Pokušali smo poslati obrazac, ali smo dobili informaciju da lokacija nije pronađena

20. Atribute klase možemo gledati kao varijable u funkciji, ali u instanci (objektu) atributi su:

- vrijednosti
- isto varijable
- objekti
- konstante

21. Kada korisnik pošalje obrazac pritiskom na dugme tipa Submit podaci iz obrasca, ovisno o vrijednosti METHOD atributa, mogu biti poslani u kojem dijelu HTTP zahtjeva?

- Samo u prvom
- Prvom ili drugom
- Prvom ili trećem
- Drugom ili trećem

22. Ključne riječi include, include_once, require i require_once obično koristimo za uključivanje biblioteka funkcija u glavni program. Jedina razlika je u tome što želimo da se

dogodi ukoliko datoteka nije pronađena ili ukoliko je ranije uključena. Ako nam datoteka koju uključujemo nije ključna za rad aplikacije (odnosno neće uzrokovati prekid rada), ali želimo biti sigurni da neće biti uključena više od jednom koristiti ćemo:

- require
- include_once
- include
- require_once

23. World Wide Web i Internet su termini koji se često poistovjećuju, međutim WORLD WIDE WEB (pojednostavljeno: HTTP) je usluga koja pripada _____ razini OSI modela, dok INTERNET (pojednostavljeno: IP) pripada _____ razini.

- prvoj, trećoj
- prvoj, drugoj
- drugoj, četvrtoj
- prvoj, petoj

24. Prilikom izlaska korisnika iz lokalne mreže na Internet, privatna IP adresa se pretvara u javnu IP adresu pomoću metode:

- ATP
- NAT
- WAT
- IPA

25. Ukoliko izradimo obrazac sa poljem tipa FILE (input type=file) odnosno omogućimo korisniku upload datoteka, u kojem superglobalnom polju PHP-a ćemo pronaći datoteku (na strani poslužitelja) kada korisnik pošalje obrazac?

- FILE
- POST
- GET
- FILES

26. Ukoliko u kodu HTML stranice primjetimo ovako napisane elemente: <br />, <img.... />, <hr /> to je siguran znak da je autor stranice koristio:

- XHTML
- HTML 5
- XML
- HTML 4.0

27. Ukoliko na jednoj web stranici stranici imamo 3 obrasca i sva tri se šalju na jednu PHP stranicu, moramo poslati informaciju PHP stranici o tome koji se obrazac šalje (inače ćemo

morati "ručno" prolaziti kroz POST i "otkriti" o kojem se obrascu radi). Najlakši i najjednostavniji (u smislu linija koda) način da to učinimo je:

- dodati parametar u querystring action atributa
- dodati ID atribut u svaki obrazac
- dodati skriveno polje u svaki obrazac
- dodati name atribut u svaki obrazac

28. Koji od navedenih protokola NIJE tehnički preduvjet za funkcioniranje World Wide Web-a:

- TCP
- HTTP
- IP
- DNS

29. Kada u web pregledniku duže vremena čekamo učitavanje web stranice, zapravo TCP protokol na našoj mrežnoj kartici čeka _____ bit.

- PSH
- RST
- URG
- ACK

30. Sučelje neke klase čine sve njene _____ metode.

- protected
- private
- final
- public

Privitak 2. Test znanja za domenu „Uzorci dizajna“

1. Omogućuje rad sa velikim brojem manjih objekata koji dijele zajedničko stanje (vrijednosti jednog ili više atributa), te imaju neko svoje, jedinstveno stanje (vrijednost jednog ili više atributa). Time se postiže smanjenje redundancije i poboljšavaju performanse. O kojem je uzorku riječ?

- Proxy
- Flyweight
- Composite
- Command

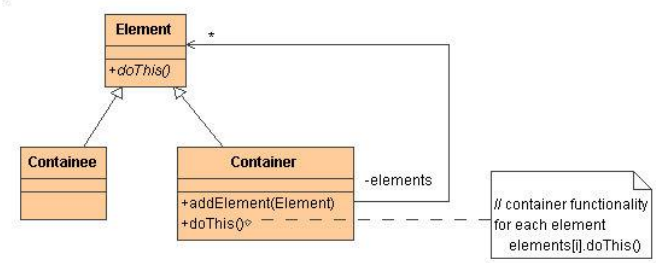
2. Što od navedenog točno opisuje Singleton uzorak?

- Koristimo je kada želimo poslati više podataka u jednom zahtjevu
- Uključuje jednu klasu koja je odgovorna za kreiranje najviše jedne instance
- Ovim uzorkom kreiramo objekte bez otkrivanja logike za kreiranje objekata
- Definira sučelje za kreiranje tvornice povezanih objekata

3. Koji od uzoraka mijenja ponašanje (funkcionalnost) u ovisnosti od svog stanja (vrijednosti atributa)?

- Template Method
- Mediator
- Strategy
- State

4. Koji uzorak predstavlja ovaj model?



- Facade
- Adapter
- Bridge
- Composite

5. Što je od navedenog točno za Strategy uzorak?

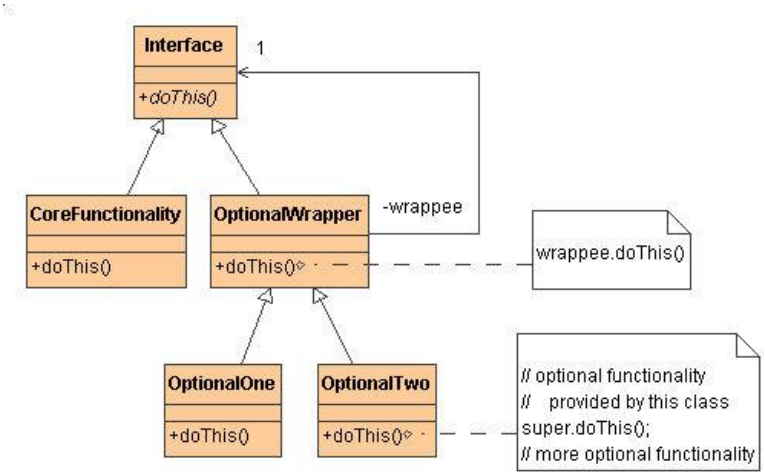
- Mijenja funkcionalnost u ovisnosti o stanju
- Omogućava prolazak kroz niz objekata
- Kreira objekt na temelju skrivene logike

- Omogućava dinamički promjenjivost jedne funkcionalnosti (algoritma)

6. Što od navedenog je točno za uzorke dizajna (design patterns)?

- Rješenja za česte probleme sa kojima se susreću programeri
- Predstavljaju "najbolje prekse" koje koriste iskusni OO programeri
- Razvili su se metodom pokušaja i pogrešaka kroz duži vremenski period
- Svi odgovori su točni

7. Koji uzorak predstavlja ovaj model?



- Adapter
- Decorator
- Composite
- Facade

8. Koji od navedenih uzoraka služi kao most/poveznica između dvaju nekompatibilnih sučelja (metoda)

- Builder
- Adapter
- Composite
- Prototype

9. Što od navedenog točno opisuje Abstract Factory uzorak?

- Koristimo je kada želimo poslati više argumenata objektu u jednom zahtjevu
- Koristi se sučelje za kreiranje tvornice povezanih objekata bez definiranja točno određene klase
- Jedna klasa odgovorna za kreiranje najviše jedne instance
- Kreira objekte bez otkrivanja logike kreiranja

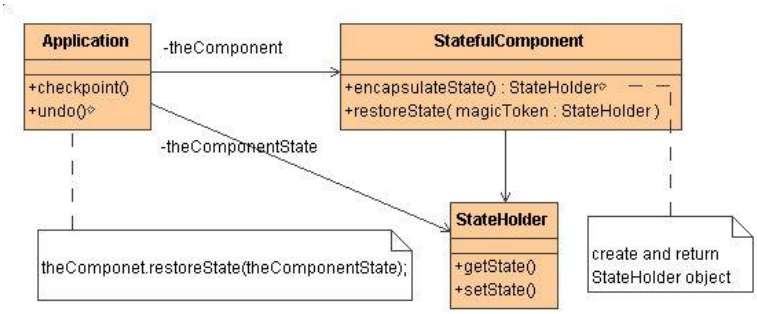
10. Definira vezu 1 prema više (1:M) između objekata tako da, kada jedan objekt promjeni stanja (podatak u nekom atributu), svi povezani objektu budu obavješteni o tome i ažurirani.

- Mediator
- Observer
- Event Notification
- Chain of Responsibility

11. Što od navedenog točno opisuje uzorke ponašanja (behavioral patterns)?

- Bave se kompozicijama (strukturama) klasa i objekata primjenjujući koncept nasljeđivanja
- Ovaj skup objekata bavi se razinom prezentacije
- Omogućuju kreiranje objekata uz skrivanje logike kreiranja
- Ovaj skup uzoraka bavi se komunikacijom između objekata

12. Koji uzorak predstavlja ovaj model?

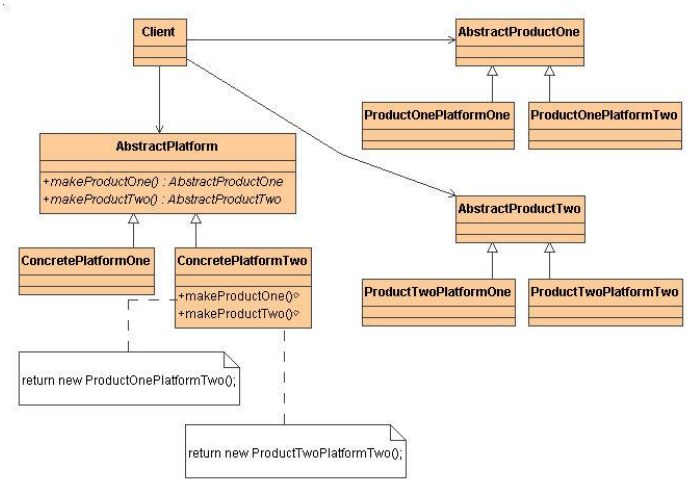


- Command
- Memento
- Mediator
- State

13. Definira osnovnu strukturu algoritma (niza naredbi koji obavljaju neki proces), ali implementaciju nekih koraka (metoda koje se pozivaju u algoritmu) prepušta podklasama. Time se omogućuje da isti algoritam bude implementiran na više različitih načina, ovisno o situaciji. O kojem je uzorku riječ?

- Prototype
- Interpretor
- Template Method
- Chain of Responsibility

14.Koji uzorak predstavlja ovaj dijagram?



- Builder
- Abstract Factory
- Factory Method
- Template Method

15. Osigurava da klasa može imati samo jednu instancu (objekt) i omogućuje globalnu jedinstvenu točku pristupa toj instanci. O kojem je uzorku riječ?

- Flyweight
- Builder
- Proxy
- Singleton

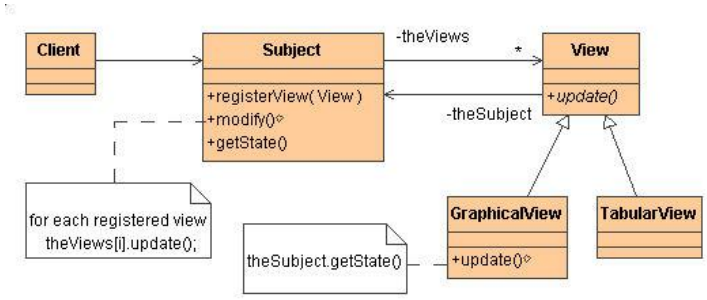
16. Koji uzorak koristimo kada moramo tretirati skup objekata na sličan način kao jedan objekt?

- Flyweight
- Facade
- Decorator
- Composite

17. Što je od navedenog točno za Decorator uzorak?

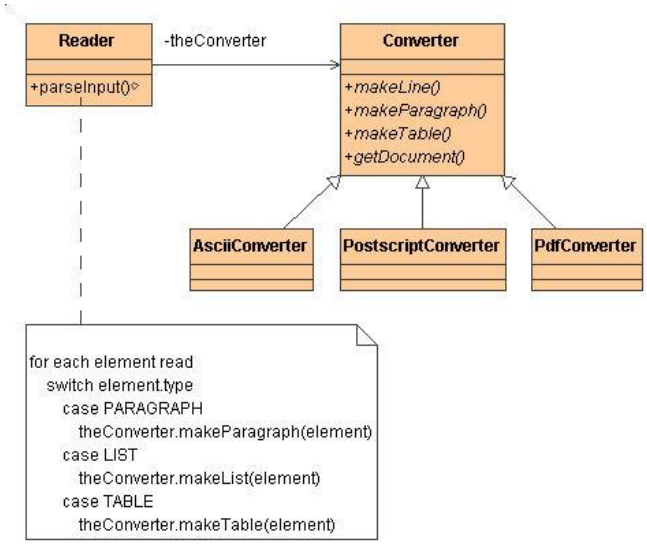
- Smanjuje broj kreiranih objekata i zauzeće memorije
- Skriva kompleksnost sustava pružajući jedinstveno sučelje za pristup sustavu
- Omogućuje dodavanje novih funkcionalnosti objektu bez promjene strukture
- Koristi se kada moramo tretirati skup objekata kao jedan

18. Koji uzorak predstavlja ovaj model?



- Observer
- MVC
- Strategy
- State

19. Koji uzorak predstavlja ovaj model?



- Command
- Abstract Factory
- Mediator
- Builder

20. Daje jedinstveno sučelje (skup public metoda) za skupinu sučelja (klasa i njihovih public metoda) koji čine neki podsustav. Definira sučelje više razine čime se klijentima olakšava korištenje cijelog podsustava. O kojem je uzorku riječ?

- Mediator
- Adapter
- Strategy
- Facade

21. Omogućuje sekvencijalni (jedan po jedan) pristup elementima agregiranog (složenog) objekta bez otkrivanja unutarnje reprezentacije objekata. O kojem je uzorku riječ?

- Composite
- Command
- Visitor
- Iterator

22. Točan popis skupina uzoraka dizajna je:

- Creational, Structural, Architectural
- Creational, Executional, Behavioral patterns
- Executional, Structural, Behavioral patterns
- Creational, Structural, Behavioral patterns

23. Enkapsulira zahtjev (poziv metode) kao objekt, čime se omogućuje dodavanje različitih parametara (argumenata) klijentima sa različitim zahtjevima, te omogućuje poništavanja (undo) pokušane (a neuspjele) operacije.

- Decorator
- Adapter
- Composite
- Command

24. Definira sučelje za kreiranje objekta, ali prepušta odluku koji/kakav objekt kreirati svojim podklasama. Postupak kreiranja može se razlikovati po podklasama. O kojem se uzorku radi?

- Abstract Factory
- Builder
- Factory Method
- Prototype

25. Koji uzorak koristimo kada je postupak kreiranja objekta "skup"?

- Prototype
- Filter
- Bridge
- Adapter

26. Što od navedenog točno opisuje Mediator uzorak?

- Omogućuje povrat stanja objekta u prethodno stanje
- Koristimo za smanjenje komunikacijske složenosti između objekata ili klasa
- Koristimo kada postoji 1:M veza između objekata kako bi obavijestili povezane objekte o promjeni

- Koristi se kada trebamo sekvencijalno pristupiti elementima složenog objekta

27. Definira skup algoritama, enkapsulira svaki zasebno i čini ih lako promjenjivima/zamjenjivima. Algoritam na taj način možemo lako mijenjati bez da korisnik klase zna išta o tome. O kojem je uzorku riječ?

- Strategy
- Template Method
- Decorator
- Visitor

28. Dinamički dodaje dodatne odgovornosti (funkcionalnosti) objektu i predstavlja alternativu izradi strukture podklasa u cilju proširenja funkcionalnosti. O kojem je uzorku riječ?

- Decorator
- Adapter
- Chain of Responsibility
- Composite

29. Kod originalne (ne Web) verzije MVC uzorka MODEL predstavlja objekte kojima se prenose podaci, ali može sadržavati i logiku za ažuriranje Controller-a ako se podaci u modelu promjene.

- Točno
- Netočno

30. Što označava termin "Gang of Four"?

- Autore knjige Design Patterns
- Ništa od navedenog
- Uzorke dizajna
- Sinonim za knjigu Design Patterns

Privitak 3. Anketa za provjeru zadovoljstva korisnika sustavom DITUS

1	**OPĆE INFORMACIJE**	
1	Spol	M Ž
2	Starost	
3	Razina studija (preddiplomski/diplomski)	PD D
4	Studijska grupa (jednopredmetni/dvopredmetni)	1P 2P
5	Studijska kombinacija	
2	**NAČIN KORIŠTENJA SUSTAVA**	
6	Koliko ste često koristili sustav za e-učenje pripremajući se za kolokvij iz kolegija Dinamičke web aplikacije 1:	Često Povremeno Rijetko Nikad
7	Od kuda ste najčešće pristupali sustavu:	Od kuće Sa fakulteta Sa radnog mjesta Druge lokacije
8	Koji ste modul sustava najćešće koristili:	Učenje Provjera
9	Koji je modul, po Vašem mišljenju, najkorisniji:	Učenje Provjera
10	Koji je modul, po Vašem mišljenju, najmanje koristan:	Učenje Provjera Nema nekorisnih modula
11	Jeste li koristili i neku drugu domenu osim Dinamičke web aplikacije 1:	Da Ne
3	**SADRŽAJI ZA UČENJE** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se slažem)	
13	Sadržaj materijala za učenje je, općenito gledajući, kvalitetan:	1 2 3 4 5
14	Sadržaj je prezentiran na jasan način:	1 2 3 4 5
15	Predstavljeni sadržaji su jasno zaokružene cjeline:	1 2 3 4 5
16	Prezentirani sadržaji su međusobno povezani i nadopunjuju se:	1 2 3 4 5
17	Prezentiran je najnoviji sadržaj (najnovije informacije) iz područja:	1 2 3 4 5
18	Problemi koji su izloženi u materijalima sagledani su iz različitih stajališta, ne samo sa jednog:	1 2 3 4 5
19	Predstavljeni sadržaji su previše detaljni:	1 2 3 4 5
20	Odabrane teme su raznovrsne:	1 2 3 4 5
21	Sadržaj unutar pojedinih tema dobro strukturiran i logički predstavljen:	1 2 3 4 5
22	Predstavljene teme su relevantne s obzirom na ishode učenja ovog kolegija:	1 2 3 4 5
23	U materijalima je jasno označen sadržaj zaštićen autorskim pravom (citati, poveznice):	1 2 3 4 5
24	Sadržaj me motivirao na učenje:	1 2 3 4 5
25	Prezentirani sadržaj mi je bio koristan:	1 2 3 4 5
4	**GRAFIČKI ELEMENTI U SUSTAVU** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se	

	slažem)	
26	Grafičko oblikovanje sustava je, općenito govoreći, kvalitetno:	1 2 3 4 5
27	Sučelje je prilagođeno korisniku:	1 2 3 4 5
28	Vizualizacija savladanosti (lijevi stupac) pomaže u praćenju savladanosti domene:	1 2 3 4 5
29	Vizualizacija (mapa) domene znanja pomaže mi u odabiru pojmova za učenje/ponavljanje:	1 2 3 4 5
30	Fontovi koji se koriste u sustavu su dobro odabrani:	1 2 3 4 5
31	Slike koje ilustriraju sadržaje za učenje čine se kvalitetnima:	1 2 3 4 5
32	Slika koje ilustriraju sadržaje za učenje trebalo bi biti više:	1 2 3 4 5
5	**SUGESTIJE (samo grupa Tutoring+DM)** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se slažem)	
33	Zadovoljni ste načinom na koji su prezentirane sugestije (vizualno):	1 2 3 4 5
34	Općenito govoreći, sugestije su koristan mehanizam za pomoć pri odabiru pojmova za učenje/ponavljanje:	1 2 3 4 5
35	Sugestije su imale pozitivan utjecaj na Vaš proces učenja (vrijeme/brzina savladavanja pojmova):	1 2 3 4 5
36	Sugestije su utjecale na moju odluku o odabiru sljedećeg pojma (koristili ste ih/niste ih koristili, tj. klikali):	1 2 3 4 5
37	Pojmovi ponuđeni u sugestijama činili su vam se smislenima/logičnima u odnosnu na pojam na kojem ste se trenutno nalazili i trenutnu savladanost domene:	1 2 3 4 5
38	Sugestije "Što učiti PRIJE" odabranog pojma bile su korisne:	1 2 3 4 5
39	Sugestije "Što učiti POSLIJE" odabranog pojma bile su korisne:	1 2 3 4 5
6	**NAVIGACIJA** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se slažem)	
40	Struktura pojmova je dobro organizirana:	1 2 3 4 5
41	Strukturu pojmova je lako pretraživati:	1 2 3 4 5
42	Prelasci s pojma na pojam su jednostavni i brzi:	1 2 3 4 5
43	Struktura sadrži prevelik broj pojmova:	1 2 3 4 5
44	Hijerarhija je prikladan način prikaza strukture pojmova:	1 2 3 4 5
45	Prelazak iz bilo kojeg dijela sustava u bilo koji drugi dio sustava je jednostavan i brz:	1 2 3 4 5
46	Pristup različitim uslugama koje sustav nudi je jednostavan i brz:	1 2 3 4 5
47	Objektima sustava se može lako manipulirati:	1 2 3 4 5
48	Opcije za selekciju objekata lako su vidljive:	1 2 3 4 5
49	Instrukcije za korištenje sustava su jasne i lako dostupne:	1 2 3 4 5
7	**FLEKSIBILNOST** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se slažem)	
50	Sustav omogućuje jednake prilike za korisnike sa različitim razinama predznanja:	1 2 3 4 5
51	Sustav omogućuje poveznice i rad sa materijalima izvan sustava:	1 2 3 4 5
52	Sustav je prilagođen i naprednim korisnicima jer ima i materijale za naprednije:	1 2 3 4 5
53	Sadržaj u sustavu prilagođava se mojim potrebama:	1 2 3 4 5
54	Sustav mi predlaže tijek učenja ovisno o mojem znanju:	1 2 3 4 5
8	**ISKUSTVO/KORISNOST** 1 (uopće se ne slažem), 2 (uglavnom se ne slažem), 3 (niti se slažem niti se ne slažem), 4 (uglavnom se slažem), 5 (u potpunosti se slažem)	

55	Sviđa mi se koristiti sustav:	1 2 3 4 5
56	Moja iskustva u radu sa sustavom su pozitivna:	1 2 3 4 5
57	Rad sa sustavom je koristan za usvajanje znanja iz kolegija:	1 2 3 4 5
58	Sustav bi se trebao koristiti i u sklopu drugih kolegija:	1 2 3 4 5
59	Koristeći sustav brže usvajam gradivo:	1 2 3 4 5
60	Lakše učim koristeći sustav nego učeći na klasičan način:	1 2 3 4 5
9	**KOMENTARI**	
61	Koje promjene bi, po Vašem mišljenju, trebalo uvesti u sustav da bi se njegova kvaliteta poboljšala?	

Privitak 4. Rezultati ankete za domenu „Osnovne web tehnologije“

		KONTROLNA SKUPINA							EKSPERIMENTALNA SKUPINA						
0	Broj anketa	**22**							**28**						
	OPĆE INFORMACIJE														
1	Spol	**M**	9	41%	**Ž**	13	59%		**M**	12	43%	**Ž**	16	57%	
2	Starost	**M**	0		**Ž**	0			**M**	0		**Ž**	0		
3	Razina studija (preddiplomski/diplomski)	**PD**	22	100%	**D**	0	0%		**PD**	28	100%	**D**	0		
4	Studijska grupa	**1P**	16	73%	**2P**	6	27%		**1P**	24	86%	**2P**	4	14%	
5	Studijska kombinacija														
	NAČIN KORIŠTENJA SUSTAVA	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
6	Koliko ste često koristili sustav za e-učenje pripremajući se za kolokvij iz kolegija Dinamičke web aplikacije 1:	14%	23%	32%	32%				4%	14%	43%	39%			
7	Od kuda ste najčešće pristupali sustavu:	95%	5%	0%	0%				100%	0%	0%	0%			
8	Koji ste modul sustava najćešće koristili:	45%	55%						54%	46%					
9	Koji je modul, po Vašem mišljenju, najkorisniji:	41%	59%						50%	50%					
10	Koji je modul, po Vašem mišljenju, najmanje koristan:	5%	9%	86%					4%	7%	89%				
11	Jeste li koristili i neku drugu domenu osim Dinamičke web aplikacije 1:	50%	50%						71%	29%					
	SADRŽAJI ZA UČENJE	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
12	Sadržaj materijala za učenje je, općenito gledajući, kvalitetan:	0%	0%	9%	55%	36%	4,27	0,62	0%	11%	11%	46%	32%	4,00	0,93
13	Sadržaj je prezentiran na jasan način:	0%	0%	18%	55%	27%	4,09	0,67	0%	7%	32%	29%	32%	3,86	0,95
14	Predstavljeni sadržaji su jasno zaokružene cjeline:	0%	9%	14%	41%	36%	4,05	0,93	0%	0%	18%	46%	36%	4,18	0,71
15	Prezentirani sadržaji su međusobno povezani i nadopunjuju se:	0%	0%	14%	45%	41%	4,27	0,69	0%	0%	18%	39%	43%	4,25	0,74
16	Prezentiran je najnoviji sadržaj (najnovije informacije) iz područja:	0%	0%	5%	55%	41%	4,36	0,57	0%	0%	18%	64%	18%	4,00	0,60

17	Problemi koji su izloženi u materijalima sagledani su iz različitih stajališta, ne samo sa jednog:	0%	0%	23%	36%	41%	4,18	0,78	0%	14%	18%	54%	14%	3,68	0,89
18	Predstavljeni sadržaji su previše detaljni:	5%	36%	36%	18%	5%	2,82	0,94	11%	36%	36%	11%	7%	2,68	1,04
19	Odabrane teme su raznovrsne:	0%	0%	9%	55%	36%	4,27	0,62	0%	0%	18%	54%	29%	4,11	0,67
20	Sadržaj unutar pojedinih tema dobro strukturiran i logički predstavljen:	0%	9%	18%	45%	27%	3,91	0,90	0%	4%	21%	54%	21%	3,93	0,75
21	Predstavljene teme su relevantne s obzirom na ishode učenja ovog kolegija:	0%	14%	14%	32%	41%	4,00	1,04	0%	14%	18%	43%	25%	3,79	0,98
22	U materijalima je jasno označen sadržaj zaštićen autorskim pravom (citati, poveznice):	0%	5%	18%	32%	45%	4,18	0,89	0%	7%	32%	21%	39%	3,93	1,00
23	Sadržaj me motivirao na učenje:	0%	9%	27%	45%	18%	3,73	0,86	7%	14%	50%	18%	11%	3,11	1,01
24	Prezentirani sadržaj mi je bio koristan:	0%	0%	18%	41%	41%	4,23	0,73	4%	4%	29%	39%	25%	3,79	0,98
	GRAFIČKI ELEMENTI U SUSTAVU	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
25	Grafičko oblikovanje sustava je, općenito govoreći, kvalitetno:	0%	0%	14%	45%	41%	4,27	0,69	0%	11%	32%	32%	25%	3,71	0,96
26	Sučelje je prilagođeno korisniku:	0%	5%	9%	50%	36%	4,18	0,78	0%	11%	29%	36%	25%	3,75	0,95
27	Vizualizacija savladanosti (lijevi stupac) pomaže u praćenju savladanosti domene:	0%	0%	5%	36%	59%	4,55	0,58	0%	0%	18%	39%	43%	4,25	0,74
28	Vizualizacija (mapa) domene znanja pomaže mi u odabiru pojmova za učenje/ponavljanje:	0%	0%	9%	45%	45%	4,36	0,64	0%	4%	25%	29%	43%	4,11	0,90
29	Fontovi koji se koriste u sustavu su dobro odabrani:	0%	0%	5%	50%	45%	4,41	0,58	0%	0%	11%	46%	43%	4,32	0,66
30	Slike koje ilustriraju sadržaje za učenje čine se kvalitetnima:	5%	5%	5%	36%	50%	4,23	1,04	4%	0%	25%	50%	21%	3,86	0,87
31	Slika koje ilustriraju sadržaje za učenje trebalo bi biti više:	0%	5%	32%	36%	27%	3,86	0,87	0%	0%	25%	50%	25%	4,00	0,71
	SUGESTIJE	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
32	Zadovoljni ste načinom na koji su prezentirane sugestije (vizualno):								0%	4%	36%	46%	14%	3,71	0,75

33	Općenito govoreći, sugestije su koristan mehanizam za pomoć pri odabiru pojmova za učenje/ponavljanje:								0%	0%	39%	50%	11%	3,71	0,65
34	Sugestije su imale pozitivan utjecaj na Vaš proces učenja (vrijeme/brzina savladavanja pojmova):								4%	18%	54%	25%	0%	3,00	0,76
35	Sugestije su utjecale na moju odluku o odabiru sljedećeg pojma (koristili ste ih/niste ih koristili, tj. klikali):								11%	14%	32%	36%	7%	3,14	1,09
36	Pojmovi ponuđeni u sugestijama činili su vam se smislenima/logičnima u odnosnu na pojam na kojem ste se trenutno nalazili i trenutnu savladanost domene:								0%	4%	32%	57%	7%	3,68	0,66
37	Sugestije "Što učiti PRIJE" odabranog pojma bile su korisne:								0%	11%	43%	39%	7%	3,43	0,78
38	Sugestije "Što učiti POSLIJE" odabranog pojma bile su korisne:								4%	11%	46%	29%	11%	3,32	0,93
	NAVIGACIJA	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
39	Struktura pojmova je dobro organizirana:	0%	0%	14%	50%	36%	4,23	0,67	0%	0%	14%	57%	29%	4,14	0,64
40	Strukturu pojmova je lako pretraživati:	0%	9%	14%	50%	27%	3,95	0,88	0%	11%	18%	39%	32%	3,93	0,96
41	Prelasci s pojma na pojam su jednostavni i brzi:	0%	14%	18%	41%	27%	3,82	0,98	0%	14%	14%	29%	43%	4,00	1,07
42	Struktura sadrži prevelik broj pojmova:	14%	18%	32%	27%	9%	3,00	1,17	0%	11%	54%	21%	14%	3,39	0,86
43	Hijerarhija je prikladan način prikaza strukture pojmova:	0%	0%	18%	41%	41%	4,23	0,73	0%	0%	11%	39%	50%	4,39	0,67
44	Prelazak iz bilo kojeg dijela sustava u bilo koji drugi dio sustava je jednostavan i brz:	0%	18%	18%	45%	18%	3,64	0,98	7%	0%	29%	36%	29%	3,79	1,08
45	Pristup različitim uslugama koje sustav nudi je jednostavan i brz:	0%	5%	23%	45%	27%	3,95	0,82	0%	11%	29%	39%	21%	3,71	0,92
46	Objektima sustava se može lako manipulirati:	0%	0%	32%	41%	27%	3,95	0,77	4%	7%	32%	39%	18%	3,61	0,98

47	Opcije za selekciju objekata lako su vidljive:	0%	0%	0%	73%	27%	4,27	0,45	0%	11%	14%	50%	25%	3,89	0,90
48	Instrukcije za korištenje sustava su jasne i lako dostupne:	0%	0%	5%	50%	45%	4,41	0,58	4%	4%	18%	43%	32%	3,96	0,98
	FLEKSIBILNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
49	Sustav omogućuje jednake prilike za korisnike sa različitim razinama predznanja:	0%	0%	18%	50%	32%	4,14	0,69	4%	4%	18%	36%	39%	4,04	1,02
50	Sustav omogućuje poveznice i rad sa materijalima izvan sustava:	0%	0%	0%	50%	50%	4,50	0,50	4%	4%	14%	36%	43%	4,11	1,01
51	Sustav je prilagođen i naprednim korisnicima jer ima i materijale za naprednije:	0%	5%	23%	41%	32%	4,00	0,85	4%	7%	43%	25%	21%	3,54	1,02
52	Sadržaj u sustavu prilagođava se mojim potrebama:	0%	9%	18%	55%	18%	3,82	0,83	0%	11%	29%	36%	25%	3,75	0,95
53	Sustav mi predlaže tijek učenja ovisno o mojem znanju:	5%	0%	27%	45%	23%	3,82	0,94	0%	11%	32%	36%	21%	3,68	0,93
	ISKUSTVO/KORISNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
54	Sviđa mi se koristiti sustav:	0%	5%	5%	45%	45%	4,32	0,76	7%	7%	7%	32%	46%	4,04	1,21
55	Moja iskustva u radu sa sustavom su pozitivna:	0%	0%	5%	50%	45%	4,41	0,58	0%	11%	18%	32%	39%	4,00	1,00
56	Rad sa sustavom je koristan za usvajanje znanja iz kolegija:	0%	0%	23%	23%	55%	4,32	0,82	0%	7%	29%	25%	39%	3,96	0,98
57	Sustav bi se trebao koristiti i u sklopu drugih kolegija:	0%	5%	5%	9%	82%	4,68	0,76	0%	4%	29%	18%	50%	4,14	0,95
58	Koristeći sustav brže usvajam gradivo:	0%	0%	27%	23%	50%	4,23	0,85	4%	0%	36%	25%	36%	3,89	1,01
59	Lakše učim koristeći sustav nego učeći na klasičan način:	5%	9%	5%	32%	50%	4,14	1,14	4%	7%	18%	43%	29%	3,86	1,03
	KOMENTARI	**BROJ**							**BROJ**						
60	Koje promjene bi, po Vašem mišljenju, trebalo uvesti u sustav da bi se njegova kvaliteta poboljšala?	7							12						

Privitak 5. Rezultati ankete za domenu „Uzorci dizajna“

		KONTROLNA SKUPINA							EKSPERIMENTALNA SKUPINA						
0	Broj anketa	**19**							**28**						
	OPĆE INFORMACIJE														
1	Spol	**M**	9	47%	**Ž**	10	53%		**M**	13	46%	**Ž**	15	54%	
2	Starost	**M**	21		**Ž**	21			**M**	22		**Ž**	21		
3	Razina studija (preddiplomski/diplomski)	**P**	19		**D**	0			**P**	28		**D**	0		
4	Studijska grupa	**1P**	15		**2P**	4			**1P**	25		**2P**	3		
5	Studijska kombinacija														
	NAČIN KORIŠTENJA SUSTAVA	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
6	Koliko ste često koristili sustav za e-učenje pripremajući se za kolokvij iz kolegija Dinamičke web aplikacije 1:	16%	37%	32%	16%				7%	43%	14%	36%			
7	Od kuda ste najčešće pristupali sustavu:	95%	0%	0%	5%				100%	0%	0%	0%			
8	Koji ste modul sustava najćešće koristili:	32%	68%	0%	0%				39%	61%	0%	0%			
9	Koji je modul, po Vašem mišljenju, najkorisniji:	42%	58%	0%	0%				61%	39%	0%	0%			
10	Koji je modul, po Vašem mišljenju, najmanje koristan:	5%	0%	95%	0%				4%	7%	89%	0%			
11	Jeste li koristili i neku drugu domenu osim Dinamičke web aplikacije 1:	84%	16%	0%	0%				89%	11%	0%	0%			
	SADRŽAJI ZA UČENJE	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
12	Sadržaj materijala za učenje je, općenito gledajući, kvalitetan:	0%	0%	16%	37%	47%	4,28	0,73	0%	4%	4%	54%	39%	4,33	0,75
13	Sadržaj je prezentiran na jasan način:	0%	5%	5%	42%	47%	4,28	0,80	0%	4%	11%	46%	39%	4,28	0,80
14	Predstavljeni sadržaji su jasno zaokružene cjeline:	0%	11%	11%	37%	42%	4,11	0,99	0%	4%	11%	36%	50%	4,28	0,80
15	Prezentirani sadržaji su međusobno povezani i nadopunjuju se:	0%	0%	5%	42%	53%	4,50	0,60	0%	0%	11%	43%	46%	4,28	0,65
16	Prezentiran je najnoviji sadržaj (najnovije informacije) iz područja:	0%	0%	16%	53%	32%	4,17	0,69	0%	0%	14%	32%	54%	4,28	0,73

17	Problemi koji su izloženi u materijalima sagledani su iz različitih stajališta, ne samo sa jednog:	0%	5%	16%	32%	47%	4,17	0,90	0%	7%	25%	36%	32%	3,94	0,97
18	Predstavljeni sadržaji su previše detaljni:	5%	21%	47%	16%	11%	2,94	0,91	4%	21%	43%	18%	14%	3,11	0,99
19	Odabrane teme su raznovrsne:	0%	0%	5%	32%	63%	4,56	0,60	0%	11%	18%	32%	39%	3,83	1,07
20	Sadržaj unutar pojedinih tema dobro strukturiran i logički predstavljen:	0%	5%	5%	68%	21%	4,06	0,70	0%	7%	21%	46%	25%	3,94	0,91
21	Predstavljene teme su relevantne s obzirom na ishode učenja ovog kolegija:	0%	11%	16%	37%	37%	4,00	1,00	0%	14%	14%	39%	32%	3,94	0,97
22	U materijalima je jasno označen sadržaj zaštićen autorskim pravom (citati, poveznice):	0%	0%	16%	42%	42%	4,28	0,73	0%	0%	25%	32%	43%	4,00	0,82
23	Sadržaj me motivirao na učenje:	5%	11%	32%	21%	32%	3,61	1,21	7%	14%	25%	25%	29%	3,78	1,31
24	Prezentirani sadržaj mi je bio koristan:	0%	11%	16%	21%	53%	4,11	1,05	0%	14%	14%	36%	36%	3,89	1,05
	GRAFIČKI ELEMENTI U SUSTAVU	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
25	Grafičko oblikovanje sustava je, općenito govoreći, kvalitetno:	0%	0%	16%	26%	58%	4,39	0,76	0%	7%	18%	39%	36%	4,00	0,94
26	Sučelje je prilagođeno korisniku:	0%	5%	11%	42%	42%	4,22	0,85	0%	4%	18%	54%	25%	3,89	0,81
27	Vizualizacija savladanosti (lijevi stupac) pomaže u praćenju savladanosti domene:	0%	0%	0%	37%	63%	4,67	0,47	0%	7%	21%	32%	39%	4,06	1,03
28	Vizualizacija (mapa) domene znanja pomaže mi u odabiru pojmova za učenje/ponavljanje:	0%	0%	0%	53%	47%	4,50	0,50	0%	11%	14%	36%	39%	3,78	1,03
29	Fontovi koji se koriste u sustavu su dobro odabrani:	0%	0%	11%	21%	68%	4,61	0,68	0%	0%	7%	36%	57%	4,39	0,68
30	Slike koje ilustriraju sadržaje za učenje čine se kvalitetnima:	0%	11%	11%	37%	42%	4,11	0,99	0%	7%	25%	36%	32%	3,89	0,99
31	Slika koje ilustriraju sadržaje za učenje trebalo bi biti više:	0%	0%	16%	53%	32%	4,11	0,66	0%	11%	32%	43%	14%	3,61	0,83
	SUGESTIJE	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
32	Zadovoljni ste načinom na koji su prezentirane sugestije (vizualno):								4%	4%	36%	29%	29%	3,72	1,10

33	Općenito govoreći, sugestije su koristan mehanizam za pomoć pri odabiru pojmova za učenje/ponavljanje:								0%	7%	32%	36%	25%	4,06	0,78
34	Sugestije su imale pozitivan utjecaj na Vaš proces učenja (vrijeme/brzina savladavanja pojmova):								7%	18%	36%	21%	18%	3,61	1,11
35	Sugestije su utjecale na moju odluku o odabiru sljedećeg pojma (koristili ste ih/niste ih koristili, tj. klikali):								11%	32%	29%	7%	21%	3,22	1,18
36	Pojmovi ponuđeni u sugestijama činili su vam se smislenima/logičnima u odnosnu na pojam na kojem ste se trenutno nalazili i trenutnu savladanost domene:								0%	14%	39%	29%	18%	3,67	0,88
37	Sugestije "Što učiti PRIJE" odabranog pojma bile su korisne:								0%	14%	46%	11%	29%	3,83	1,01
38	Sugestije "Što učiti POSLIJE" odabranog pojma bile su korisne:								0%	7%	54%	21%	18%	3,61	0,89
	NAVIGACIJA	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
39	Struktura pojmova je dobro organizirana:	0%	5%	11%	37%	47%	4,22	0,85	0%	4%	14%	46%	36%	4,06	0,85
40	Strukturu pojmova je lako pretraživati:	5%	5%	21%	32%	37%	3,83	1,12	4%	7%	18%	25%	46%	3,83	1,21
41	Prelasci s pojma na pojam su jednostavni i brzi:	0%	11%	26%	26%	37%	3,83	1,01	0%	11%	21%	43%	25%	3,72	1,10
42	Struktura sadrži prevelik broj pojmova:	11%	21%	21%	26%	21%	3,17	1,26	11%	21%	32%	21%	14%	3,44	1,17
43	Hijerarhija je prikladan način prikaza strukture pojmova:	0%	5%	21%	32%	42%	4,11	0,94	0%	0%	14%	29%	57%	4,28	0,80
44	Prelazak iz bilo kojeg dijela sustava u bilo koji drugi dio sustava je jednostavan i brz:	0%	11%	5%	58%	26%	4,00	0,88	4%	14%	14%	36%	32%	3,83	1,12
45	Pristup različitim uslugama koje sustav nudi je jednostavan i brz:	0%	11%	21%	42%	26%	3,83	0,96	0%	4%	21%	46%	29%	4,06	0,70
46	Objektima sustava se može lako manipulirati:	0%	5%	37%	32%	26%	3,72	0,87	4%	4%	36%	32%	25%	3,83	1,01

47	Opcije za selekciju objekata lako su vidljive:	0%	0%	16%	47%	37%	4,17	0,69	0%	7%	21%	50%	21%	3,83	0,90
48	Instrukcije za korištenje sustava su jasne i lako dostupne:	0%	0%	16%	26%	58%	4,39	0,76	4%	0%	21%	39%	36%	4,00	1,00
	FLEKSIBILNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
49	Sustav omogućuje jednake prilike za korisnike sa različitim razinama predznanja:	0%	5%	16%	32%	47%	4,17	0,90	4%	7%	14%	36%	39%	4,11	1,10
50	Sustav omogućuje poveznice i rad sa materijalima izvan sustava:	5%	0%	16%	26%	53%	4,17	1,07	0%	18%	7%	29%	46%	4,00	1,11
51	Sustav je prilagođen i naprednim korisnicima jer ima i materijale za naprednije:	0%	5%	16%	26%	53%	4,28	0,93	0%	4%	25%	36%	36%	4,00	0,88
52	Sadržaj u sustavu prilagođava se mojim potrebama:	0%	0%	21%	47%	32%	4,11	0,74	0%	11%	21%	39%	29%	4,00	0,94
53	Sustav mi predlaže tijek učenja ovisno o mojem znanju:	0%	16%	11%	42%	32%	3,89	1,05	4%	18%	18%	29%	32%	3,89	1,15
	ISKUSTVO/KORISNOST	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**	**1**	**2**	**3**	**4**	**5**	**AS**	**SD**
54	Sviđa mi se koristiti sustav:	0%	11%	16%	16%	58%	4,17	1,07	7%	4%	11%	39%	39%	4,06	1,35
55	Moja iskustva u radu sa sustavom su pozitivna:	0%	0%	21%	26%	53%	4,33	0,82	4%	4%	18%	29%	46%	4,17	1,21
56	Rad sa sustavom je koristan za usvajanje znanja iz kolegija:	0%	11%	5%	32%	53%	4,28	0,99	0%	14%	14%	25%	46%	4,17	1,07
57	Sustav bi se trebao koristiti i u sklopu drugih kolegija:	0%	0%	16%	37%	47%	4,33	0,75	0%	7%	14%	29%	50%	4,22	1,08
58	Koristeći sustav brže usvajam gradivo:	0%	11%	11%	32%	47%	4,11	0,99	4%	4%	21%	36%	36%	4,00	1,15
59	Lakše učim koristeći sustav nego učeći na klasičan način:	0%	11%	16%	32%	42%	4,00	1,00	4%	7%	18%	43%	29%	3,83	1,07
	KOMENTARI	**BROJ**							**BROJ**						
60	Koje promjene bi, po Vašem mišljenju, trebalo uvesti u sustav da bi se njegova kvaliteta poboljšala?	7							8						

Životopis

Igor Jugo je rođen 26. rujna 1978. godine u Rijeci. Nakon završetka osnovnoškolskog i srednješkolskog obrazovanja (Elektrotehnička škola u Rijeci, smjer: tehničar za elektroniku) upisuje Pedagoški (danas: Filozofski) fakultet u Rijeci, smjer: pedagogija i informatika. Diplomirao je 2003. godine obranom diplomskog rada pod nazivom „Razvoj web trgovine korištenjem ASP.NET platforme“ (mentor: prof. Mario Radovan). Od travnja 2004. zaposlen je na Odsjeku za informatiku Filozofskog fakulteta, na radnom mjestu asistenta. Aktivno sudjeluje u pripremi i izvođenju vježbi iz kolegija na preddiplomskom studiju (Računalne mreže, Programiranje za internet). Od travnja 2008. godina zaposlen je na Odjelu za informatiku Sveučilišta u Rijeci gdje izvodi vježbe iz kolegija na prediplomskom (Programiranje 1, Dinamičke web aplikacije 1, Dinamičke web aplikacije 2, Informatika), te diplomskom studiju (Elektroničko gospodarstvo). Od 2011. godine povjerena su mu predavanja iz kolegija Dinamičke web aplikacije 2.
Magistarski studij informatike na Fakultetu organizacije i informatike u Varaždinu upisao je 2005. godine, a završio obranom magistarskog rada pod naslovom „Analiza i evaluacija tehnika za poboljšanje performansi web aplikacija“ (mentor. prof. Dragutin Kermek). Poslijediplomski doktorski studij informacijskih znanosti upisao je na Fakuletu organizacije i informatike u Varaždinu, kojeg nastavlja na Odjelu za informatiku Sveučilišta u Rijeci.
Sudjelovao je u više nacionalnih i europskih znanstvenih projekata. Sudjelovao je na brojnim međunarodnim znanstvenim konferencijama, te samostalno i u suautorstvu objavio više znanstvenih i stručnih radova. Znanstveni interesi obuhvaćaju područje web inženjerstva, te dubinske analize podataka u obrazovanju. Ima iskustva u pisanju i prijavi europskih (7. okvirni program, Centri izvrsnosti), te nacionalnih (MZOS, HZZZ) znanstvenih projekata.

U nastavku slijedi popis objavljenih znanstvenih radova.

1. Jugo, Igor; Kovačić, Božidar; Slavuj, Vanja. Increasing the Adaptivity of an Intelligent Tutoring System with Educational Data Mining: a System Overview. // International Journal of Emerging Technologies in Learning. (2016), 3; 67-70. (članak, znanstveni)

2. Jugo, Igor; Kovačić, Božidar; Tijan, Edvard. Cluster analysis of student activity in a web-based intelligent tutoring system. // Pomorstvo : journal of maritime studies. 29 (2015) , 1; 80-88 (članak, znanstveni).
3. Jugo, Igor; Kermek, Dragutin; Meštrović, Ana. Analysis and Evaluation of Web Application Performance Enhancement Techniques. // Lecture Notes in Computer Science - LNCS. 8541 (2014) ; 40-56 (predavanje,međunarodna recenzija,objavljeni rad, znanstveni).
4. 1. Jugo, Igor; Kovačić, Božidar; Slavuj, Vanja. Integrating a Web-based ITS with DM tools for Providing Learning Path Optimization and Visual Analytics // Proceedings of the 8th International Conference on Educational data mining / J.G. Boticario, O.C. Santos, C. Romero, M. Pechenizkiy, A. Merceron, P. Mitros, J.M. Luna, C. Mihaescu, P. Moreno, A. Hershkovitz, S. Ventura, M. Desmarais (ur.). Madrid, 2015. 574-575 (poster,međunarodna recenzija,objavljeni rad,znanstveni).
5. Slavuj, Vanja; Kovačić, Božidar; Jugo, Igor. Intelligent Tutoring Systems for Language Learning // Proceeding of the 38th International Convention MIPRO / Petar Biljanović (ur.). Rijeka : Croatian Society for Information and Communication Technology, Electronics and Microelectronics - MIPRO, 2015. 915-920 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
6. Jugo, Igor; Kovačić, Božidar; Slavuj, Vanja. Using Data Mining for Learning Path Recommendation and Visualization in an Intelligent Tutoring System // Proceedings of the 37th International Convention MIPRO 2014 / Biljanović, Petar (ur.). Opatija : Croatian Society for Information and Communication Technology, Electronics and Microelectronics - MIPRO, 2014. 1042-1047 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
7. Jugo, Igor; Kovačić, Božidar; Slavuj, Vanja. A proposal for a web based educational data mining and visualization system // Proceedings of the 5th International Conference on Information Technologies and Information Society ITIS 2013 / Levnajić, Zoran (ur.). Dolenjske Toplice : Faculty of Information Studies in Novo mesto, Slovenia, 2013. 59-64 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
8. Slavuj, Vanja; Kovačić, Božidar; Jugo, Igor. Assessing user satisfaction with a system for e- learning based on dialogue // La médiation numérique : renouvellement et diversification des pratiques / Boustany, Joumana ; Broudoux, Evelyne ; Chartron,

Ghislaine (ur.). Bruxelles : De Boeck, 2013. (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).

9. Kovačić, Božidar; Jugo, Igor; Slavuj, Vanja. Improvement of system for distance learning based on dialogue by appliance of statistical analysis // MIPRO 2012 Jubilee 35th International Convention / Biljanović, Petar (ur.). Rijeka : Croatian Society for Information and Communication Technology, Electronics and Microelectronics - MIPRO, 2012. 1794-1797 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
10. Babić, Arsen; Poščić, Patrizia; Jugo, Igor. Bibloogle – sustav za upravljanje i pretragu podataka // MIPRO, 2011 Proceedings of the 34th International Convention, Computer in Education / Čičin-Šain, Marina ; Uroda, Ivan ; Turčić Prstačić, Ivana ; Sluganović, Ivanka (ur.). Zagreb : MIPRO, 2011. 495-500 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
11. Maržić, Sanjin; Jugo, Igor; Radovan, Mario. Developing dynamic Web applications:“Exam scheduler” // MIPRO, 2011 Proceedings of the 34th International Convention. Zagreb, 2011. 1127-1132 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
12. Skendžić, Aleksandar; Kovačić, Božidar; Jugo, Igor. Decrease of Information technology expenses by emulator programs on Windows and Linux platforms // Digital economy 8 th ALADIN - DE & ISS & miproBIS GLGPS & SP / Čišić, Dragan ; Hutinski, Željko ; Baranović, Mirta ; Mauher, Mladen ; Ordanić, Lea (ur.). Zagreb : MIPRO, 2011. 41-44 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
13. Brkić, Marija; Matetić, Maja; Jugo, Igor. Corpora Building and Processing // Proceedings of the2nd International Conference on Human System Interaction (HSI'09) / Lo Bello, Lucia ; Iannizzotto, Giancarlo (ur.). 2009. 251-254 (poster,međunarodna recenzija,objavljeni rad,znanstveni).
14. Kovačić, Božidar; Jugo, Igor. Applying a Distance Learning System Based on Dialogue in e-commerce // Proceedings of 32nd international conference on information and communication technology, electronics and microelectornics, MIPRO, VOL V., DE $ ISS $ miproBIS $ LG $ SP / Dragan Čišić, Željko Hutinski, Mirta Baranović, Mladen Mauher, Veljko Dragšić (ur.). Opatija : MIPRO, Rijeka, 2009. 36-39 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
15. Jugo, Igor. Online CASE Tool for Development of Web Applications // Proceedings of the 19th Central European Conference on Information and Intelligent Systems /

Aurer, Boris Bača, Miroslav Rabuzin, Kornelije (ur.). Varaždin : Faculty of Organization and Informatics Varaždin, University of Zagreb, Pavlinska 2, 42000 Varaždin, Croatia, 2008. 641-647 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).

16. Jugo, Igor; Radovan, Mario. New Technologies for Web Applications // Proceedings of the 18th International Conference on Information and Intelligent Systems. Varaždin, 2007. 193-198 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
17. Radovan, Mario; Jugo, Igor. Computation and Evolution // Proceedings of the 18th International Conference on Information and Intelligent Systems. Varaždin, 2007. 227-234 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).
18. Radovan, Mario; Jugo, Igor. On Data Gathering and Surveillance // Proceedings of the 17th International Conference on Information and Intelligent Systems. Varaždin, 2006. 435-440 (predavanje,međunarodna recenzija,objavljeni rad,znanstveni).