

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Nenad Babić Bodiroža

**Spletna obdelava rezultatov  
ocenjevanja**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM  
PRVE STOPNJE  
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2018

COPYRIGHT. Rezultati diplomske naloge so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavo in koriščenje rezultatov diplomske naloge je potrebno pisno privoljenje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

*Besedilo je oblikovano z urejevalnikom besedil L<sup>A</sup>T<sub>E</sub>X.*

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Za ocenjevanje pedagoških obveznosti se uporablja različna programska oprema, ki omogoča vnos in obdelavo podatkov ter izvoz rezultatov. Kandidat naj v diplomskem delu zasnuje in implementira spletno aplikacijo za uvoz datotek v formatih .xlsx, .csv in .txt ter omogoči nadaljnjo obdelavo rezultatov sprotnih obveznosti in izpitov. Analiza naj bo predstavljena v tabelaričnih ali grafičnih oblikah za ogled, pripravo in izvoz v obliki poročila. Za implementacijo naj uporabi ogrodje Bootstrap, programski jezik php in podatkovno bazo MySQL. Delovanje in zahtevane funkcionalnosti naj bodo predstavljene za rezultate ocenjevanja sprotnih obveznosti (e-učilnica) in izpitov (studis) pri enem od predmetov na FRI.



*Zahvaljujem se mentorici doc. dr. Miri Trebar za vso pomoč, potrpežljivost in vodenje pri izdelavi diplomskega dela.*

*Zahvaljujem se tudi staršem, ki so mi omogočili študij in me med študijem vedno podpirali.*



# Kazalo

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Teoretične osnove</b>	<b>3</b>
2.1	Spletne tehnologije . . . . .	4
2.2	Orodja . . . . .	6
<b>3</b>	<b>Opis problema</b>	<b>9</b>
3.1	Pregled in analiza podatkov . . . . .	10
3.2	Obdelava podatkov . . . . .	14
3.3	Predstavitev podatkov . . . . .	15
<b>4</b>	<b>Implementacija</b>	<b>17</b>
4.1	Načrtovanje spletne aplikacije . . . . .	17
4.2	Podatkovna baza . . . . .	21
4.3	Implementacija spletne aplikacije . . . . .	25
4.4	Testiranje . . . . .	34
<b>5</b>	<b>Sklepne ugotovitve</b>	<b>39</b>
	<b>Literatura</b>	<b>42</b>





# Seznam uporabljenih kratic

kratica	angleško	slovensko
<b>AJAX</b>	Asynchronous JavaScript and XML	asinhroni JavaScript in XML
<b>CSS</b>	Cascading Style Sheets	kaskadna stilska predloga
<b>CSV</b>	Comma-separated values	vrednosti, ločene z vejico
<b>HTML</b>	Hyper Text Markup Language	jezik za označevanje nadbese- dila
<b>HTTP</b>	Hypertext Transfer Protocol	protokol za prenos nadbese- dila
<b>JSON</b>	JavaScript Object Notation	objektni zapis JavaScript
<b>MVC</b>	Model, View, Controller	model, pogled, krmilnik
<b>PDF</b>	Portable document format	format prenosnih datotek
<b>PDO</b>	PHP Data Object	podatkovni objekt PHP
<b>PHP</b>	Hypertext Preprocessor	strežniški skriptni program- ski jezik
<b>PNG</b>	Portable Network Graphics	bitni (rastrski) format slike
<b>SQL</b>	Structured Query Language	strukturirani povpraševalni jezik
<b>XAMPP</b>	Cross (X) platform Apache, MySQL, PHP and Perl	zbirka orodij za razvoj sple- tnih vsebin.
<b>XLSX</b>	Open XML Spreadsheet	odprtokodni format za raz- predelnice XML
<b>XML</b>	Extensible markup language	razširljivi označevalni jezik



# Povzetek

**Naslov:** Spletna obdelava rezultatov ocenjevanja

**Avtor:** Nenad Babić Bodiroža

Pri izvajanju pedagoškega procesa se na fakultetah pogosto srečujejo s problemom preprostega beleženja, obdelave, grafične predstavitve in izdelave poročil o rezultatih ocenjevanja. Ponavadi se to rešuje z vnašanjem študentskih rezultatov v različne informacijske sisteme fakultete, uporabo orodij za grafični prikaz, ali pa z beleženjem študentskih uspehov v dokumente. V diplomskem delu smo z uporabo spletnih tehnologij zasnovali in ustvarili aplikacijo, ki bo omogočala uvoz in obdelavo pridobljenih rezultatov, izdelavo poročil o študentskih uspehih in preprost način izvažanja podatkov v datoteke, za tiskanje ali uvoz v druge informacijske sisteme. Spletna aplikacija je bila zasnovana z uporabo odprtokodnega orodja Bootstrap, ki nam omogoča preprosto uporabo aplikacije na mobilnih napravah. Celotna rešitev je bila testirana z uporabo rezultatov ocenjevanja izbranega predmeta na Fakulteti za računalništvo in informatiko.

**Ključne besede:** spletna aplikacija, obdelava rezultatov, predstavitev podatkov, ocenjevanje.



# Abstract

**Title:** Web processing of grading results

**Author:** Nenad Babić Bodiroža

During the pedagogical process, the faculty personnel often encounters a problem of simple recording, processing, displaying and generating reports of student grading results. Such problems are commonly solved by importing the student results into multiple faculty information systems, using tools for graphical presentation or by writing and storing the results in documents. In the diploma thesis, we used different web technologies to design and create a web application, that would enable importing and processing of student data, report generation of student results and a simple way to export the data into appropriate files, that can be used for importing the student data into other information systems. The web application was designed with the usage of an open source framework Bootstrap, that allows the usage of the web application on mobile devices. The functionality of the application was tested with student data from the Faculty of Computer and Information Science.

**Keywords:** web application, data processing, data presentation, grading.



# Poglavje 1

## Uvod

Pri izvajanju pedagoškega procesa vse bolj pomembne postajajo rešitve, ki bi izvajalcem omogočile hitro in učinkovito ocenjevanje ter shranjevanje zgodovine študentskih dosežkov. Zaradi problematike preproste predstavitve, urejanja in obdelave večje vsote študentskih uspehov in rezultatov ocenjevanja se na fakultetah pojavlja potreba po celoviti rešitvi, ki bi na hiter in učinkovit način odpravila omenjene probleme. Rešitev bi poleg vseh funkcij morala omogočati tudi način prenosa obdelanih rezultatov med izdelano aplikacijo, elektronskim indeksom in spletno učilnico fakultete. Cilj diplomskega dela je zasnova in ustvarjanje spletne aplikacije, ki bo izvajalcem predmetov skupaj z drugimi pedagogi, ki so vključeni v proces izvajanja predmetov, omogočala enostaven način pregledovanja in obdelave rezultatov študentskega ocenjevanja. Spletna rešitev bo omogočala tudi enostaven in hiter način izvažanja urejenih rezultatov, kot pa tudi možnost izdelave poročil o študentskih uspehih in grafično predstavitev doseženih rezultatov pri izbranemu predmetu.

Spletna aplikacija bo temeljila na sodobnih spletnih tehnologijah, ki bodo z uporabo relacijske podatkovne baze omogočale realizacijo predhodno omejenih zahtev. Do spletne aplikacije oz. storitve bodo uporabniki dostopali preko spletnega brskalnika. Aplikacija bo podatke pridobivala z branjem uvoženih datotek pedagogov, ki bodo vsebovale rezultate ocenjevanj in izvajalčeve komentarje vezane na izvajanje predmeta. Hranjenje podatkov v

podatkovni bazi bo uporabnikom omogočalo tudi generiranje poročil o uspehih študentov, pregledovanje in primerjanje študentskih uspehov s predhodnimi izvajanji izbranih predmetov. Pedagogi bodo lahko podatke znotraj aplikacije tudi grafično predstavili in izvozili v slikovnem formatu za nadaljnjo uporabo.



# Poglavje 2

## Teoretične osnove

Pri izdelavi rešitve bomo uporabljali večje število različnih tehnologij in orodij, ki so primerna za razvoj spletnih aplikacij. Vizualna podoba grafičnega vmesnika bo realizirana z uporabo jezika za označevanje HTML [37], stiskih predlog CSS [36] in odprtokodnega orodja Bootstrap [2], ki naši aplikaciji doda odzivno "mobile-first" obliko [32, 20], ta oblika bo uporabnikom omogočala optimalen in prilagodljiv prikaz strani na vseh napravah. Programiranje dinamičnih elementov spletne aplikacije bo izvedeno s skriptnim jezikom JavaScript [17] v povezavi s knjižnico jQuery [19], ki se izvaja znotraj odjemalčevega oz. uporabnikovega spletnega brskalnika. Za obdelavo podatkov ter logiko aplikacije skrbi programski jezik PHP [27], ki nam bo tudi služil za pridobivanje, dostop in urejanje podatkovne baze MySQL [22] z uporabo poizvedb SQL [34]. Komunikacija s strežnikom in pridobivanje podatkov bo potekala z uporabo tehnologij Ajax [1]. Grafična predstavitev rezultatov ocenjevanja znotraj grafikonov nam bo omogočala JavaScript knjižnica Chart.js [4]. Izdelavo poročil v PDF formatu bomo dosegli z uporabo knjižnice FPDF [12]. Celotna aplikacija bo temeljila na arhitekturnem vzorcu MVC [21]. Aplikacijo bomo poganjali na strežniku Apache [16], podatki pa bodo shranjeni v podatkovni bazi MySQL [22]. Strežnik in podatkovna baza sta vsebovana znotraj skupine orodij imenovane XAMPP [38].

## 2.1 Spletne tehnologije

**HTML** (Hyper Text Markup Language) [37] je jezik za označevanje nadbese-dila, ki nam omogoča označevanje datotek HTML. Z uporabo jezika HTML v datoteki vzpostavimo strukturo spletne strani oz. dokumenta. Podatki in ele-menti, ki se bodo nahajali na spletni strani morajo biti v dokumentu HTML [14] označeni z ustreznimi značkami. Značke predstavljajo ukaze HTML, ki skrbijo za prikaz elementov na spletni strani. Poznamo dve vrsti značk, prva vrsta so značke, ki potrebujejo zaključno značko, drugo vrsto pa predstavljajo samostojne značke. Zaključne značke označimo enako kot začetno značko, z razliko dodajanja poševnice pred njeno ime. Najnovejša in trenutna verzija je HTML5 [13], ki jo podpirajo vsi modernejši brskalniki. Pri dostopu do spletne strani oz. aplikacije, brskalnik pridobi podatke iz dokumenta HTML in interpretira vsebino značk za prikaz elementov znotraj brskalnika.

**CSS** (Cascading Style Sheets) [36] je stilska predloga, ki opisuje vizualno podobo elementov v dokumentu HTML. Stilska predloga omogoča spremi-njanje vizualne podobe posameznih elementov HTML ali pa večje skupine elementov HTML, ki se nahajajo znotraj enakega razreda ali pa pripadajo istemu tipu značk. Najnovejša in trenutna verzija stilskih predlog se imenuje CSS3 [6].

**Bootstrap** [2] je zastojnsko odprtokodno HTML, CSS in JavaScript ogrodje, ki razvijalcem omogoča ustvarjanje odzivnih "mobile-first" spletnih strani. Ta način oblikovanja uporabnikom omogoča prilagajanje prikaza spletne strani na vseh napravah. Bootstrap temelji na novi CSS3 razporeditvi elementov, ki je sestavljena iz mrežnega sistema stolpcev, vrstic in vsebnikov [3, 10].

**JavaScript** [17] je skriptni programski jezik, ki se v večini primerov upora-blja pri dodajanju in programiranju dinamičnih elementov na spletni strani. JavaScript se podobno kot HTML in CSS izvaja znotraj brskalnika, ki do-stop do spletne strani.

**jQuery** [19] je najbolj znana odprtokodna knjižnica JavaScript, ki omogoča programiranje in izvajanje JavaScript programske kode z uporabo preprostejših sintaks.

**Chart.js** [4] je preprosta odprtokodna in prilagodljiva knjižnica JavaScript, ki omogoča izris interaktivnih dinamičnih grafikov in diagramov na HTML5 spletni strani.

**FileSaver.js** [9] je odprtokodna knjižnica JavaScript, ki omogoča preprosto shranjevanje elementov HTML5 v slikovni obliki.

**DataTables** [7] je vtičnik za JavaScript knjižnico jQuery, ki omogoča dodajanje naprednih funkcij, ki omogočajo interakcijo in manipuliranje podatkov znotraj tabel HTML.

**Ajax** [1] predstavlja skupek tehnologij spletnega razvoja, ki omogočajo izdelavo asinhronskih spletnih aplikacij. Ajax omogoča asinhrono komuniciranje brskalnika s strežnikom in pridobivanje in nalaganje podatkov v dokument HTML brez stalne potrebe po ponovnem osveževanju brskalnika.

**Font Awesome** [11] je odprtokodna zbirka ikon, ki nam omogoča njihovo postavljanje na spletno stran.

**PHP** (Hypertext Preprocessor) [27] je skriptni programski jezik namenjen razvoju spletnih rešitev, ki se primarno izvaja na strežniškem oz. zalednem delu aplikacije in ne v odjemalčevem brskalniku. PHP se izvede preden strežnik uporabnikovem brskalniku pošlje dokument HTML. Najnovejša in trenutna verzija programskega jezika je PHP 7.1.

**FPDF** (Free PDF) [12] je zastonski razred programskega jezika PHP, ki

omogoča dinamično izdelavo PDF [24] datotek.

**MVC** (Model - View - Controller) [21] je arhitektura, ki nam omogoča ločeno obdelavo podatkov za potrebe spletnih strani oz. aplikacij. Prvi del arhitekture MVC, **model** vsebuje poslovno logiko in način pridobivanja uporabniških podatkov, drugi del, **view** ima nalogo prikazovanja podatkov, ki smo jih predhodno obdelali, zadnji del imenovan **controller** pa je namenjen preslikovanju uporabniških ukazov v ukaze modela oz. pogleda.

## 2.2 Orodja

**XAMPP** (Cross platform Apache, MariaDB, PHP and Perl) [38] je odprtokodna zbirka programskih storitev, ki omogoča postavitev lokalnega spletnega strežnika Apache [16] za namene razvijanja spletnih aplikacij. XAMPP vsebuje spletni strežnik Apache, ki podpira podatkovno bazo MySQL [22], skriptni jezik PHP [27] in programski jezik Perl [25].

**MySQL** [22] je najbolj priljubljen odprtokodni sistem za upravljanje relacijskih podatkovnih baz, ki uporablja strukturiran povpraševalni jezik oz. SQL [34].

**PhpMyAdmin** [29] je odprtokodno zastojnsko orodje napisano v jeziku PHP, preko katerega lahko v brskalniku urejamo in pregledujemo relacijske podatkovne baze MySQL, phpMyAdmin je del programskega paketa XAMPP in nam omogoča dostop do baze, iz katere bo naša aplikacija črpala podatke.

**JetBrains PhpStorm** [18] je integrirano razvojno okolje (ang. integrated development environment) namenjeno razvijanju aplikacij PHP. Poleg programskega jezika PHP okolje nudi podporo tudi jezikom HTML, CSS

in JavaScript. Okolje s sprotnim pregledovanjem napak v kodi razvijalcem omogoča hitrejši in učinkovitejši razvoj spletnih aplikacij.



# Poglavje 3

## Opis problema

Pri izvajanju pedagoškega procesa se na fakultetah pogosto pojavlja problem pregledovanja, shranjevanja, urejanja in obdelave rezultatov študentskih uspehov. Pri analizi rezultatov sprotne delo in ocen študentov se pojavlja potreba po preprosti grafični predstavitvi pridobljenih rezultatov ter izdelavi poročil o uspešnosti študentov med izbranim študijskim letom za potrebe izboljšave učinkovitosti pedagoškega procesa. Opisane težave osebje velikokrat rešuje z uporabo različnih orodij, ki so pogosto zapletena in časovno potratna, ali pa je potrebno dodatno delo z ustvarjanjem in združevanjem nepovezanih datotek, ki služijo kot način hranjenja ter urejanja rezultatov ocenjevanja.

V pedagoškem procesu se točkovanje, ocenjevanje in označevanje prisotnosti študentov običajno vnaša v različne informacijske sisteme, ali pa se podatke s podpisovanjem beleži na papirnati list in označuje v tabele. Težave se pojavljajo tudi pri vpisovanju rezultatov ocenjevanja v elektronski indeks fakultete, saj je zaradi razpršenosti podatkov med večimi informacijskimi sistemi in listi vpisovanje teh rezultatov in ocen zelo zamudno.

V okviru diplomskega dela bomo razvili spletno aplikacijo, ki bo omogočala preprosto obdelavo rezultatov ocenjevanja. Ponujala bo podroben način prikazovanja in urejanja podatkov, vodenja evidence ter izdelave poročil študentskih uspehov skupaj z možnostjo izvažanja obdelanih podatkov v for-

matu primernem za uvoz v elektronski indeks fakultete. Zaradi dovolj preprostega prikazovanja informacij bo aplikacija omogočala tudi sprotni vpogled v uspešnost izvajanja predmeta in omogočala vpeljavo izboljšav v pedagoški proces. Aplikacija bo zaradi svoje preprostosti pedagogom omogočila hitrejši vnos in popravljanje ocen, to pa jim bo omogočilo več časa, ki ga bodo lahko namenili delu s študenti in drugim aktivnostim.

### 3.1 Pregled in analiza podatkov

Podatki za namene obdelave in prikazovanja bodo podani v večjem številu različnih datotek in formatih. Vsaka datoteka bo vsebovala splošne podatke o vpisanih študentih in njihove dosežene rezultate pri izbrani obveznosti. Podane bodo v različnih oblikah z ustreznimi podatki:

- datoteka s seznamom vpisanih študentov pri predmetu,
- datoteka s seznamom rezultatov sprotnih obveznosti študentov (pristnost, domače naloge, kvizi, seminarji), datoteka lahko vsebuje tudi rezultate kolokvijev in izpitov.
- datoteka s seznamom rezultatov za posamezni kolokvij (lahko vsebuje končne ocene),
- datoteka s seznamom rezultatov za posamezen izpitni rok v študijskem letu (lahko vsebuje končne ocene),
- datoteka s komentarji o izvajanju izbranega predmeta.

Datoteke s seznamami vpisanih študentov, rezultatov posameznih kolokvijev in izpitov bodo v prvi vrstici vedno vsebovale podatke o predmetu. Podatki vključujejo ime predmeta ter študijsko leto izvajanja predmeta.

Slika 3.1 prikazuje seznam vpisanih študentov, ki ga pridobimo z izvozom iz fakultetnega informacijskega sistema STUDIS [35] v formatu CSV (ang.



comma-separated values) [5].

Digitalna vezja (63718) 2017/18 - seznam študentov izvajanja					
Vpisna številka	Priimek, Ime	E-mail	Letnik program	Tip vpisa	Obiskuje predavanja
			3 BVS-RI LJ	01 - Prvi vpis v letnik	1
			2 BVS-RI LJ	01 - Prvi vpis v letnik	1
			2 BVS-RI LJ	01 - Prvi vpis v letnik	1

Prvo poslušanje	Sprotne obveznosti	Točke kolokvijev	Točke izpita/rok	Končna ocena
2017/18				
2017/18				
2017/18				

Slika 3.1: Seznam študentov pri predmetu Digitalna Vezja

Datoteke s seznamami rezultatov sprotnih obveznosti bodo za posameznega študenta vsebovale samo ime, priimek, vpisno številko in e-poštni naslov. Zaradi dogovora z izvajalci lahko vsebujejo tudi študente, ki niso prisotni v seznamu vpisanih študentov. Posamezne obveznosti znotraj datoteke bodo pred imenom vsebovale predpono, ki bo označevala tip obveznosti. Obveznosti se bodo delile na:

- naloge (predpona *Naloga:*),
- kvize (predpona *Kviz:*),
- seštevke (predpona *Seštevke:*)
- prisotnosti (predpona *Prisotnost:*),
- kolokvije (predpona *Kolokvij:*),
- izpite (predpona *Izpit:*).

Slika 3.2 prikazuje seznam z rezultati obveznosti študentov, ki ga pridobimo z izvozom iz spletne učilnice fakultete [33] v formatu XLSX (ang. Open

XML Spreadsheet) [23].

Ime	Priimek	Identifikacijska številka	Naslov e-pošte	Naloga:DV_Seminar 1 (Poročilo 1. del - oddaja) (Prave)	Naloga:DV_Seminar 1 (Oddaja - Poročilo 2)				
				1	5				
				1	4				
				1	4,5				
Naloga:DV_Semafor (Oddaja) (Prave)		Naloga:DV_Seminar 2 (Oddaja - Poročilo) (Prave)		Seštevek: 5-7		Kviz:Kviz 6 (Prave)		Kviz:Kviz 8 (Prave)	
-				2,5		7,5		1	
-				2,5		6,5		1	
-				2,5		7		1	
Kviz:Kviz 9 (Prave)		Seštevek: 9-11		Kviz:Kviz 1 (Prave)		Kviz:Kviz 2 (Prave)		Kviz:Kviz 3 (Prave)	
1		3		3		3		3,3	
1		3		2		3		3,3	
1		3		4		2		4	
Kviz:Kviz 7 (Prave)		Kviz:Kviz 10 (Prave)		Kviz:Kviz 11 (Prave)		Seštevek: 13-20		Seštevek predmeta (Prave)	
4,5		1		2		24,8		32,3	
2		1		1		18,3		24,8	
5		1		1		25		32	

Slika 3.2: Primer podatkov s študentskimi obveznostmi za predmet Digitalna vezja (2017/18)

Datoteke z rezultati posameznih kolokvijev in izpitov bodo poleg predhodno omenjenih podatkov vsebovale še datum in čas izvedbe kolokvija oz. izpita. Vsaka izmed naštetih datotek bo za posameznega študenta vsebovala vpisno številko študenta, ime, priimek, e-poštni naslov, letnik študija, tip vpisa, prvo poslušanje, število opravljanj predmeta ter sprotne obveznosti. Datoteka bo lahko poleg doseženih rezultatov vsebovala tudi predhodno vsebovane končne ocene, ki so jih študenti dosegli na kolokviju.

Slika 3.3 prikazuje seznam študentov z rezultati posameznega kolokvija, ki ga pridobimo z izvozom iz fakultetnega informacijskega sistema STUDIS [35] v formatu CSV (ang. comma-separated values) [5].

23. 11. 2017 (čet) kolokvij, ob 16:00, Digitalna vezja (63718) 2017/18					
Vpisna številka	Priimek, ime	Vpis	Tip vpisa	Prvo poslušanje	Število opravljanja predmeta
		3 BVS-RI LJ	Prvi vpis v letnik	2017/18	1.
		2 BVS-RI LJ	Prvi vpis v letnik	2017/18	1.
		2 BVS-RI LJ	Prvi vpis v letnik	2017/18	1.

Sprotne obveznosti	Utežena vsota kolokvijev	Točke roka	Končna ocena
		64,5	77
		57	74
		76,5	79
			8
			7
			10

Slika 3.3: Rezultati kolokvijev pri predmetu Digitalna vezja (2017/18)

Slika 3.4 prikazuje seznam študentov z rezultati posameznih izpitov, ki ga pridobimo z izvozom iz fakultetnega informacijskega sistema STUDIS [35] v formatu CSV (ang. comma-separated values) [5].

29. 01. 2018 (pon) pisni, ob 10:00, Digitalna vezja (63718) 2017/18						
Vpisna številka	Priimek, ime	Vpis	Tip vpisa	Prvo poslušanje	Število opravlja	Sprotne obvezn
		2 BVS-RI LJ	Prvi vpis v letni	2017/18	1.	
		1 BVS-RI LJ	Ponavljanje leti	2017/18	1.	
		2 BVS-RI LJ	Prvi vpis v letni	2017/18	1.	

Kolokvij 23. 11.	Kolokvij 18. 01.	Utežena vsota k	Točke roka	Končna ocena
74	40	57	77	7
85	30	57,5	74	9
68	40	54	50	7

Slika 3.4: Rezultati izpitov pri predmetu Digitalna vezja (2017/18)

Datoteke s komentarji o izvajanju izbranega predmeta bodo podane v tekstovnem (.txt) formatu, služile bodo kot dodaten opis posebnosti izvajanja izbranega predmeta v študijskem letu.

Zaradi zagotavljanja varnosti in anonimnosti podatkov bodo podatki o imenih, priimkih ter e-poštnih naslovih odstranjeni.

## 3.2 Obdelava podatkov

Vse pridobljene podatke in rezultate ocenjevanj iz predhodno opisanih datotek bomo združili v enotno tabelo, ki bo prikazovala osnovne informacije o študentih vpisanih v predmet, ocene njihovih obveznosti in ostale dosežene rezultate. Tabelo bomo uredili po vrstnem redu ocen in obveznosti. Podatke o rezultatih ocenjevanja bo treba kategorizirati glede na vrsto rezultata oz. ocene. Kategorije se bodo delile na sprotne kvize, domače naloge, prisotnosti, kolokvije, izpite in ocene ustnega izpita.

Za pridobljene rezultate sprotnega ocenjevanja bo treba izvajati tudi preverjanje pogoja za opravljanje izpita, izračunavanje ocene sprotnih obveznosti in vpis končne ocene posameznega študenta pri izbranem predmetu.

Pri preverjanju pogoja za opravljanje izpita bomo izvajali računanje z uporabo vseh predhodno vnesenih podatkov kot so ocene nalog, rezultati kolokvijev in izpitov ter prisotnosti. Število doseženih točk posameznega študenta pri izbranem predmetu bomo izračunali z uporabo enačbe, ki jo uporabnik vpiše preko vnosnega polja. Po izračunanem rezultatu enačbe bomo glede na vpisano mejo pridobili informacijo, ki bo določala ali bo študentu dovoljeno pristopiti k izpitu.

Vsi rezultati ocenjevanj posameznih študentov in enačbe bodo spremenljivi glede na želje izvajalca predmeta. Obdelava osebnih podatkov kot so ime, priimek in e-poštni naslov študenta bo zaradi ohranjanja anonimnosti študentov onemogočena. Obdelane podatke bomo hranili kot zgodovino za nadaljno obdelavo ter uporabo pri grafični predstavitvi podatkov znotraj različnih vrst dokumentov in grafov.

Urejene in popravljene podatke bomo lahko izvozili v nove datoteke enakih formatov za lažje uvažanje novih podatkov v elektronski indeks fakultete, kjer bodo lahko pedagogi z uporabo novih podatkov študentom zaključili in dodelili končne ocene.

### 3.3 Predstavitev podatkov

Podatke bomo predstavili v enotni tabeli, ki bo vsebovala vse predhodno omenjene kolone v uvoženih datotekah. Združene in po potrebi še dodatno v aplikaciji obdelane podatke bomo uporabljali za grafični prikaz študijskih uspehov glede na izbran predmet in študijsko leto v katerem se je predmet izvajal, ter zgodovino vseh obdelanih rezultatov. Podatki bodo predstavljeni v sklopu večih kategorij. Vsaka kategorija bo prikazovala posamezen tip izrisa, ter zelene podatke znotraj grafa.

Kategorija prikaza študentskih rezultatov skozi leta oz. cikle izvajanja predmeta bo vsebovala stolpčne diagrame, ki bodo po študijskih letih prikazovali uspeh študentov in njihove dosežene rezultate na kolokvijih oz. izpitih.

Podatke o študijskih uspehih študentov bomo skupaj z razmerjem med številom vpisanih študentov v predmet in številom študentov, ki so opravili izbran predmet po študijskih letih prikazovali v črtnem grafu.

Uspehe študentov za posamezno leto bomo prikazovali znotraj tortnih in stolpičnih grafov. Tortni grafi bodo ponazarjali deleže obveznosti s katerimi so študenti opravili izbran predmet, stolpični pa bodo prikazovali razmerje med študenti, ki so opravljali določeno obveznost in tistimi, ki so to obveznost uspešno opravili.

Izvajalcem bo omogočen tudi prikaz podatkov znotraj izdelanih poročil formata PDF [24], vsako poročilo bo vsebovalo podatke o predmetu in študijskem letu, izvajalčev komentar o izvedbi predmeta, podatke o številu študentov, ki so opravljali in opravili posamezne kolokvije ter izpite. Znotraj poročila bo tudi izrisan stolpični diagram, ki bo prikazoval razmerje med številom vseh vpisanih študentov in študentov, ki so predmet opravili.



# Poglavje 4

## Implementacija

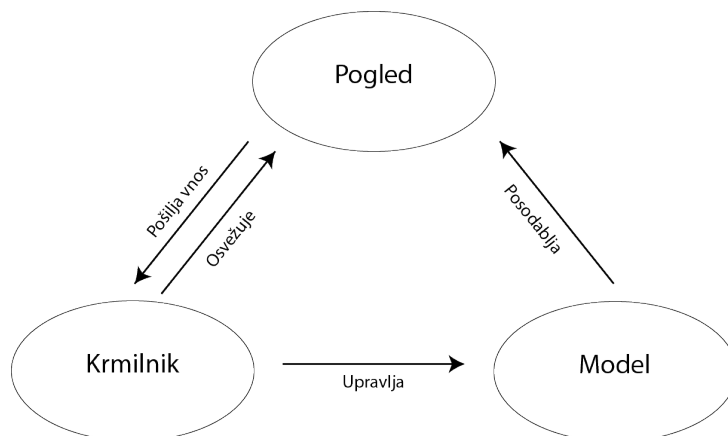
### 4.1 Načrtovanje spletne aplikacije

Spletna aplikacija za obdelavo rezultatov ocenjevanja bo ustvarjena z uporabo tehnologij HTML [37], CSS [36], JavaScript [17], PHP [27], Bootstrap [2] in MySQL [22]. Arhitektura aplikacije bo temeljila na arhitekturnem vzorcu **Model-view-controller** (Model-pogled-krmilnik) [21] ali MVC. Aplikacijo bomo po principih MVC razdelili na tri sloje, kot je prikazano na sliki 4.1, vsak sloj bo vseboval potrebne informacije ter operacije za izvajanje procesov in pravilno delovanje spletne aplikacije. Vzorec nam bo zaradi modularnosti in prilagodljivosti programske kode omogočil hitrejši razvoj spletne aplikacije.

**Model** bo skrbel za zaledno logiko aplikacije oz. za dostop in urejanje podatkov, ki so potrebni za pravilno delovanje spletne aplikacije.

**Pogled** bo s pomočjo funkcij razreda Helper prikazoval podatke, ki jih bomo predhodno obdelali znotraj modela. Za dostavljanje podatkov iz modela bo skrbel Krmilnik.

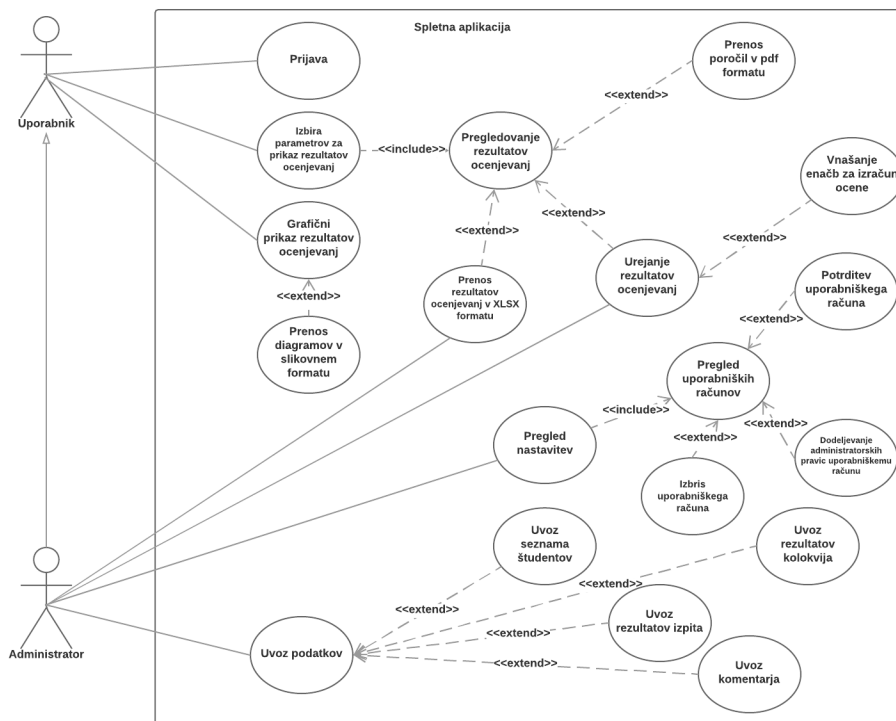
**Krmilnik** skrbi za preslikovanje vhodnih podatkov v ustrezne prikaze pogledov in podatkov znotraj pogleda.



Slika 4.1: Diagram arhitekturnega vzorca MVC.

Dostop do funkcij spletne aplikacije bo omogočen s prijavo oz. registracijo uporabniškega računa pedagoga izvajalaca, vsakemu uporabniškemu računu bodo dodeljene ustrezne pravice. Delili se bodo po pravicah na osnovnega uporabnika in administratorja. Osnovnemu uporabniku bo po administratorjevi potrditvi računa omogočeno preverjanje in pregledovanje rezultatov ocenjevanja, ne pa tudi urejanje. Uporabniškim računom, ki imajo dodeljene administratorske pravice bo omogočeno uvažanje podatkov, urejanje doseženih rezultatov študentov, izvažanje podatkov in upravljanje z ostalimi uporabniškimi računi v podatkovni bazi. Pravice uporabniških računov so prikazane v diagramu primerov uporabe na sliki 4.2.





Slika 4.2: Diagram primerov uporabe v spletni aplikaciji

Aplikacija bo sledila načelom Responsive Web Design (RWD) [32] in Mobile-first design [20], ki bodo skrbeli za prilagodljivo velikost vizualne podobe spletne aplikacije glede na napravo oz. velikost okna brskalnika s katerim dostopamo do spletne aplikacije.

Spletna aplikacija za obdelavo rezultatov ocenjevanja bo vsebovala poglede oz. strani kot so prijavna stran, stran za registracijo, stran za uvoz podatkov, stran za izbiro predmeta, ki ga želimo pregledati ali urejati, stran za prikaz rezultatov ocenjevanja izbranega predmeta, stran za grafično predstavitev podatkov in stran z nastavitvami uporabnikov. Načrt postavitve pogledov oz. strani spletne aplikacije predstavlja slika 4.3.



Slika 4.3: Načrt spletnih strani v spletni aplikaciji

Pridobivanje podatkov ocenjevanja za obdelavo bo potekalo z nalaganjem datotek, ki vsebujejo rezultate ocenjevanj, obdelavo naloženih podatkov z uporabo PHP knjižnice `PhpSpreadsheet` [28] in preslikovanjem teh podatkov v ustrezne entitete znotraj MySQL relacijske podatkovne baze. Prikaz rezultatov ocenjevanja znotraj spletne aplikacije bo potekal v interaktivni enotni tabeli, ki bo za vsakega študenta oz. vpisno številko prikazovala študentove dosežene rezultate pri izpisani obveznosti. Aplikacija bo omogočala računanje točk in omejitev za pristop k izpitom z uporabo uporabniško definiranih enačb. Po končanem urejanju bo podatke v obliki tabele možno izvoziti v formatih PDF [24], CSV [5] in XLSX [23]. Rezultate bomo lahko prikazali tudi grafično z uporabo JavaScript [17] knjižnice `Chart.js` [4]. Prikazovali bomo lahko študentske uspehe izbranega predmeta za vsako leto ali pa zgodovino uspehov za vsa študijska leta, ki se nahajajo v podatkovni bazi. Prikazane dinamične grafe bomo zmožni tudi izvoziti v slikovnem formatu `Portable Network Graphics` [30] oz. PNG z uporabo JavaScript knjižnice `FileSaver` [9].

Uporabniku z administratorskimi pravicami bo omogočeno tudi upravlja-

nje z drugimi uporabniškimi računi, ki se nahajajo v podatkovni bazi. Izpis uporabniških računov bo administratorju prikazal vse podatke posameznega uporabniškega računa razen gesla. Administrator bo lahko izvajal tri vrste operacij nad zelenim uporabniškim računom, te operacije so potrjevanje uporabniškega računa, kar bo omogočilo uspešno prijavo v sistem, nastavljanje administratorskega privilegija in izbris uporabniškega računa.

## 4.2 Podatkovna baza

Spletna aplikacija mora za prikazovanje, obdelavo, izvažanje in izris zagotavljati hranjenje uvoženih in obdelanih podatkov. Zasnova relacijske podatkovne baze je potekala na osnovi izbire prikaza zelenega predmeta s strani pedagoga, vsak pedagog bo za prikaz podatkov moral izbrati fakulteto, študijsko leto in predmet, ki želi prikazati oz. urediti. Napreden diagram entiten in razmerij (ang. Enhanced entity-relationship diagram [8]) podatkovne baze si lahko ogledamo na sliki 4.4.

Za potrebe hranjenja podatkov v spletni aplikaciji bo definirana podatkovna baza z večjim številom entitet. Prva entiteta imenovana *"users"* bo vsebovala podatke, ki se navezujejo na uporabniške račune pedagogov kot so uporabniško ime, e-poštni naslov, geslo ter pravice, ki so uporabniškemu računu oz. pedagogu dodeljene.

Rezultati ocenjevanja študentov se bodo nahajali znotraj preostalih entitet oz. tabel podatkovne baze.

Vsaka podana fakulteta bo zajeta v entiteti *"faculty"*, ki bo vsebovala edinstveno identifikacijsko številko in ime fakultete.

Vsako izbrano študijsko leto bo vsebovano znotraj entitete *"year"*, ki vsebuje edinstveno identifikacijsko številko ter letnico, ki opisuje začetek študijskega leta. Entiteta tudi vsebuje identifikacijsko številko entitete *faculty*, ki ji pripada.

Za posamezen predmet je definirana entiteta *subject*, ki bo vsebovala

edinstveno identifikacijsko številko, ime predmeta, uporabljeno enačbo in mejo za izračun točk, identifikacijsko številko študijskega leta, ki mu pripada ter nekaj podatkov o številu kolokvijev ter izpitov.

Podatki o posameznemu študentu bodo hranjeni znotraj entitete *student*, ki bo pripadala posamezni entiteti *subject*. Vsak primerek oz. entiteta študenta bo vsebovala edinstveno identifikacijsko številko študenta (uporabljena bo samo za potrebe podatkovne baze), vpisno številko študenta, študentove povprečne točke in dosežene končne ocene pri kolokvijih oz. izpiti, rezultat ustnega ocenjevanja ter nekatere preostale podatke namenjene izračunavanju končne ocene.

Vsakemu študentu bo pripadalo večje število drugih entitet, ki predstavljajo posamezne ocene nalog oz. dosežke preverjanj. Pri izpisu rezultatov ocenjevanja posameznega študenta bomo prikazali vse obstoječe instance entitete, ki se navezujejo na posameznega študenta.

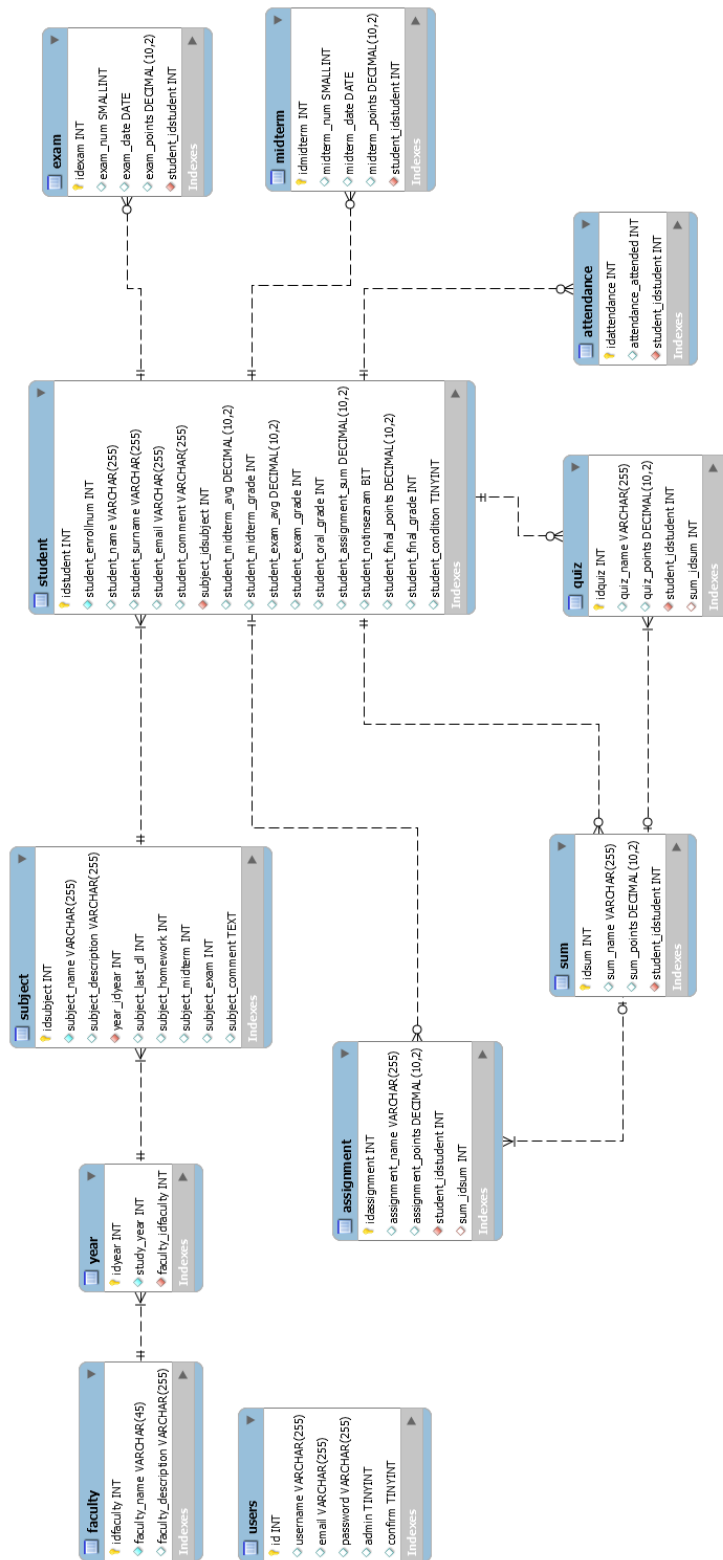
Podatki o posameznem kolokviyu oz. izpitu bodo zaradi ohranjanja preglednosti baze shranjeni znotraj posameznih entitet *midterm* in *exam*, vsaka entiteta bo vsebovala svojo edinstveno identifikacijsko številko, zaporedno število kolokvija oz. izpita, datum opravljanja, dosežene točke ter identifikacijsko številko študenta, ki mu pripada.

Podatki posameznih nalog in kvizov bodo predstavljeni znotraj entitet *assignment* in *quiz*, obe entiteti vsebujeta ime naloge ali kviza, dosežene točke, identifikacijsko številko študenta in identifikacijsko številko seštevka, znotraj katerega je posamezna entiteta zajeta.

Vsaka entiteta *assignment* ali *quiz* pripada entiteti *sum*, ki vsebuje edinstveno identifikacijsko številko, ime seštevka, točke seštevka ter podatke o študentu, ki mu pripada. V primeru, da bo izvajalec ročno spreminjal vrednosti doseženih točk pri izbrani nalogi, se bodo podatki znotraj entitete *sum* ustrezno popravili. To funkcionalnost bomo v podatkovni bazi dosegli z uporabo sprožilcev.

Kadar bo študentu pripadala prisotnost, se bodo podatki o prisotnosti shranili v entiteto *attendance*, ki vsebuje svojo edinstveno identifikacijsko

številko, polje attended, ki vsebuje podatek o prisotnosti študenta in identifikacijsko številko študenta, ki mu pripada.



Slika 4.4: Relacijski diagram podatkovne baze

## 4.3 Implementacija spletne aplikacije

### 4.3.1 Grafični vmesnik

Vizualna podoba aplikacije je realizirana z uporabo tehnologij HTML [37], CSS [36], Javascript [17], jQuery [19] in front-end ogrodja Bootstrap [2], ki omogoča preprosto in hitro izdelavo odzivnih spletnih strani [32].

Navigacija po spletni aplikaciji poteka z uporabo orodne vrstice, ki se na vrhu spletne aplikacije. Povezave v orodni vrstici se navezujejo na izbrano stran in se dinamično spreminjajo.

### 4.3.2 Arhitektura aplikacije

Aplikacija bo temeljila na arhitekturnemu vzorcu Model-pogled-krmilnik oz. MVC [21]. Izvedbo osnovne oblike MVC smo za potrebe spletne aplikacije z uporabo programskega jezika PHP [27] realizirali sami. Na začetku smo ustvarili ustrezno datotečno strukturo, ki bo hranila programsko kodo aplikacije. Po končanem ustvarjanju strukture smo izdelali *usmerjevalnik*, ki skrbi za ustrezno preslikovanje vnešenega URL-ja v funkcije *krmilnika* ter *pogled* spletne aplikacije. Po končani izdelavi *usmerjevalnika* smo ustvarili *krmilnik* in razred, ki z uporabo PDO [31] omogoča povezavo z podatkovno bazo. Vse poizvedbe SQL [34], ki jih bomo izvajali znotraj zalednega dela aplikacije bodo uporabljale PDO Prepared Statement [26], kar bo omogočilo večjo hitrost in varnost njihovega izvajanja.

### 4.3.3 Registracija in prijava

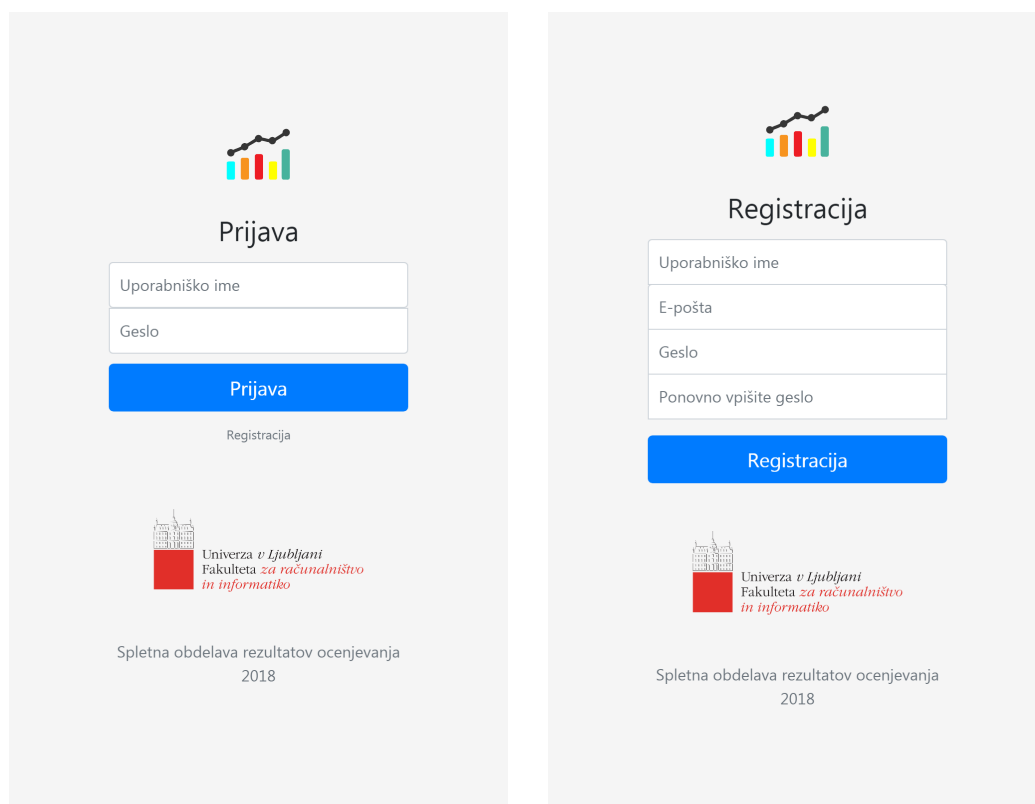
Za uporabo spletne aplikacije se je pedagogu potrebno prijaviti z uporabniškim računom. Če oseba še nima ustvarjenega uporabniškega računa, ga lahko ustvari z izbiro povezave *Registracija* na prijavni strani aplikacije in ga bo preusmerila na stran za registracijo.

Po vnosu zelenih podatkov se s pritiskom na gumb *Registracija*, podatki preko HTTP metode POST [15] prenesejo v zaledje strežnika. V zaledju

se nad podatki z uporabo programskega jezika PHP izvede validacija, če med validacijo pride do napak, uporabniku izpišemo ustrezno opozorilo, če je validacija uspešna se podatke z uporabo PDO Prepared Statement vnese v poizvedbo SQL, ki nato poskrbi za shranjevanje podatkov v podatkovno bazo.

Strani za prijavo in registracijo sta prikazani na sliki 4.5. Po stvaritvi novega uporabniškega računa je pred prijavo treba počakati na potrditev uporabniškega računa s strani administratorja.





(a) Prijavna stran

(b) Registracijska stran

Slika 4.5: Prijavna in registracijska stran spletne aplikacije

#### 4.3.4 Uvoz podatkov

Za uvoz datotek s podatki študentskega ocenjevanja morajo biti pedagogi prijavljeni v račun z administratorskimi privilegiji. Ob prihodu na stran se nam ponudi več možnosti za izbiro vrste podatkov, ki jih želimo uvoziti v podatkovno bazo spletne aplikacije. Ponujene možnosti vključujejo uvoz seznama študentov, rezultatov kolokvijev, rezultatov izpitov, rezultatov vseh obveznosti ter komentarja. Ob izbiri posamezne možnosti za uvoz se nam prikažejo izbirna in vnosna polja, ki so dinamično napolnjena z uporabo tehnologij Ajax, v katerih izberemo ali vnesemo ustrezne podatke, ki se navezujejo na vnesen seznam študentov.

Pri uvozu seznama vpisanih študentov se nam prikažejo vnosna in izbirna

polja *ime fakultete*, *število kolokvijev* predmeta in število izpitov predmeta. Po izpolnjenih vnosnih poljih lahko v polju za izbiro datoteke formata CSV izberemo seznam vpisanih študentov. Ob kliku gumba *Potrdi* se vsebina datoteke z uporabo metode HTTP POST pošlje v obdelavo zaledju aplikacije. V primeru napak uporabniku sporočimo tip napake.

Uvažanje seznama vseh obveznosti študentov in komentarja nam prikaže izbirna polja *ime fakultete*, *študijsko leto* in *izbira predmeta* s katerimi določimo kateri izvedbi predmeta pripadajo podatki. Po izbiri podatkov v polje za izbiro datoteke vnesemo ustrezno datoteko formata XLSX za obveznosti ali TXT za komentar. Ob potrditvi se vsebina datoteke z uporabo metode HTTP POST pošlje v obdelavo zaledju aplikacije. V primeru napak uporabniku izpišemo tip napake.

Možnosti za uvoz seznamov z rezultati kolokvijev in izpitov nam prikažejo izbirna polja *ime fakultete*, *vsebnost končnih ocen v datoteki* in *zaporedna števila kolokvija*. Po izpolnjenih izbirnih poljih z izbiro datoteke formata CSV vnesemo seznam študentov, ki so sodelovali na izbranem kolokviju. Ob kliku gumba *Potrdi* se vsebina datoteke z uporabo metode HTTP POST pošlje v obdelavo zaledju aplikacije. V primeru napak na strani izpišemo tip napake.

Zaledje aplikacije obdelane podatke z uporabo PDO Prepared Statement vstavi v poizvedbe SQL [34] in shrani v podatkovno bazo. Če se v datotekah pojavi študent, ki prvotno ni bil vnešen v seznam vpisanih študentov, ga vstavimo v ustrezen predmet, mu v tabeli podatkovne baze označimo da ni bil v prvotnem seznamu in izvajalcu prikažemo opozorilo. Če entiteta vnešenega predmeta ali študenta že obstaja posodobimo stare attribute entitet z novimi, ki smo jih pridobili iz naložene datoteke.

Ime vsake uspešno uvožene datoteke se skupaj s časom uvoza shranjuje v datoteko *history.log*. Ob izbiri povezave *Zgodovina* se nam prikaže pojavno okno, ki z uporabo tehnologije Ajax prikaže vsebino datoteke.

Uvoz Parametri Predstavitel Uvoz podatkov Zgodovina Nastavitve Odjava

### Uvoz podatkov

Izberite datoteko za uvoz.

# Seznam študentov @ Rezultati Kolokvijev @ Rezultati Izpitov @ Rezultati vseh obveznosti / Komentar

Izberite fakulteto

Fakulteta za računalništvo in informatiko

Področje	Število
Kolokvij	Št. Kolokvijev
Izpit	Št. Izpitov

Izberite .csv datoteko... Izbira datoteke

Potrdi Ponastavi

Inženirstvo in Informatika  
Fakulteta za računalništvo in informatiko  
© 2018

Spletna obdelava rezultatov ocenjevanja  
Diplomsko delo

**Povezave**  
Fakulteta za računalništvo in informatiko  
Spletna učilnica  
STUDIS

Slika 4.6: Stran za uvoz podatkov

### 4.3.5 Izbira parametrov za prikaz

Stran za izbiro parametrov za prikaz in urejanje, ki jo prikazuje slika 4.7, nam ponudi izbirna polja z imeni v podatkovni bazi shranjenih fakultet, po izbiri fakultete se nam z uporabo Ajaxa dinamično napolni še izbirno polje s študijskimi leti, ki se navezujejo na izbrano fakulteto, po izbiri študijskega leta se enak postopek izvede še za izbirno polje, ki vsebuje predmet, ki ga želimo prikazati. Ko smo izbrali vrednosti s pritiskom na gumb Potrdi podatke z uporabo HTTP metode POST pošljemo v zaledje spletne aplikacije, tam s pomočjo poizvedb SQL pridobimo podatke glede na vnešene parametre ter prikažemo "view" oz. stran za prikaz rezultatov ocenjevanja.

V primeru dostopa do strani z administratorskim računom nam stran ponudi še opcijo uvoza novih podatkov s pritiskom na gumb Uvoz novih podatkov, ki se nahaja pod ikono, ustvarjeno z uporabo Font Awesome paketa ikon [11]. Pri dostopu s strani navadnega uporabnika gumb za uvoz novih podatkov ostane onemogočen.

Slika 4.7: Stran za izbiro parametrov za prikaz

### 4.3.6 Prikaz rezultatov ocenjevanja

Stran za prikaz rezultatov ocenjevanja (slika 4.8), nam prikaže podatke, ki so bili predhodno izbrani na strani za izbiro parametrov, v enotni tabeli HTML. Podatki se pridobivajo in osvežujejo dinamično z uporabo tehnologije Ajax.

Tabeli lahko spremenimo vsebino prikaza po kategorijah naloge, kvizi, kollokviji in izpiti. Tabela z uporabo jQuery vtičnika DataTables [7] pridobi funkcionalnosti iskanja, prehoda po straneh, nastavljanja števila prikazanih vrstic in ohranjanja fiksne stolpca z vpisnimi številkami. Vsako polje oz. rezultat v tabeli lahko pedagog z dodeljenimi administratorskimi pravicami ureja. Končne točke studentov v tabeli lahko izračunamo z vnašanjem enačb naslednje oblike:

$$C1 + C2 - C3 * C4 / C5 + \dots + Cn$$

V vnosno polje za enačbe, se ob kliku gumba *Potrdi*, ustrezno izpolnejo polja za izračun točk. Stolpce, ki jih želimo uporabiti v enačbi pri izračunu označujemo z simbolom  $Cn$ , kjer  $n$  predstavlja število stolpca. V enačbi lahko

uporabljam o osnovne matematične operatorje kot so seštevanje, odštevanje, množenje in deljenje. Po izračunu točk z uporabo enačbe lahko v vnosnem polju za mejo nastavimo še mejo oz. pogoj za pristop k izpitu, ki predstavlja najmanjše število točk, ki ga mora študent doseči, da se mu omogoči pristop k izpitu. Ob pritisku na gumb *Potrdi* aplikacija preveri, ali so izračunane točke študenta večje ali enake vneseni meji. V primeru, da so študentove točke večje ali enake vnesenemu pogojju se študentov pogoj postavi na 1, kar pomeni da študent lahko pristopi k izpitu. Če vrednost pogoja ostane na 0, pomeni da študent ne more pristopiti k izpitu. Prav tako lahko enačbe in pogoj nastavlja samo uporabnik z dodeljenimi administratorskimi pravicami. Polja za izračun končnih točk in pogojev so v tabeli zaradi preglednosti barvno označena. Vpisne številke študentov, ki niso bili vsebovani v začetnem seznamu predmeta so označene z rdečo barvo.

Administrator lahko podatke z izbiro ustrezne povezave v orodni vrstici iz tabele tudi izvozi v datoteke formatov XLSX [23], CSV [5] ter poročilo v formatu PDF [24] (slika 4.9). Za izvoz podatkov v formatu XLSX in CSV smo uporabili PHP razred *PhpSpreadsheet* [28]. Za izdelavo PDF poročila o uspehu študentov uporabljamo PHP razred *FPDF* [12].

Uvoz Parametri Obdelava Predstavitev Izvoz Obdelava ocen Nastavitve Odjava

### Pregled rezultatov ocenjevanja

Vsi podatki
  Naloge
  Kvizi
  Kolokviji
  Izpiti

Enačba  Potrdi Meja  Potrdi

Prikaži 10 rezultatov na stran  Iskanje

Vplivna številka	Naloga:DV_Semafor (Oddaja) (Prave)	Naloga:DV_Seminar 1 (Oddaja - Poročilo 2) (Prave)	Naloga:DV_Seminar 1 (Poročilo 1. del - oddaja) (Prave)	Naloga:DV_Seminar 2 (Oddaja - Poročilo) (Prave)	Kviz:Kviz 1 (Prave)	Kviz:Kviz 10 (Prave)	Kviz:Kviz 11 (Prave)
f	0.00	5.00	1.00	2.50	3.00	1.00	1.00
f	0.00	5.00	1.00	2.50	3.00	1.00	2.00
f	0.00	5.00	1.00	2.50	2.00	2.00	2.00
f	0.00	5.00	1.00	2.50	4.00	2.00	0.00
f	0.00	5.00	1.00	2.50	3.00	0.00	2.00
f	0.00	5.00	1.00	2.50	2.00	1.00	1.00
f	0.00	5.00	1.00	2.50	4.00	2.00	2.00
f	0.00	5.00	1.00	2.50	3.00	2.00	1.00
f	0.00	5.00	1.00	2.50	4.00	1.00	2.00
f	0.00	5.00	1.00	2.50	3.00	2.00	0.00

Prikazujem 1 stran od 4 Prva stran Nazaj 1 2 3 4 Naprej Zadnja stran

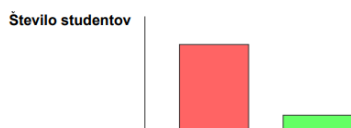
Slika 4.8: Stran za prikaz rezultatov ocenjevanja

## Rezultati ocenjevanja za Digitalna vezja (63718) - 2017/18

## Komentar izvajalca:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc suscipit orci sit amet sapien sodales lobortis. Maecenas bibendum commodo mi feugiat molestie. Curabitur tristique sem ligula, eget placerat erat vulputate eu. Nam vel posuere felis, non dignissim nulla. Donec quis nulla ipsum. Aenean bibendum massa id commodo ultrices. Integer erat arcu, blandit et dolor eu, tristique aliquam velit. Phasellus aliquet magna arcu, ac consectetur erat luctus ut.

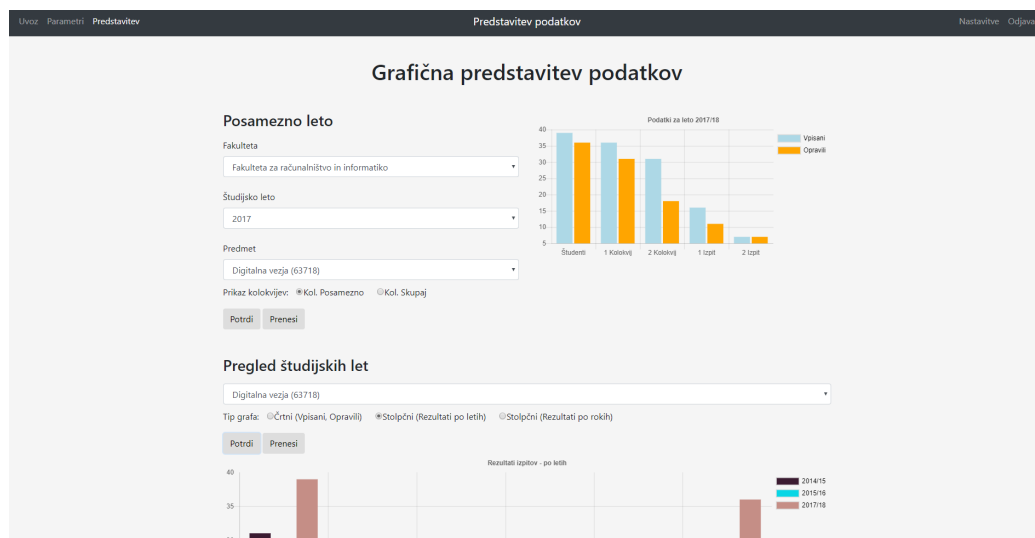
2017/18	K1	K2	I1	I2
Prijavljenih	36	31	16	7
Opravili	31	18	11	7



Slika 4.9: Primeri izvoženega poročila

### 4.3.7 Grafična predstavitev podatkov

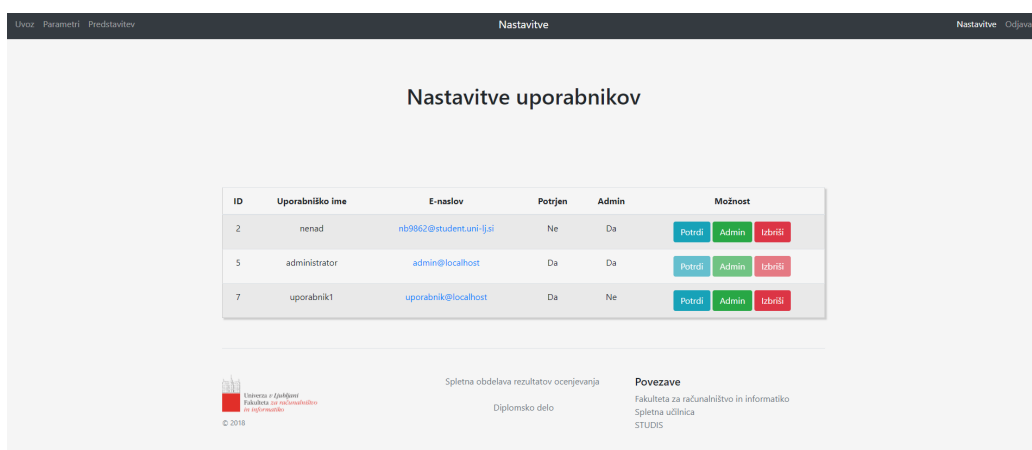
Stran za grafični prikaz rezultatov ocenjevanja, ki jo predstavlja slika 4.10, nam omogoča prikaz podatkov v grafih. Na strani se nam ponudijo dinamično napolnjena izbirna polja, v katerih izberemo podatke, kot so *ime fakultete*, *študijsko leto*, *ime predmeta* in *tip grafa*. Za zelen prikaz predmeta, ob kliku na gumb *Potrdi*, zaledje spletne aplikacije z uporabo PDO vstavi izbrane podatke v poizvedbo SQL in z uporabo tehnologije Ajax pošlje podatke v obliki notacije JSON knjižnici Chart.js [4], ki poskrbi za grafični prikaz podatkov znotraj izbrane vrste diagrama. Knjižnica Chart.js omogoča barvni prikaz elementov v grafu, ročno določitev barv le v primeru, da poznamo število podatkov, ki bodo prikazani v tabeli. Zaradi spreminjajočega števila podatkov se barve stolpcev v grafu izračunajo naključno. Spletna stran omogoča tudi shranjevanje diagramov v obliki slik formata PNG [30] z uporabo JavaScript knjižnice FileSaver [9].



Slika 4.10: Stran za grafično predstavitev podatkov (UL FRI, Digitalna vezja, število studentov / opravili izpit)

### 4.3.8 Nastavitve uporabnikov

Stran za upravljanje z uporabniki spletne aplikacije, ki jo prikazuje slika 4.11, omogoča uporabniku z administratorskimi pravicami prikaz vseh registriranih uporabnikov in njihovih podatkov. Administrator lahko pri uporabniških računih izvaja operacije potrjevanja, dodeljevanja administratorskih pravic in brisanja. Ob izbiri želene operacije se podatki prenesejo v zaledje aplikacije, kjer izvede ustrezno poizvedbo SQL, ki v podatkovni bazi spremeni vrednosti atributov izbranega uporabniškega računa.



Slika 4.11: Stran za prikaz nastavitvev uporabnikov

## 4.4 Testiranje

Testiranje delovanja aplikacije bomo ponazorili na primeru uvoza in obdelave podatkov pri predmetu Digitalna vezja, ki se je izvajal v študijskem letu 2016/17. Primer ne vsebuje uvažanja datotek s sprotnimi obveznostmi in rezultati kolokvijev. Ker se je predmet že izvajal, so končne ocene znotraj seznama z rezultati izpitov že vnesene.

Ob uspešni prijavi z administratorskim računom nas spletna stran preusmeri na stran za določanje parametrov. Tam s klikom na gumb *Uvoz novih podatkov* preidemo na stran za uvoz podatkov.



Odpre se nam nova stran, na kateri lahko izberemo ustrezne datoteke za uvoz podatkov. Prva datoteka je seznam vpisanih študentov v predmet, kjer v polji *Št. Kolokvijev* in *Št. Izpitov* vnesemo števili 0 in 3. Ob pritisku na gumb *Izbira datoteke* izberemo seznam vseh študentov ter ga s klikom na gumb *Potrdi* predamo zaledju aplikacije, ki ga vnese v podatkovno bazo.

Slika 4.12 prikazuje vnos seznama vpisanih študentov pri predmetu Digitalna vezja (2016/17).

**Uvoz podatkov**  
Izberite datoteko za uvoz.

Seznam študentov    Rezultati Kolokvijev    Rezultati Izpitov    Rezultati vseh obveznosti / Komentar

Izberite fakulteto

Fakulteta za računalništvo in informatiko

Področje	Število
Kolokvij	0
Izpit	3

2016-17 DV\_seznam.csv   Izbira datoteke

Slika 4.12: Vnos seznama vpisanih študentov

Ko je uvoz seznama končan, uvoz ponovimo z izbiro možnosti *rezultati izpitov*, v izbirnih poljih izberemo fakulteto in označimo opcijo, ki določa da datoteka vsebuje končne ocene. V vnosno polje *Številka izpita* vnesemo zaporedno številko izpita, katerega rezultate uvažamo. Celoten postopek ponovimo še za preostale izpite.

Za konec uvozimo še tekstovno datoteko z vsebino za vpis v poročilo. V izbirnih poljih izberemo ustrezen predmet, ki mu rezultati pripadajo in s potrditvijo uvozimo podatke v podatkovno bazo. Ob zaključku uvoza nas

aplikacija avtomatsko preusmeri na spletno stran za izbiro parametrov za prikaz. Ob potrditvi izbranih parametrov se prikaže glavna stran za prikaz podatkov. V tabeli ročno uredimo oceno izpita posameznega študenta, kot je prikazano na sliki 4.13.

### Pregled rezultatov ocenjevanja

Vsi podatki
  Naloge
  Kvizi
  Kolokviji
  Izpiti

Enačba Potrdi Meja Potrdi

Podatki uspešno posodobljeni!

Prikaži  rezultatov na stran Iskanje

Vpisna številka	Seštevek predmeta (nalog)	Pogoj	Izpit 1	Izpit 2	Izpit 3	Končna ocena kolokvij	Končna ocena izpit	Ustni izpit	Izračun točk	Komentar
	0.00	1	0.00			0	10	0	0.00	
	0.00	1	0.00			0	10	0	0.00	
	0.00	1		0.00		0	10	0	0.00	
	0.00	1	0.00			0	10	0	0.00	
	0.00	1		0.00		0	9	0	0.00	
	0.00	1	0.00			0	9	0	0.00	
	0.00	1		0.00		0	9	0	0.00	
	0.00	1	0.00			0	9	0	0.00	
	0.00	1		0.00		0	9	0	0.00	
	0.00	1		0.00		0	8	0	0.00	

Prikazujem 1 stran od 3 Prva stran Nazaj  2 3 Naprej Zadnja stran

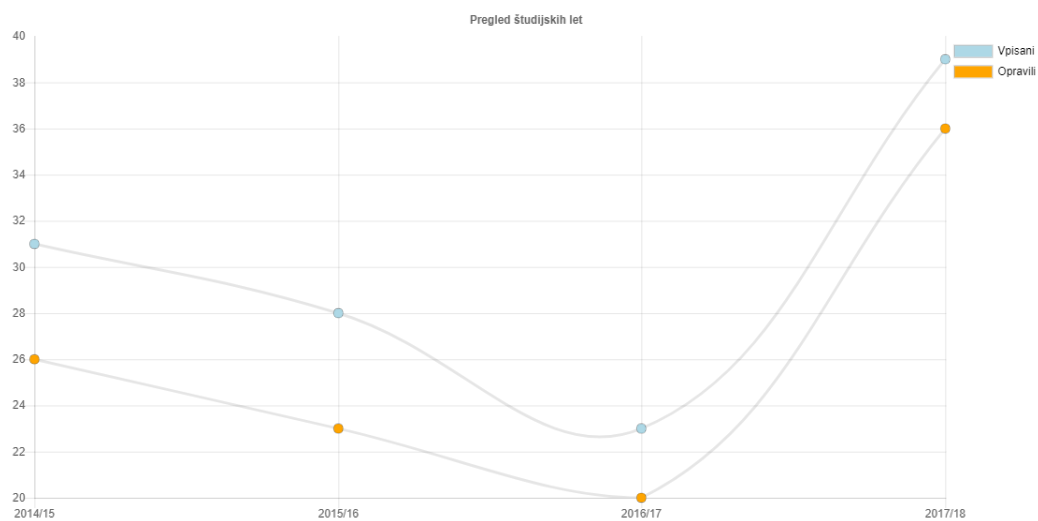
Slika 4.13: Popravljanje izpitne ocene izbranega študenta

Po končanem urejanju rezultatov ocenjevanj izvozimo podatke v formatu XLSX in generiramo novo PDF poročilo o uspehu študentov z izbiro opcije *Izvoz*. Po končani obdelavi podatkov v orodni vrstici izberemo možnost *Predstavitev*, spletna aplikacija nas ob pritisku na gumb preusmeri na stran za predstavitev podatkov. Z izbiro zelenih parametrov in klikom na gumb *Potrdi* prikažemo zeleno vrsto grafa. Pritisk gumba *Prenesi* nam prenese izrisan grafikon v slikovnem formatu PNG na računalnik (slika 4.14).



Slika 4.14: Prikaz uspešnosti opravljanja obveznosti pri predmetu Digitalna vezja (2016/17)

Uspešnost študentskega opravljanja predmeta po posameznih študijskih letih si lahko ogledamo v razdelku *Pregled študijskih let*, v izbirnem polju za izbiro imena zelenega predmeta izberemo *Digitalna vezja (63718)* ter tip grafa nastavimo na črtni, s pritiskom na gumb *Potrdi graf izrišemo*. Klik na gumb *Prenesi* nam izrisan grafikon v formatu PNG prenese na računalnik, prenešen graf je prikazan na sliki 4.15.



Slika 4.15: Prikaz uspešnosti opravljanja predmeta Digitalna vezja po študijskih letih

## Poglavje 5

### Sklepne ugotovitve

Namen diplomskega dela je bil izdelava spletne aplikacije, ki bi omogočala obdelavo rezultatov ocenjevanja. Izdelana aplikacija omogoča preprost uvoz, urejanje, grafično predstavitev podatkov in izvoz rezultatov ocenjevanja študentov. Cilji, ki so bili postavljeni ob začetku izdelave spletne aplikacije so bili doseženi, saj njena uporaba olajša evidentiranje, urejanje in prenašanje rezultatov ocenjevanj med viri podatkov, ki so pridobljeni v datotekah iz različnih informacijskih sistemov.

Razvijanje spletne aplikacije z uporabo sodobnih spletnih tehnologij nam je omogočilo izdelavo preproste, a vendar zmogljive rešitve, ki omogoča enostavno nadgrajevanje in prilagodljivost zaledne logike aplikacije. Aplikacija je zaradi razvoja po načelih arhitekture MVC bolj modularna, to pa omogoča hitrejša in enostavnejša odpravljanje napak v programski kodi.

Med izdelovanjem spletne aplikacije smo naleteli na probleme branja podatkov pri datotekah, ki so bile šifrirane v obliki UTF-16LE. Problem smo odpravili z uporabo PHP funkcije *mb\_convert\_encoding* tako, da smo pri sprotnem branju vrstic iz datoteke te vrstice pretvarjali v obliko UTF-8.

Aplikacijo bi v prihodnosti lahko nadgradili z dodatnimi funkcijami za analizo posameznih kategorij sprotnih obveznosti in preverjanje rezultatov neaktivnih ali manj aktivnih študentov v pregledni tabeli. Lahko bi razširili nabor grafov za prikaz vmesnih rezultatov. Nadgradili bi lahko tudi izdelavo

PDF poročil z večjim naborom možnosti. Administratorju bi v aplikaciji na strani z nastavitvami dodali več nadzora in možnost direktnega spreminjanja atributov entitet v podatkovni bazi. Zaradi ohranjanja varnosti aplikacije bi bilo ob dodajanju novih funkcionalnosti potrebno sproti dopolnjevati tudi varnostno zaščito aplikacije, saj bi le tako aplikaciji omogočili varnost pri hranjenju in obdelovanju zaupnih podatkov.







# Literatura

- [1] Ajax. Dosegljivo: <http://adaptivepath.org/ideas/ajax-new-approach-web-applications/>. [Dostopano 8.9.2018].
- [2] Bootstrap. Dosegljivo: <http://getbootstrap.com>. [Dostopano 8.9.2018].
- [3] Bootstrap grid system. Dosegljivo: <http://getbootstrap.com/docs/4.1/layout/grid/>. [Dostopano 8.9.2018].
- [4] Chart.js Library. Dosegljivo: <https://chartjs.org>. [Dostopano 8.9.2018].
- [5] Comma-separated values. Dosegljivo: <http://www.ietf.org/rfc/rfc4180.txt>. [Dostopano 8.9.2018].
- [6] CSS3 5. Dosegljivo: <https://www.w3schools.com/css>. [Dostopano 8.9.2018].
- [7] Data Tables. Dosegljivo: <http://datatables.net>. [Dostopano 8.9.2018].
- [8] Enhanced entity-relationship diagram. Dosegljivo: <https://dev.mysql.com/doc/workbench/en/wb-creating-eer-diagram.html>. [Dostopano 8.9.2018].
- [9] File Saver Plugin. Dosegljivo: <https://github.com/eligrey/FileSaver.js/>. [Dostopano 8.9.2018].

- 
- [10] Flexbox. Dosegljivo: <http://flexboxgrid.com/>. [Dostopano 8.9.2018].
  - [11] Font Awesome. Dosegljivo: <https://fontawesome.com>. [Dostopano 8.9.2018].
  - [12] Free PDF. Dosegljivo: <http://fpdf.org>. [Dostopano 8.9.2018].
  - [13] HTML 5. Dosegljivo: [https://www.w3schools.com/html/html5\\_intro.asp](https://www.w3schools.com/html/html5_intro.asp). [Dostopano 8.9.2018].
  - [14] HTML DOM. Dosegljivo: [https://www.w3schools.com/js/js\\_htmlDOM.asp](https://www.w3schools.com/js/js_htmlDOM.asp). [Dostopano 8.9.2018].
  - [15] HTML POST. Dosegljivo: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/POST>. [Dostopano 8.9.2018].
  - [16] HTTPD Apache. Dosegljivo: <https://httpd.apache.org/>. [Dostopano 8.9.2018].
  - [17] JavaScript. Dosegljivo: <https://javascript.com>. [Dostopano 8.9.2018].
  - [18] JetBrains PhpStorm. Dosegljivo: <https://www.jetbrains.com>. [Dostopano 8.9.2018].
  - [19] jQuery Library. Dosegljivo: <https://jquery.com>. [Dostopano 8.9.2018].
  - [20] Mobile first design. Dosegljivo: <https://www.uxpin.com/studio/blog/a-hands-on-guide-to-mobile-first-design/>. [Dostopano 8.9.2018].
  - [21] Model-view-controller. Dosegljivo: [https://developer.mozilla.org/en-US/docs/Web/Apps/Fundamentals/Modern\\_web\\_app\\_architecture/MVC\\_architecture](https://developer.mozilla.org/en-US/docs/Web/Apps/Fundamentals/Modern_web_app_architecture/MVC_architecture). [Dostopano 8.9.2018].
  - [22] MySQL. Dosegljivo: <https://www.mysql.com/>. [Dostopano 8.9.2018].

- 
- [23] Open Office XML. Dosegljivo: <https://support.office.com/en-us/article/open-xml-formats-and-file-name-extensions-5200d93c-3449-4380-8e11-31ef14555b18>. [Dostopano 8.9.2018].
- [24] PDF. Dosegljivo: <https://www.pdfa.org/>. [Dostopano 8.9.2018].
- [25] Perl. Dosegljivo: <https://www.perl.org/>. [Dostopano 8.9.2018].
- [26] PHP Prepared Statement. Dosegljivo: <http://php.net/manual/en/mysqli.quickstart.prepared-statements.php>. [Dostopano 8.9.2018].
- [27] PHP.net. Dosegljivo: <https://php.net>. [Dostopano 8.9.2018].
- [28] PhpSpreadsheet Class. Dosegljivo: <https://phpspreadsheet.readthedocs.io>. [Dostopano 8.9.2018].
- [29] PhpyMyAdmin. Dosegljivo: <https://www.phpmyadmin.net>. [Dostopano 8.9.2018].
- [30] Portable network graphics. Dosegljivo: <http://www.libpng.org/pub/png/>. [Dostopano 8.9.2018].
- [31] Prepared statements - MySQL. Dosegljivo: <http://php.net/manual/en/book.pdo.php>. [Dostopano 8.9.2018].
- [32] Responsive Web Design (RWD) and User Experience. Dosegljivo: <https://www.nngroup.com/articles/responsive-web-design-definition/>. [Dostopano 8.9.2018].
- [33] Spletna učilnica FRI. Dosegljivo: <https://ucilnica.fri.uni-lj.si>. [Dostopano 8.9.2018].
- [34] SQL. Dosegljivo: <http://www.iana.org/assignments/media-types/application/sql>. [Dostopano 8.9.2018].
- [35] STUDIS FRI. Dosegljivo: <https://studij.fri.uni-lj.si>. [Dostopano 8.9.2018].

- [36] W3 CSS. Dosegljivo: <https://www.w3.org/Style/CSS/>. [Dostopano 8.9.2018].
- [37] W3 Html. Dosegljivo: <https://www.w3.org/html/>. [Dostopano 8. 9. 2018].
- [38] XAMPP. Dosegljivo: <https://www.apachefriends.org/index.html>. [Dostopano 8.9.2018].