



**Univerzitet u Beogradu
Elektrotehnički fakultet**

Principi softverskog inženjerstva

ProLab



ProLab

Projektni zadatak

Verzija 1.1

Tim:

ProLabTim

Članovi tima:

- 1. Nemanja Lazić (tim lider) 2018/0004**
- 2. Slobodan Katanić 2018/0133**
- 3. Valerijan Matvejev 2018/0257**
- 4. Sreten Živković 2018/0008**

Istorija izmena

| Datum | Verzija | Opis | Autori |
|-------------|---------|--------------------|-----------------------------------|
| 12.03.2021. | 1.0 | Inicijalna verzija | Nemanja Lazić Slobodan Katanić |
| 26.03.2021. | 1.1 | Dopunjena verzija | Slobodan Katanić |
| | | | |
| | | | |

Sadržaj

| | |
|--|-----------|
| 1. Uvod | 5 |
| 1.1. Rezime..... | 5 |
| 1.2. Cilj i namena dokumenta | 5 |
| 2. Opis problema | 5 |
| 2.1. Motivacija | 5 |
| 2.2. Struktura problema | 5 |
| 3. Kategorije korisnika | 6 |
| 3.1. Gost..... | 6 |
| 3.2. Student | 6 |
| 3.3. Profesor..... | 6 |
| 3.4. Administrator | 6 |
| 4. Tehnologija | 6 |
| 5. Opis proizvoda | 7 |
| 5.1. Pregled arhitekture sistema | 7 |
| 5.2. Pregled karakteristika | 7 |
| 6. Funkcionalni zahtevi..... | 7 |
| 6.1. Autorizacija studenata, profesora i administratora | 7 |
| 6.1.1. Prijavljivanje korisnika..... | 7 |
| 6.1.2. Registracija korisnika | 8 |
| 6.2. Ostali posetioci | 8 |
| 6.3. Funkcionalnosti studenta | 8 |
| 6.3.1. Izbor i prijava na predmet..... | 8 |
| 6.3.2. Prijava i odjava laboratorijskih vežbi | 8 |
| 6.3.3. Zamena termina laboratorijskih vežbi | 8 |
| 6.3.4. Kreiranje timova za projekte | 9 |
| 6.3.5. Prijavljivanje u tim za projekat..... | 9 |
| 6.4. Funkcionalnosti profesora..... | 9 |
| 6.4.1. Dodavanje predmeta..... | 9 |
| 6.4.2. Dodavanje termina za laboratorijske vežbe | 9 |
| 6.4.3. Definisanje/uklanjanje projekta | 9 |
| 6.4.4. Prihvatanje zahteva..... | 9 |
| 6.5. Funkcionalnosti administratora | 10 |
| 6.5.1. Prihvatanje zahteva..... | 10 |
| 6.5.2. Brisanje entiteta | 10 |

| | |
|--|-----------|
| 7. Pretpostavke i ograničenja | 10 |
| 8. Kvalitet | 10 |
| 9. Nefunkcionalni zahtevi | 10 |
| 9.1. Sistemski zahtevi | 10 |
| 9.2. Ostali zahtevi | 10 |
| 10. Zahtevi za korisničkom dokumentacijom | 10 |
| 10.1. Uputstva za korišćenje sajta..... | 10 |
| 10.2. Označavanje | 11 |
| 11. Plan i prioriteti | 11 |

1. Uvod

1.1. Rezime

Projekat ProLab je deo praktične nastave na predmetu Principi softverskog inženjerstva. Aplikacija je namenjena svim predavačima i studentima Elektrotehničkog fakulteta koji su angažovani odnosno prate odgovarajući predmet u okviru koga se deo ispitnih poena stiče izradom laboratorijskih vežbi ili izradom timskog projekta. Aplikacija treba da pomogne studentima u pronalaženja tima za izradu projekta i odgovarajućih termina za laboratorijske vežbe, a predavačima da olakša organizaciju laboratorijskih vežbi i projekata.

1.2. Cilj i namena dokumenta

Ovaj dokument predstavlja projektni zadatak za projekat pod nazivom ProLab. Namenjen je svim članovima tima kao i klijentu za lakšu interakciju. Dokument definiše namenu projektnog zadatka, probleme koje aplikacija rešava, zahteve koji će biti realizovani u vidu funkcionalnosti sistema i zahteve kojima se može unaprediti sistem.

2. Opis problema

2.1. Motivacija

Na Elektrotehničkom fakultetu dosta značaja se pridaje praktičnoj nastavi te određeni deo aktivnosti na mnogim predmetima čini izrada laboratorijskih vežbi i projekata. Fakultet ima veliki broj studenata, a broj raspoloživih sala i laboratorija ne ide tome u prilog i zbog toga predmetni predavači nemaju mnogo fleksibilnosti pri organizaciji laboratorijskih vežbi i odbrani projekata i potrebno je sve to uskladiti na odgovarajući način sa obavezama studenata na drugim predmetima. Studenti nekada mogu biti sprečeni da dodju u nekom predviđenom terminu i neki drugi termin bi im više odgovarao, dok pri izradi timskih projekata može im biti teško da pronađu odgovarajuće kolege sa kojima bi sarađivali. Upravo ovakve probleme aplikacija nastoji da reši.

2.2. Struktura problema

Predavač može da kreira predmete na kojima je angažovan zadajući šifru predmeta i pritom bira i ostale predavače angažovane na datom predmetu. Nakon formiranja predmeta predavač može da kreira laboratorijsku vežbu zadajući datum, mesto i pojedinačne termine izrade same vežbe. Takođe može da definiše projekat iz odgovarajućeg predmeta koji ima svoj naziv, maksimalan i minimalan broj članova projektnog tima i krajnji rok za formiranje timova. Predavač odobrava prijavu studenata na predmet.

Nakon registracije student može da se prijavljuje na predmete pojedinačno. Može se prijaviti za jedan od slobodnih termina trenutno aktivne laboratorijske vežbe izabranog predmeta, a može i da izvrši zamenu termina sa studentom i da se pridruži odgovarajućem timu i kreira svoj tim.

Administrator sistema vodi računa o autorizaciji korisnika i prihvata/odbija zahteve za odgovarajućim akcijama od ostalih korisnika.

3. Kategorije korisnika

3.1. Gost

Gost ima mogućnost prijavljivanja na aplikaciju. Nema pravo pristupa nikakvim informacijama u okviru aplikacije.

3.2. Student

Student je korisnik aplikacije koji ima mogućnost da se prijavljuje na predmete. Jednom kada je prijavljen na predmet, student može da vidi sve aktivne laboratorijske vežbe i projekte na predmetu. Takođe ima mogućnost da se prijavi/odjavi sa termina laboratorijskih vežbi. Ako studentu ne odgovara određeni termin, a željeni termin je zauzet, postoji mogućnost da zatraži zamenu termina.

3.3. Profesor

Ima mogućnost da odobrava prijave studenata na predmetima koje predaje. Takođe ima mogućnost da kreira predmete laboratorijske vežbe i postavlja termine za njihovo izvođenje kao i da kreira projekte.

3.4. Administrator

Administrator mreže ima mogućnost manipulacije predmetima i aktivnostima vezanim za predmete. Može i da briše profesorske i studentske naloge kao i da prihvata razne vrste zahteva.

4. Tehnologija

Aplikacija će biti izrađena pomoću veb tehnologija. Na strani klijenta se koriste HTML5, CSS3 i Javascript. Od biblioteka na klijentskoj strani se koriste MaterializeCSS koja omogućava pravljenje korisničkog interfejsa po pravilima materijal-dizajna. Takođe se koristi i jezik Sass koji se prevodi u CSS. ReactJS, VueJS, Angular ili Redux se koriste da bi se olakšalo kreiranje interaktivnih korisničkih interfejsa. Na serverskoj strani se koriste PHP i MySQL kao baza podataka. Bazom podataka se upravlja pomoću veb aplikacije phpMyAdmin. Koriste se i PHP biblioteke Laravel, CodeIgniter ili Symfony. Za šablone HTML stranica se koristi Moustache.

Za komunikaciju se koristi Microsoft Teams, WhatsApp itd. Za verzionisanje koda se koriste alati Git i GitHub. Za automatizaciju se koristi GulpJS. Za testiranje klijentske strane se koriste Chrome i Firefox sa njihovim alatima za programere. Za testiranje REST servisa se koristi

Postman. Za izradu logotipa se koristi program Inkscape. Za projektovanje softvera se koristi UML jezik i alat StarUML.

5. Opis proizvoda

5.1. Pregled arhitekture sistema

Sistem radi kao dinamički generisan veb sajt. Sistem ima 4 nivoa pristupa: administrator, predavač, student i gost. Na server strani se koristi PHP jezik koji obrađuje upite klijenata i MySQL baza podataka gde se čuvaju svi podaci aplikacije. Klijenti šalju upite ka serveru, i dobijaju odgovore u vidu generisanih HTML i JSON fajlova.

5.2. Pregled karakteristika

„ProLab“ je veb aplikacija koja olakšava organizaciju laboratorijskih vežbi i projekata na fakultetima. Aplikacija se pravi pre svega za Elektrotehnički fakultet, ali može da se koristi i u svim drugim visokoškolskim ustanovama koje imaju sličan način ocenjivanja. Postoji podrška za lokalizaciju aplikacije, tako da može da se koristi i van srpskog govornog područja.

Prednosti ove aplikacije su te što se u svakom trenutku omogućava pristup sistemu. Usluge su dostupne online tako da studenti i profesori ne moraju biti fizički prisutni na fakultetu, pri kreiranju i popunjavanju termina za laboratorijske vežbe ili neke druge aktivnosti vezane za projekte. Informacije se bezbedno skladište u bazi podataka i pristup podacima je ograničen tako da niko van školske ustanove nema pristup nijednom podatku.

6. Funkcionalni zahtevi

Ovde se definišu funkcionalnosti koje sistem treba da pruži korisnicima, za svaku od kategorija ponaosob. Tamo gde način prenosa podataka i evidentiranje akcija nije specijalno naznačen, podrazumeva se da se unos vrši preko odgovarajuće HTML stranice, slanjem podataka ka odgovarajućem PHP skriptu, koji zatim vrši potrebne promene u bazi podataka.

6.1. Autorizacija studenata, profesora i administratora

6.1.1. Prijavljivanje korisnika

Prijavljivanje na sistem se vrši unosom korisničkog imena i šifre, koji se moraju poklopiti sa odgovarajućim podacima u bazi podataka, kako bi se autorizacija i autentifikacija uspešno obavila. Za sve korisnike, prijavljivanje se dešava na isti način. Nakon uspešnog prijavljivanja na sistem, korisnik može da interaguje sa sistemom na način koji je dozvoljen skupom funkcionalnosti koji mu je dodeljen.

6.1.2. Registracija korisnika

Novi korisnici koji nemaju kreiran nalog mogu se registrovati unosom imena, prezimena, korisničkog imena, lozinke, email-a i izborom tipa korisnika kojem oni pripadaju. Tip unete email adrese se mora poklopiti sa tipom korisnika. Nakon uspešne registracije korisnik može da interaguje sa sistemom.

6.2. Ostali posetioci

Ostali posetioci ne moraju imati nikakvu autorizaciju. Njima se omogućava odabir opcije za prijavu kao Student, Profesor ili Administrator.

6.3. Funkcionalnosti studenta

U ovom delu, definišu se funkcionalnosti koje su predviđene za Studente.

6.3.1. Izbor i prijava na predmet

Studenti na početku mogu da izaberu jedan od predmeta na koji su već prijavljeni ili da se prijave na neki od ponuđenih predmeta koji ne pohađaju. Nakon uspešne prijave na predmet i odobravanja prijave od profesora, studentu se u spisak predmeta koje pohađa dodaje novoizabrani predmeta. Izborom predmeta koji već pohađa, studentu se omogućavaju funkcionalnosti navedene ispod.

6.3.2. Prijava i odjava laboratorijskih vežbi

Student je u mogućnosti da izabere opciju za prijavu/odjavu laboratorijske vežbe, nakon odabira predmeta, i tom prilikom se prikazuju (listaju) sve laboratorijske vežbe vezane za taj predmet. Student može da se odabirom termina prijavi ili odjavi sa njega, sve do isteka roka za prijavljivanje, definisanog od Profesora.

6.3.3. Zamena termina laboratorijskih vežbi

Studenti su u mogućnosti da zamenjuju termine za laboratorijske vežbe međusobno na sledeći način. Preduslov za zamenu termina jeste da je student već prijavljen na neki termin, ali da mu taj termin iz nekog razloga ne odgovara. Student odabirom opcije za zamenu termina, dobija uvid u listu svih slobodnih studenata (studenata koji su, žargonski rečeno “free agents”), kojima takođe ne odgovara sopstveni termin, a kojima odgovara termin ovog Studenta. Zatim, Student je u mogućnosti da odabere termin iz liste ponuđenih koji mu odgovara (ako mu neki uopšte odgovara). U slučaju da Studentu odgovara nečiji termin, sistem vrši zamenu termina dvojici posmatranih Studenata. U suprotnom, Student može napraviti sopstveni zahtev za zamenu termina, pri čemu je neophodno da specificira jedan ili više željenih termina, nakon čega će biti ubačen u sistem za zamenu, i biti obavešten ako mu se želja za zamenom i ostvari.

6.3.4. Kreiranje timova za projekte

Studentu je dostupna opcija za kreiranje novog tima sa željenim nazivom. Novi tim, čiji je lider upravo osnivač tima, se dodaje u postojeću listu timova, koja je dostupna i vidljiva svim studentima odgovarajućeg predmeta. U slučaju da je Student osnovao tim, omogućava mu se i akcija zaključavanja tima, u slučaju da je broj članova premašio minimalan, a nije dostigao maksimalan dozvoljeni broj studenata po timu. Nakon zaključavanja, timu više nije moguće dodavati nove članove.

6.3.5. Prijavljivanje u tim za projekat

Student se može priključiti nekom otvorenom timu. Otvoreni timovi su timovi čiji maksimalan broj članova ne premašuje maksimalnu dozvoljenu veličinu, definisanu od profesora, i koji nije zaključen od samog lidera tima. Otvoreni timovi se mogu pregledati, listati i moguće je priključiti im se.

6.4. Funkcionalnosti profesora

U sledećim pasusima su opisane funkcionalnosti koje su vezane za Profesore.

6.4.1. Dodavanje predmeta

Profesor može dodati novi predmet sa zadatim nazivom, slanjem zahteva Administratoru.

6.4.2. Dodavanje termina za laboratorijske vežbe

Profesor može dodati novi termin za laboratorijske vežbe na predmetu sa odgovarajućim datumom, vremenom trajanja, mestom (lokacijom, laboratorijom ili dr.), opisom (koji je lab po redu, tematika laba), kao i rokom za prijavu. Nakon dodavanja, novi termin je vidljiv svim Studentima i dostupan je za pristupanje, kao i za zamenu.

6.4.3. Definisanje/uklanjanje projekta

Profesor ujedno može dodati projekat (ili više njih) za odgovarajući predmet. Tom prilikom, definišu se naziv projekta, maksimalan i minimalan broj članova po timu, kao i rok za formiranje timova. Analogno, Profesor može ukloniti već postojeći projekat, ako je on završen ili treba da bude ukinut.

6.4.4. Prihvatanje zahteva

Profesor može da prihvata/odbija zahteve za prijavu studenata na predmete na kojima je dati profesor angažovan.

6.5. Funkcionalnosti administratora

U narednim redovima su navedene osnovne funkcionalnosti Administratora.

6.5.1. Prihvatanje zahteva

Administrator je u mogućnosti da prihvata/odbija zahteve za kreiranje predmeta i zahteve za registrovanje novih korisnika.

6.5.2. Brisanje entiteta

Administrator može da briše iz sistema razne vrste entiteta kao što su korisnički nalozi, predmeti, laboratorijske vežbe i njihovi termini, projekti i projektni timovi.

7. Pretpostavke i ograničenja

Potrebno je čuvati podatke o autorizaciji i ne treba dozvoliti pristup delovima sistema korisnicima kojima te funkcionalnosti nisu namenjene. Obezbediti da isti student ne može da kreira više od jednog naloga.

8. Kvalitet

Neophodno je izvršiti testiranje svih funkcionalnosti i otpornosti na greške. Posebnu pažnju treba obratiti na testiranje preopterećenosti sistema.

9. Nefunkcionalni zahtevi

9.1. Sistemski zahtevi

Zbog korišćenja PHP-a i MySQL baze podataka, serverski deo sistema treba da pruži podršku za korišćenje istih. Što se tiče frontend dela, treba obezbediti kompatibilnost sa često korišćenim veb-pregledačima (Google Chrome, Mozilla Firefox, Microsoft Edge, ...).

9.2. Ostali zahtevi

Obezbediti dinamičan odziv i vizuelnu dinamičnost.

10. Zahtevi za korisničkom dokumentacijom

10.1. Uputstva za korišćenje sajta

Uputstvo za korišćenje sajta je potrebno svim tipovima korisnika.

10.2. Označavanje

Svaki tab bi trebao da ima odgovarajuće ime kojim je opisan njegov sadržaj i funkcionalnost. Stranice za prikaz profila korisnika moraju da sadrže vrstu korisnika, ime korisnika, kontakt, pregled predmeta koje pohađa i broj indeksa, ako je taj korisnik student.

11. Plan i prioriteti

Potrebno je obezbediti:

- Autorizaciju korisnika
- Registraciju i prijavu korisnika
- Dodavanje predmeta
- Dodavanje i brisanje studenata sa predmeta
- Definisane projekta
- Dodavanje laboratorijskih vežbi
- Zamenu termina laboratorijske vežbe
- Kreiranje tima za projekat
- Nalaženje slobodnog tima

U narednim verzijama projekta, ukoliko je to moguće, obezbediti opciju ocenjivanje studenata nakon odrađenih aktivnosti i postepeno uvođenje dela za direktnu razmenu poruka među korisnicima i dodatno uredjivanje profila korisnika.