

Izrada internetske aplikacije za praćenje obuke vozača

Svalina, Iva

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:168:666878>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-14**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1406

**IZRADA INTERNETSKE APLIKACIJE ZA PRAĆENJE OBUKE
VOZAČA**

Iva Svalina

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1406

**IZRADA INTERNETSKE APLIKACIJE ZA PRAĆENJE OBUKE
VOZAČA**

Iva Svalina

Zagreb, lipanj 2024.

ZAVRŠNI ZADATAK br. 1406

Pristupnica: **Iva Svalina (0036540196)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentor: prof. dr. sc. Damir Pintar

Zadatak: **Izrada internetske aplikacije za praćenje obuke vozača**

Opis zadatka:

U današnjem digitalnom dobu, programska podrška ima ključnu ulogu u svim sferama obrazovanja, uključujući i obuku vozača. U suvremenom pristupu obuci vozača, programska podrška igra ključnu ulogu olakšavajući praćenje i organizaciju obuke kako za instruktore tako i za polaznike. Ovakva programska podrška mogla bi omogućiti centralizirano upravljanje svim važnim terminima, obavezama i rezultatima vožnje na jednom mjestu, istovremeno pružajući mogućnost individualnog pristupa i praćenja napretka. Vaš zadatak jest uz pomoć modernih programskih tehnologija implementirati internetsku aplikaciju za praćenje obuke vozača. Konačno rješenje uključuje gotovu aplikaciju koja implementira sve zadane slučajeve uporabe, te na pregledan i učinkovit način pruža obećane funkcionalnosti korisniku.

Rok za predaju rada: 14. lipnja 2024.

Sadržaj

Uvod	1
1 Korisnički zahtjevi	2
1.1 Opis stranica	2
1.2 Vrste korisnika	3
1.3 Obrasci uporabe	5
2 Baza podataka	14
2.1 ER model	14
2.2 Opis relacija	15
2.3 Relacijski model	18
3 Implementacija	19
3.1 Korištene tehnologije	19
3.1.1 Spring Boot	19
3.1.2 React	19
3.1.3 Cloudinary	20
3.1.4 EmailJS	20
3.2 Dijagram razmještaja sustava	20
3.2.1 Klijent	21
3.2.2 Poslužitelj	21
3.3 Sigurnost	22
4 Sučelja web aplikacije	23
Zaključak	32
Literatura	33
Sažetak	34
Summary	35

Uvod

Polaganje vozačkog ispita dio je života gotovo svakog pojedinca. Prema nedavnim istraživanjima o zemljama u kojima je najteže i najskuplje naučiti voziti, Hrvatska se našla na vrhu rang-ljestvice. Procjenjuje se da je iznos koji vozači u Hrvatskoj moraju potrošiti kako bi položili sve ispite i dobili vozačku dozvolu oko 1085 eura. Kao razlog teškog polaganja vozačkog ispita se navode strogi uvjeti koji zahtijevaju veliku količinu učenja i praćenja napretka kandidata. Rješenje problema prijenosa znanja, organizacije izvođenja obuke i kontinuiranog praćenje svakog kandidata zahtjeva određenu razinu digitalizacije i implementaciju odgovarajuće aplikacije.

Cilj ovog završnog rada je kreiranje aplikacije "Vožnja +" čija je osnovna zadaća unapređenje vozačke obuke olakšavajući instruktorima i polaznicima autoškola organizaciju i praćenje usvajanja vozačkih vještina. Kako bi korisnici dobili osnovne informacije i kontakt s autoškolom, anonimnim korisnicima omogućen je pristup stranicama s osnovnim informacijama i pristup formi za slanje upita. Svaki polaznik na raspolaganju ima vlastiti kalendar, korisnički profil i stranicu s napretkom koja sadrži bilješke za svaki sat koji je održan. Instruktori imaju pregled svih kandidata kojima mogu zakazivati termine ovisno o njihovoj raspoloživosti, a nakon održanog termina instruktor unosi bilješke za održani sat. Kalendari kandidata i instruktora automatski se sinkroniziraju te se tako može učinkovito organizirati i pratiti radno vrijeme instruktora. Aplikacija administratoru pruža mogućnost registracije novih korisnika te pregled svih korisnika i njihovih kalendara u sustavu.

U nastavku ovog rada slijedi detaljan opis korisničkih zahtjeva putem kojih će se definirati glavni zahtjevi i funkcionalnosti aplikacije "Vožnja +". Poglavlje o dizajnu i arhitekturi sustava približit će čitatelju odabranu bazu podataka i implementaciju podatkovnog sloja, sloja poslovne logike i sloja korisničkog sučelja. Nakon implementacijskih detalja, bit će dan osvrt na korištene tehnologije. Za izradu web aplikacije korišteni su razvojni okviri Spring Boot i React, dok je za bazu korištena H2 baza podataka. Na kraju dolazi pregled korisničkih sučelja za sve vrste korisnika aplikacije.

1. Korisnički zahtjevi

1.1 Opis stranica

Početna stranica

Pri pokretanju web aplikacije prikazuje se početna stranica koja pruža korisniku osnovne informacije o web aplikaciji te usmjerava korisnika na prijavu u sustav. U gornjem dijelu stranice nalazi se navigacijska traka putem koje korisnik pristupa različitim stranicama ovisno o svojim ovlastima.

Prijava

Omogućuje korisniku unos korisničkih podataka i prijavu u aplikaciju pomoću adrese elektroničke pošte i lozinke. Korisnik se u sustav može samo prijaviti, registraciju novih korisnika obavlja administrator. Prijavljen korisnik neće moći pristupiti ovoj stranici.

Info

Pružila korisniku detaljnije informacije o autoškoli te približava svrhu same aplikacije. Stranica usmjerava korisnika na slanje upita putem kontaktnog obrasca ako ima dodatnih pitanja ili se želi prijaviti u autoškolu. Kontaktnim podacima može se pristupiti i u podnožju stranice.

Kontakt

Sadrži obrazac koji korisnik ispunjava pomoću imena, adrese elektroničke pošte i poruke. Ispunjen obrazac se šalje autoškoli klikom na gumb "Pošalji" koja se zatim javlja s odgovorima na potencijalna pitanja.

Profil

Prikazuje korisničke podatke unesene pri registraciji korisnika. Potrebni korisnički podaci za registraciju su ime, prezime, uloga, adresa elektroničke pošte, broj mobitela i datum rođenja. Osim obaveznih podataka, može se unijeti i napomena/bilješka za pojedinog korisnika koja se kasnije može uređivati i brisati.

Kalendar

Stranica namijenjena primarno instruktorima i polaznicima autoškole. Omogućuje prikaz, dodavanje, uređivanje i brisanje događaja u kalendar svakog korisnika. Na stranici se također nalaze korisničke upute za previđeno korištenje kalendara. Od kandidata se očekuje unos vremenske raspoloživosti prema kojoj instruktor unosi vrijeme obuke u raspored.

Napredak

Prikazuje napredak kandidata uz detaljan opis ishoda svakog sata. Realizirani ishod zabilježen je u posebnoj bilješci za svaki sat koju unosi instruktor putem forme na unos održanih sati. Klikom na pojedinu stavku iz trake za napredak prikazuje se skočni prozor sa svim detaljima za pojedini sat. Detalji koje korisnik može vidjeti su naslov održanog sata, datum, status, bilješka fotografija odvožene rute.

Kandidati

Stranica "Kandidati" pruža jednostavan pregled svih kandidata u autoškoli i omogućuje pristup podacima željenog kandidata. Podaci kojima instruktor i administrator mogu pristupiti su korisnički podaci sa stranice "Profil", kalendar i napredak željenog kandidata.

Instruktori

Omogućava jednostavan pregled svih instruktora u autoškoli i omogućuje pristup podacima željenog instruktora. Podaci kojima administrator može pristupiti su korisnički podaci sa stranice "Profil" i kalendar željenog instruktora.

1.2 Vrste korisnika

Postoje četiri kategorije korisnika sustava: neregistrirani korisnik te registrirani korisnici u koje ubrajamo polaznike autoškole, instruktore i administratora.

Anoniman korisnik

Web aplikacija "Vožnja +" neregistriranom korisniku pruža pristup sljedećim stranicama: početna stranica, stranica za prijavu i informacije te stranici za slanje kontaktnog obrasca. Korisnik može pronaći dodatne kontakt informacije i u podnožju stranice svake. Navedenim stranicama korisnik pristupa putem navigacijske trake.

Kandidat

Kao i anonimni korisnik, kandidat ima pristup početnoj stranici i stranici s dodatnim informacijama. Ako polaznik želi stupiti u kontakt s autoškolom, može to učiniti slanjem kontaktnog obrasca kojemu pristupa preko stranice "Info" ili putem kontaktnih podataka u podnožju stranice. Na stranici "Profil" korisnik ima pregled svojih korisničkih podataka. Na toj stranici posebno se ističe mogućnost dodavanja i uređivanja bilješke ili napomene za svakog kandidata. Jedna od glavnih funkcionalnosti koje aplikacija pruža kandidatu je na stranici "Kalendar" gdje korisnik može dodavati, uređivati i brisati događaje što je posebno prikladno za označavanje vremenske raspoloživosti samog kandidata, a u istom kalendaru Korisnik može vidjeti termine koje mu je zakazao instruktor. Na stranici "Napredak" kandidat ima pristup traci za napredak putem koje se prati obuka kandidata. Klikom na odgovarajući element navigacijske trake, kandidatu je omogućen pregled bilješki za pojedini sat.

Instruktor

Poput kandidata, instruktor ima pristup početnoj stranici, kontaktnoj formi putem stranice "Info", stranici "Profil" s korisničkim podacima i stranici s vlastitim kalendarom. Dodatna mogućnost koju aplikacija pruža instruktoru je pregled svih kandidata, njihovim stranicama za napredak i kalendarima. Na stranici "Kandidati" instruktor odabire polaznika čijim podacima želi pristupiti. Najprije mu se prikazuje stranica s napretkom koju može uređivati tako da dodaje i uređuje bilješke za pojedini sat i samog kandidata, a klikom na gumb "Pogledaj kalendar" korisnika se preusmjerava na kalendar kandidata kojemu on potom može upisivati, uređivati i brisati termine. Kalendar instruktora i kandidata su sinkronizirani što znači da će događaj koji instruktor upiše u kalendar polaznika biti vidljiv i u samom kalendaru instruktora.

Administrator

Administrator ima najveće ovlasti. Njegov primarni zadatak su registracija novih korisnika putem stranice "Registriraj" kojoj može pristupiti putem svoje navigacijske trake. Na stranicama "Instruktori" i "Kandidati" administratoru je omogućen jasan pregled svih korisnika sustava čijim podacima može pristupiti klikom na pojedinog korisnika. Administratoru je omogućen pregled korisničkih podataka sa stranica "Profil", događajima korisnika sa stranice "Kalendar" i napretku svakog kandidata putem stranice "Napredak". Osim privilegiranih funkcionalnosti, administrator ima pristup početnoj stranici, stranici i s dodatnim informacijama.

Svi registrirani korisnici se iz sustava odjavljuju odabirom opcije "Odjava" u navigacijskoj traci.

1.3 Obrasci uporabe

Ključni scenariji interakcije između korisnika i aplikacije prikazat će se uz pomoć obraza uporabe. Svaki UC (engl. Use Case) pružit će jasno strukturirane korake koje korisnik slijedi te omogućit će nam razumijevanje odziva sustava na te korake.

UC1 - Pregled Info stranice

- **Glavni sudionik:** Neregistrirani/neprijavljeni korisnik i prijavljeni korisnik
- **Cilj:** Pregled stranice s općim informacijama o web aplikaciji
- **Sudionici:** -
- **Preduvjet:** -
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju prikaza stranice Info
 2. Otvara se stranica i prikazuju informacije o web aplikaciji

UC2 - Registracija

- **Glavni sudionik:** Administrator
- **Cilj:** Stvaranje korisničkog računa za korištenje platforme
- **Sudionici:** Baza podataka, Cloudinary
- **Preduvjet:** Korisnik je prijavljen i ima ovlasti Administratora
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju registracije u navigacijskoj traci
 2. Korisnik unosi tražene podatke
 3. Baza podataka se osvježava
 4. Korisnik dobiva obavijest o uspješnoj registraciji te se osvježava forma za unos podataka
- **Opis mogućih odstupanja:**
 - 2.a Unesena zauzeta e-mail adresa
 1. Obavijestiti korisnika o neispravnom unosu i omogućiti ponovni unos neprihvaćenih vrijednosti
 2. Korisnik unosi nove vrijednosti i uspješno završava registraciju ili odustaje od registracije

UC3 - Prijava

- **Glavni sudionik:** Neregistrirani/neprijavljeni korisnik
- **Cilj:** Dobiti pristup odgovarajućem korisničkom sučelju ovisno o ulozi
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija od strane administratora, kontaktiranje autoškole
- **Opis osnovnog tijeka:**

1. Korisnik odabire opciju prijave u sustav
2. Korisnik unosi tražene podatke (adresa elektroničke pošte i lozinka)
3. Provjera ispravnosti unesenih podataka
4. Korisnik dobiva obavijest o uspješnoj prijavi i preusmjerava se na početnu stranicu za prijavljene korisnike

- **Opis mogućih odstupanja:**

- 2.a Unesena neispravna adresa elektroničke pošte i/ili lozinka
 1. Obavijestiti korisnika o neuspješnoj registraciji i omogućiti ponovni unos korisničkog imena i/ili lozinke
 2. Korisnik unosi nove vrijednosti i uspješno se prijavljuje ili odustaje od prijave

UC4 - Odjava

- **Glavni sudionik:** Prijavljeni korisnik (Administrator, Instruktor, Kandidat)
- **Cilj:** Odjava iz sustava
- **Sudionici:** -
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju odjave
 2. Korisnik gubi pristup korisničkim funkcijama
 3. Korisnik se preusmjerava na početnu stranicu za neregistrirane/neprijavljene korisnike

UC5 - Pregled osobnih podataka

- **Glavni sudionik:** Prijavljeni korisnik (Administrator, Instruktor, Kandidat)
- **Cilj:** Pregledati osobne podatke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za pregled svog profila
 2. Prikazuju se osobni podaci vezani uz korisnički račun

UC6 - Uređivanje osobne bilješke

- **Glavni sudionik:** Prijavljeni korisnik (Administrator, Instruktor, Kandidat)
- **Cilj:** Zmijeniti bilješku na osobnom profilu
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i otvoren mu je pregled osobnih podataka (**UC5**)
- **Opis osnovnog tijeka:**

1. Korisnik odabire polje za unos bilješke
2. Korisnik unosi/uređuje željeni tekst
3. Korisnik sprema promjene
4. Baza podataka se osvježava

UC7 - Slanje upita

- **Glavni sudionik:** Neprijavljeni korisnik i prijavljeni korisnik (Administrator, Instruktor, Kandidat)
- **Cilj:** Poslati upit putem kontaktne forme
- **Sudionici:** EmailJS servis
- **Preduvjet:** Korisnik je u navigacijskoj traci odabrao pristup stranici "Kontakt" (UC5)
- **Opis osnovnog tijeka:**
 1. Korisnik unosi tražene podatke
 2. Korisnik potvrđuje svoj zahtjev
 3. EmailJS servis šalje ispunjenu kontaktnu formu autoškoli

UC8 - Pregled svih instruktorskih računa

- **Glavni sudionik:** Administrator
- **Cilj:** Prikaz svih instruktorskih računa
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Administratora
- **Opis osnovnog tijeka:**
 1. Administrator odabire opciju pregleda instruktorskih računa
 2. Prikazuje se popis svih instruktorskih računa

UC9 - Pregled svih kandidatskih računa

- **Glavni sudionik:** Administrator i Instruktor
- **Cilj:** Prikaz svih kandidatskih računa
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Administratora ili Instruktora
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda kandidatskih računa
 2. Prikazuje se popis svih kandidatskih računa

UC10 - Pregled vlastitog kalendara

- **Glavni sudionik:** Kandidat i Instruktor
- **Cilj:** Prikaz kalendara
- **Sudionici:** Baza podataka

- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Kandidata ili Instruktor
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju pregleda svog kalendara
 2. Prikazuje se kalendar

UC11 - Uređivanje vlastitog kalendara

- **Glavni sudionik:** Kandidat i Instruktor
- **Cilj:** Uređivanje kalendara
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav, ima ovlasti Kandidata ili Instruktor i otvorena mu je stranica sa vlastitim kalendarom
- **Opis osnovnog tijeka:**
 - 1.a Korisnik odabire postojeći vremenski interval
 - 1.b Korisnik odabire željeni vremenski interval
 1. Korisnik unosi/uređuje željeni naslov
 2. Korisnik potvrđuje svoj izbor

UC12 - Pregled vlastitog napretka

- **Glavni sudionik:** Kandidat
- **Cilj:** Prikaz stranice sa napretkom kandidata
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Kandidata
- **Opis osnovnog tijeka:**
 1. Kandidat odabire opciju pregleda vlastitog napretka
 2. Prikazuje se stranica sa napretkom kandidata

UC13 - Pregled napretka kandidata

- **Glavni sudionik:** Administrator i Instruktor
- **Cilj:** Prikaz stranice sa napretkom kandidata
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Administratora ili Instruktor
- **Opis osnovnog tijeka:**
 1. Iz popisa svih kandidata korisnik odabire željenog kandidata
 2. Prikazuje se stranica sa napretkom kandidata

UC14 - Pregled instruktorovog kalendara

- **Glavni sudionik:** Administrator

- **Cilj:** Prikaz kalendara željenog korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav i ima ovlasti Administratora
- **Opis osnovnog tijeka:**
 1. Admin odabire željenog instruktora iz popisa svih instruktora
 2. Prikazuje se stranica sa kalendarom željenog korisnika

UC15 - Pregled kandidatovog kalendara

- **Glavni sudionik:** Administrator i Instruktor
- **Cilj:** Prikaz kalendara željenog korisnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav, ima ovlasti Administratora ili Instruktora i prikazana mu je stranica Napredak željenog kandidata
- **Opis osnovnog tijeka:**
 1. Korisnik na stranici Napredak odabranog kandidata odabire opciju prikaza kalendara
 2. Prikazuje se stranica sa kalendarom željenog korisnika

UC16 - Uređivanje korisničkog kalendara

- **Glavni sudionik:** Administrator i Instruktor
- **Cilj:** Uređivanje korisničkog kalendara
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav, ima ovlasti Kandidata ili Instruktor i otvorena mu je stranica sa korisničkim kalendarom
- **Opis osnovnog tijeka:**
 - 1.a Korisnik odabire postojeći vremenski interval
 - 1.b Korisnik odabire željeni vremenski interval
 1. Korisnik unosi/uređuje željeni naslov
 2. Korisnik potvrđuje svoj izbor

UC17 - Prikaz bilješke za održani sat

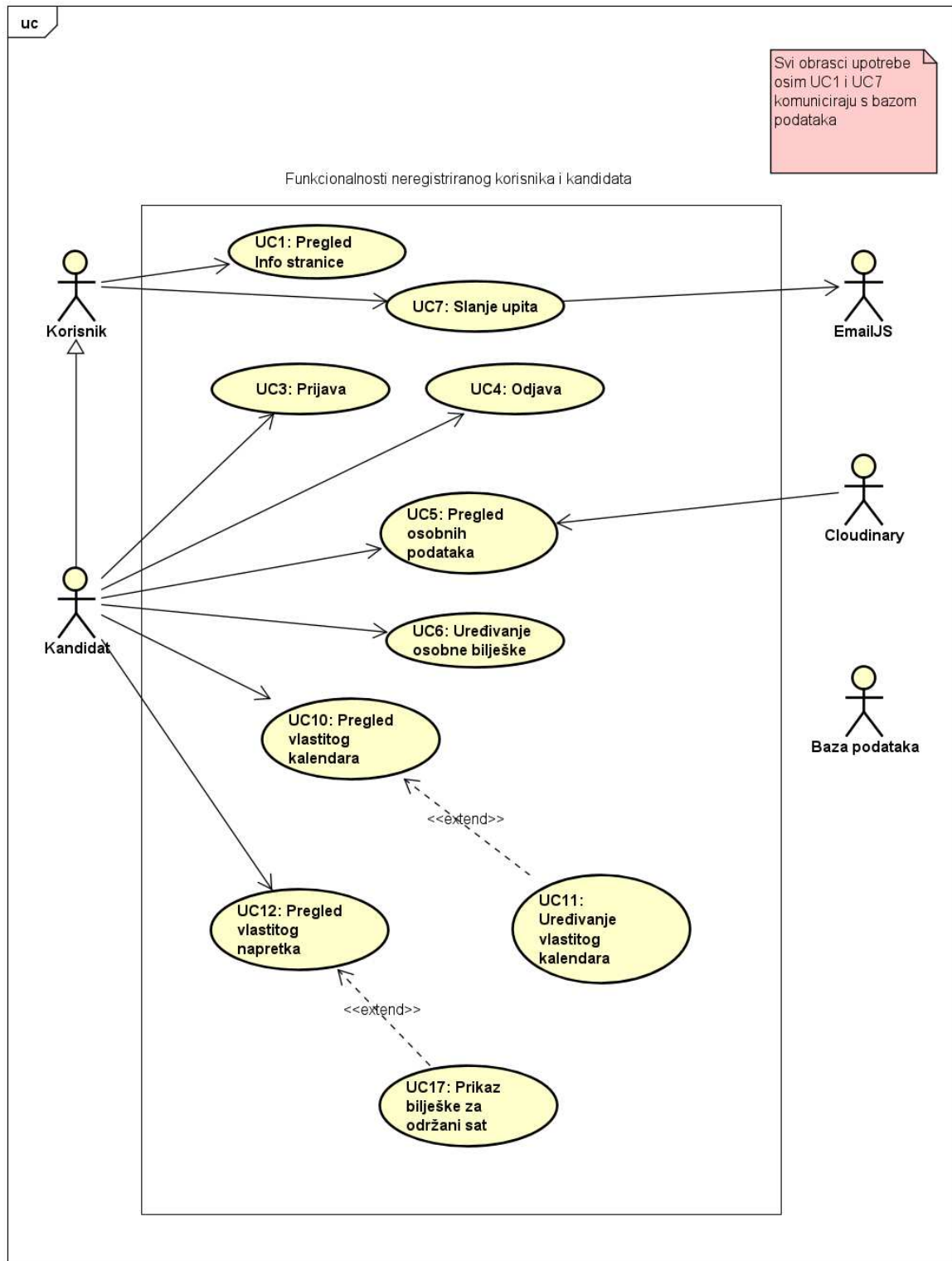
- **Glavni sudionik:** Prijavljeni korisnik (Administrator, Instruktor, Kandidat)
- **Cilj:** Prikaz bilješke za održani sat obuke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav, ima ovlasti nekog od prijavljenih korisnika i otvorena mu je stranica sa napretkom
- **Opis osnovnog tijeka:**
 1. Korisnik odabire željeni sat u traci napretka
 - 2.a Prikazuje se bilješka za održani sat

2.a Prikazuju se prazan skočni prostor - nema bilješke za odabrani sat

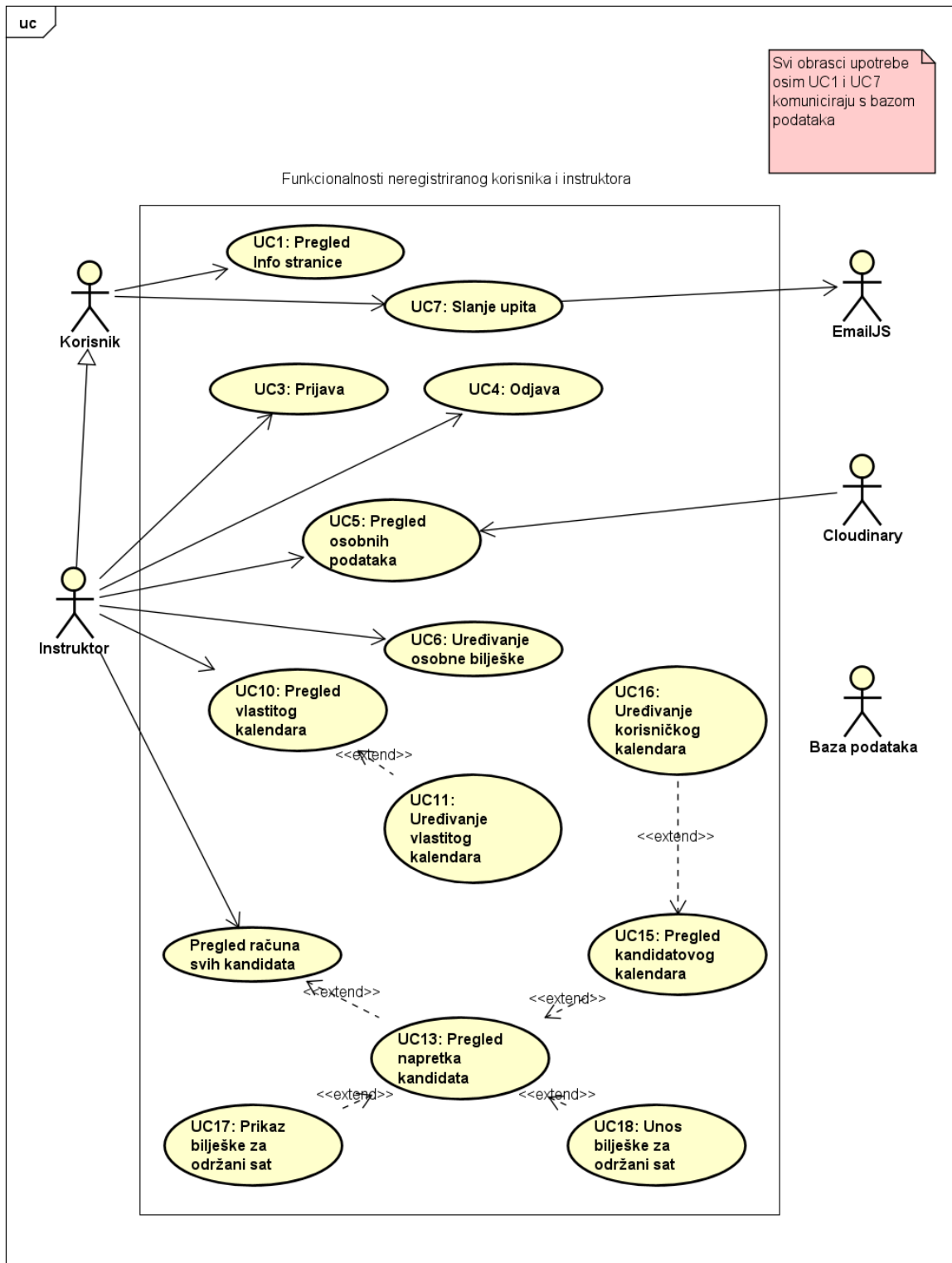
UC18 - Unos bilješke za održani sat

- **Glavni sudionik:** Administrator i Instruktor
- **Cilj:** Unos bilješke za održani sat obuke
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik je trenutno prijavljen u sustav, ima ovlasti Administratora i Instruktor otvorena mu je stranica sa napretkom
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju unosa bilješke za željeni sat
 2. Korisnik unosi tražene podatke
 3. Korisnik potvrđuje svoj izbor
 4. Ažurira se baza podataka

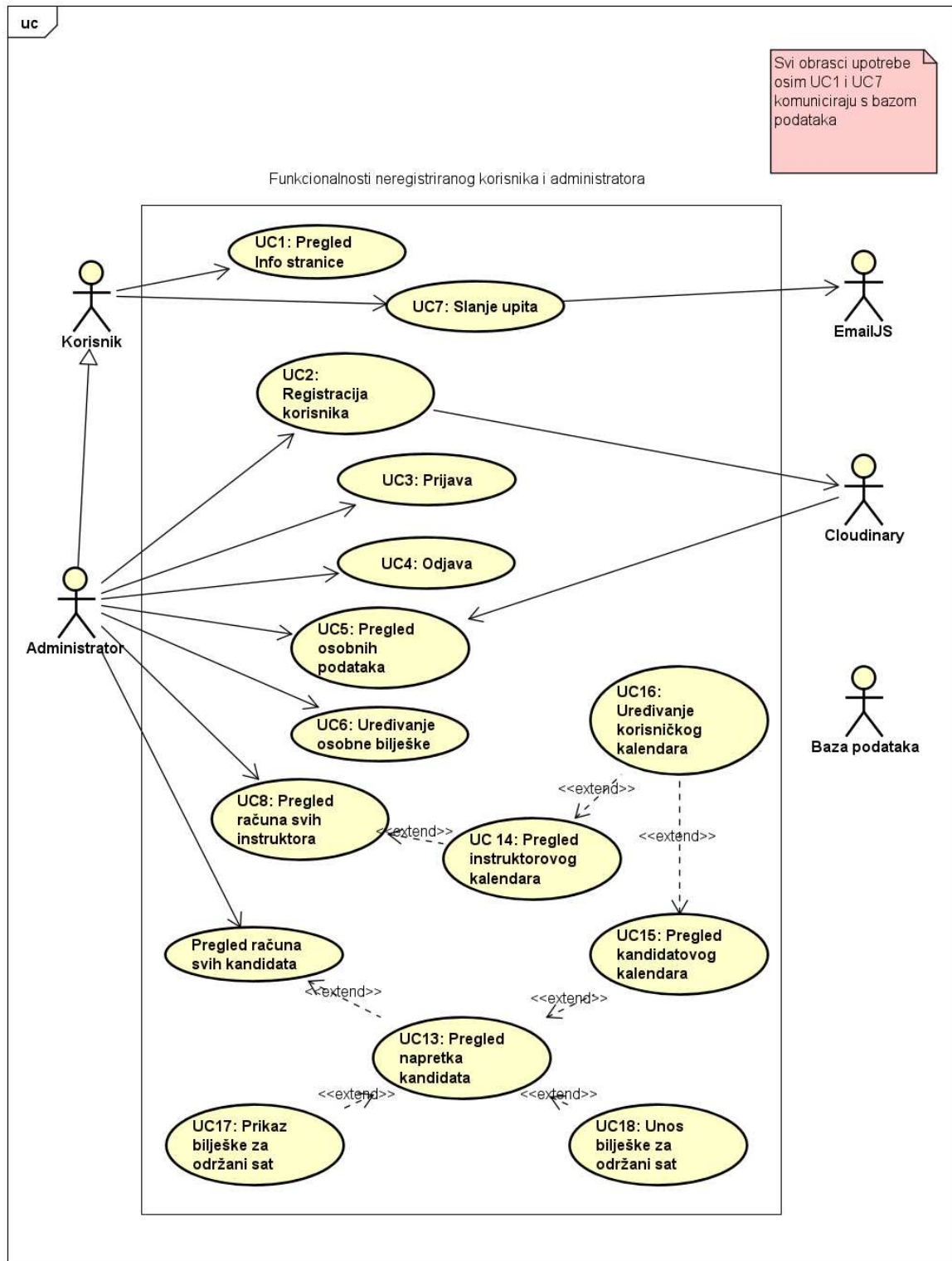
Na slici 1.1 možemo vidjeti grafički prikaz dijagrama obrazaca uporabe koji prikazuje funkcionalnosti neprijavljenog korisnika i kandidata. Funkcionalnosti instruktora možemo vidjeti na slici 1.2, dok su interakcije administratora i sustava prikazane na dijagramu na slici 1.3.



Slika 1.1: Dijagram obrazaca uporabe, funkcionalnost korisnika i Kandidata



Slika 1.2: Dijagram obrazaca uporabe, funkcionalnost korisnika i Instruktora



Slika 1.3: Dijagram obrazaca uporabe, funkcionalnost korisnika i Administratora

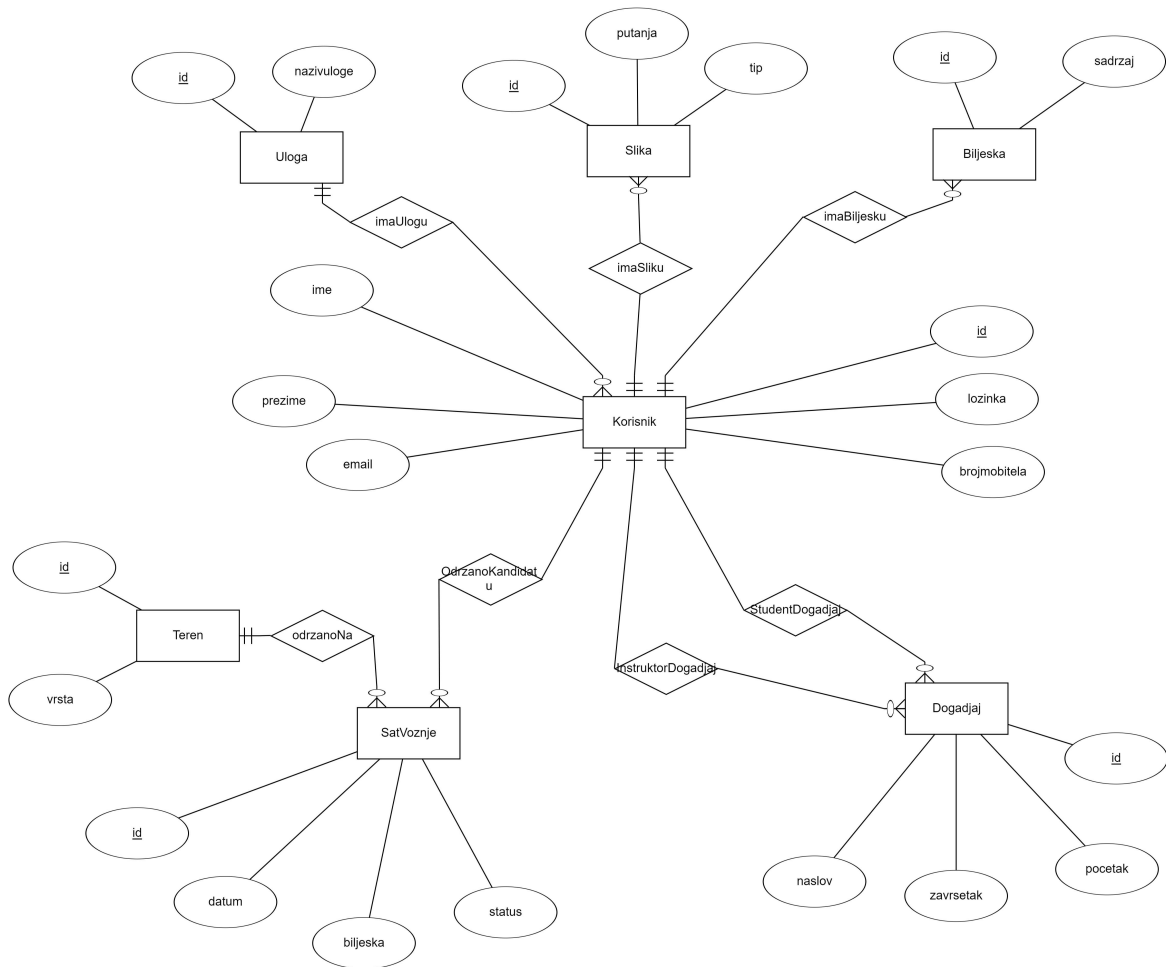
2. Baza podataka

2.1 ER model

Za konceptualno modeliranje baze podataka najprije je definiran entitetsko-relacijski model koji pomoću entiteta predstavlja objekte u stvarnom svijetu. Svaki entitet opisan je svojim atributima. Posebno se definira ključni atribut koji definira svojstvo koje identificira entitet. Relacije opisuju kako su entiteti međusobno povezani. Na slici 2.1. možemo vidjeti grafički prikaz ER modela. Svaki entitet prikazan je pravokutnikom, a njegovi atributi elipsama. Relacije su prikazane rombovima, a same veze linijama. Baza podataka ima sljedeće entitete

- Uloga
- Korisnik
- Slika
- Bilješka
- Teren
- SatiVožnje
- Dogadjaj

Svaki od prethodno navedenih entiteta zadovoljava prvu, drugu i treću normalnu formu. Prva normalna forma zahtjeva da svi atributi sadrže nedjeljive vrijednosti i da svaki atribut ima jedinstveni identifikator. Druga normalna forma zahtjeva da svi neključni atributi zavise od cijelog primarnog ključa, ne samo jednog njegovog dijela. Na kraju, treća normalna forma zahtjeva nepostojanje tranzitivne zavisnosti, tj. zahtjeva da neključni atributi ne zavise od drugih neključnih atributa.



Slika 2.1: ER dijagram baze podataka

Svaki korisnik ima jedinstvenu ulogu, sliku i bilješku koja se odnosi samo na njega. Unos bilješke i slike za pojedinog korisnika nije obavezan, ali svaki registriran korisnik mora imati dodijeljenu ulogu. Sat vožnje može se održati na jednoj vrsti terena, dok se teren može povezati s više satova. Svakom korisniku može se održati više sati vožnje, ali jedan zapis vožnje odnosi se samo na jednog korisnika, odnosno kandidata. Događaj je povezan s korisnikom tako da se odnosi na minimalno jednog, a maksimalno dva korisnika. Postoje događaji koji se odnose samo na instruktora ili samo na kandidata, primjerice raspoloživost kandidata za održavanje obuke, dok sam termin održavanja obuke se odnosi i na kandidata i na instruktora.

2.2 Opis relacija

Entitet **Uloga** sadrži informaciju o ulozi korisnika u sustavu. Svi atributi moraju biti definirani i unose se za svakog korisnika.

Uloga		
ulogaID	BIGINT	jedinstveni identifikator uloge
nazivuloge	VARCHAR	opis korisničke uloge

Entitet **Korisnik** sadrži korisničke podatke o svakom korisniku u sustavu. Svi atributi moraju biti definirani budući da entitet sadrži sve relevantne informacije o registriranom korisniku aplikacije.

Korisnik		
korisnikID	BIGINT	jedinstveni identifikator korisnika
ime	VARCHAR	ime korisnika
prezime	VARCHAR	prezime korisnika
email	VARCHAR	adresa elektroničke pošte
lozinka	VARCHAR	lozinka korisničkog računa
brojmobitela	VARCHAR	broj mobitela korisnika
ulogaID	BIGINT	jedinstveni identifikator uloge korisnika

Entitet **Biljeska** sadrži informaciju o bilješci za određenog korisnika u sustavu. Svi atributi moraju biti definirani i: obavezan unos za svakog korisnika.

Bilješka		
biljeskaID	BIGINT	jedinstveni identifikator bilješke
sadržaj	VARCHAR	tekstualni zapis bilješke
korisnikID	BIGINT	jedinstveni identifikator korisnika

Entitet **Slika** sadrži putanju do slike određenog korisnika u sustavu. Svi atributi moraju biti definirani: obavezan unos za svakog korisnika.

Slika		
slikaID	BIGINT	jedinstveni identifikator zapisa
tip	VARCHAR	tip medijskog sadržaja (image)
putanja	VARCHAR	putanja do slike
korisnikID	BIGINT	jedinstveni identifikator korisnika

Entitet **Teren** sadrži informaciju o vrsti terena na kojoj je održan sat vožnje. Moguće

vrijednosti su "V" - vježbalište koje označava održavanje sata na poligonu i "C" - cesta koja označava održavanje sata na otvorenoj cesti. Svi atributi moraju biti definirani.

Teren		
terenID	BIGINT	jedinstveni identifikator terena
vrsta	VARCHAR	vrsta terena

Entitet **SatVoznje** sadrži informaciju o održanom satu vožnje uključujući polaznika autoškole i teren na kojem je održan sat vožnje. Navedeni atributi predstavljaju zapravo strane ključeve. Svi atributi moraju biti definirani.

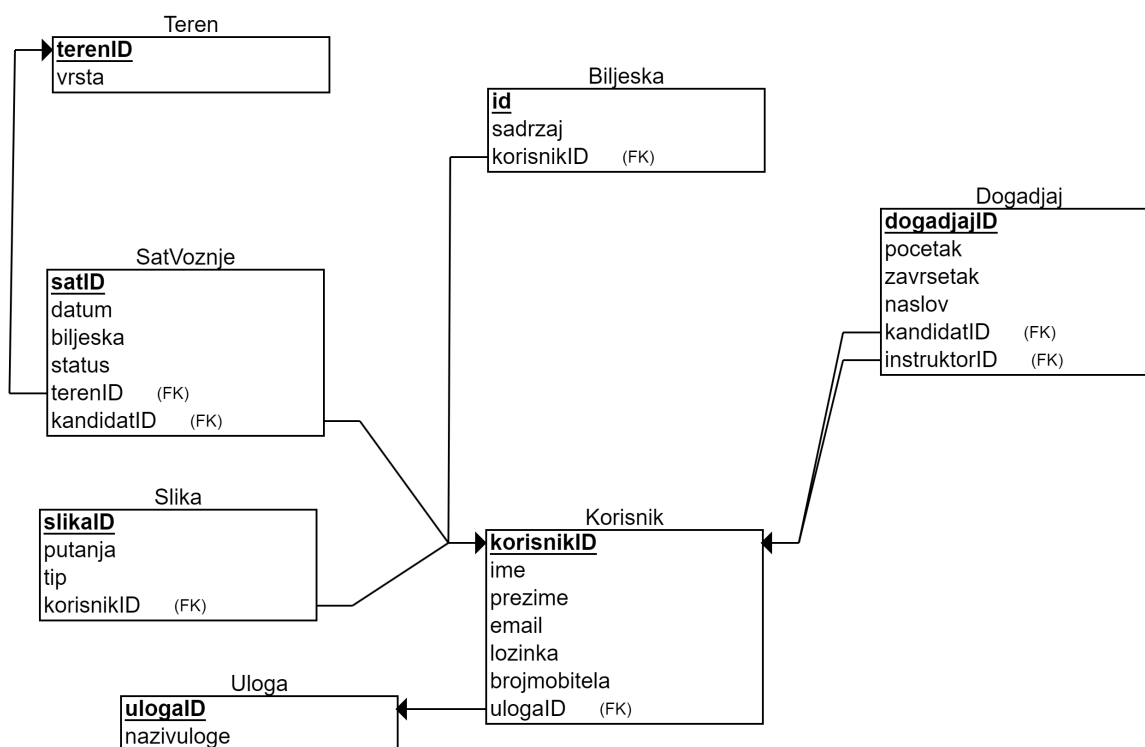
SatVoznje		
satID	BIGINT	jedinstveni identifikator održanog sata
datum	DATE	datum održanog sata
biljeska	VARCHAR	komentar od strane instruktora
status	VARCHAR	status
kandidatID	BIGINT	jedinstveni identifikator kandidata kojem je održan sat obuke
terenID	BIGINT	jedinstveni identifikator terena na kojem je održan sat

Entitet **Događaj** sadrži informaciju o pojedinom zapisu u kalendaru kandidata i instruktora. Jedinstveni identifikatori kandidata i instruktora su strani ključevi. Budući da svaki korisnik može sam unositi događaje u svoj kalendar, te instruktor može kandidatu zakazati sat, određeni zapisi mogu imati oba strana ključa definirana, dok je dozvoljeno izostavljanje jednog od ključeva ovisno u korisniku koji unosi događaj

Događaj		
događajID	BIGINT	jedinstveni identifikator događaja
pocetak	TIMESTAMP	datum održanog sata
zavrsetak	TIMESTAMP	naziv terena
naslov	VARCHAR	naslov događanja
kandidatID	BIGINT	jedinstveni identifikator kandidata
instruktorID	BIGINT	jedinstveni identifikator instruktora

2.3 Relacijski model

Na slici 3.1. možemo vidjeti relacijsku shemu baze podataka.



Slika 2.2: Dijagram baze podataka

Sustav koristi H2 relacijsku bazu podataka koja se posebno ističe po lakoj integraciji s aplikacijama razvijenim u programskom jeziku Java. Glavna karakteristika aplikacije je njena brzina i učinkovitost zbog organizacije i pohrane podataka u radnu memoriju, dok se inače u sustavima za upravljanje podacima koriste mehanizmi za pohranu na disku. Kao takva, H2 baza pruža podršku standardnim SQL naredbama. Razvoj i testiranje same baze posebno olakšava i grafičko korisničko sučelje kojem se pristupa putem weba. Iako je H2 odličan izbor za razvojne faze projekta, koristit će se i u proizvodnom okruženju web aplikacije zbog manjih zahtjeva na bazu podataka. Budući da u sustavu postoje samo tri uloge korisnika, zbog jednostavnosti u implementaciji su združeni entiteti **Korisnik** i **Uloga**. Iz istog razloga združeni su i entiteti **Teren** i **SatVoznje**. U konačnici, implementacija aplikacije je realizirana pomoću pet entiteta: **Korisnik**, **Slika**, **Bilješka**, **SatVožnje**, **Dogadjaj**.

3. Implementacija

3.1 Korištene tehnologije

3.1.1 Spring Boot

Za razvoj backenda web aplikacije "Vožnja +" izabran je Spring Boot razvojni okvir koji znatno pojednostavljuje i olakšava razvoj aplikacija u Programskom jeziku Java. Glavna prednost Spring Boota je smanjena konfiguracijska složenost i ubrzan razvoj aplikacije. Razvoj aplikacije započeo je upotrebom Spring inicijalizatora projekta na kojem se odabire projekt (Maven), programski jezik (Java) i verzija Spring Boota. Pri inicijalizaciji dodaju se potrebne ovisnosti koje se kasnije po potrebi nadograđuju u pom.xml datoteci. Osnovne konfiguracijske postavke definiraju se u datoteci application.properties. U razvoju aplikacije ključnu ulogu igraju anotacije koje omogućuju konfiguraciju i upravljanje komponentama bez potrebne za opsežnim XML konfiguracijama. Primjerice, anotacija @Autowired koristi se za automatsku injekciju zavisnosti koja eliminira potrebu za ručnom konfiguracijom. To znači da umjesto ručnog konfiguriranja komponente pri svakom njenom korištenju, sve što trebamo napraviti je iznad instanciranja komponente dodati spomenutu anotaciju. Ostatak anotacija bit će objašnjeni u poglavlju "Poslužitelj" gdje će se detaljnije objasniti implementacija poslužiteljske strane i korištenje karakterističnih Spring Boot anotacija. Korišteni radni okvir je IntelliJ.

3.1.2 React

Za razvoj frontenda korišten je skriptni jezik JavaScript koji se izvršava u web pregledniku korisnika web aplikacije. Za razvoj korisničkih sučelja korištena je React razvojna biblioteka. React je biblioteka otvorenog koda temeljena na komponentama koje se koriste s izgradnju sučelja. Svaka komponenta ima mogućnost održavanja svog stanja i komponiranja u složenije dijelove grafičkog sučelja. Glavna karakteristika Reacta je njegov dizajn koji omogućava ponovno renderiranje komponente tek pri promjeni podataka što je učinkovitije nego ponovno renderiranje cijele stranice. Pored Reacta, za učinkovit razvoj korisničkih sučelja korištena je Chakra UI biblioteka komponenti koja pruža polugotove gradivne blokove za izgradnju aplikacije. Svaki gradivni element prilagođen je svojoj namjerno kako bi se osiguralo što bolje korisničko iskustvo. Korišteni radni okvir je Visual Studio Code.

3.1.3 Cloudinary

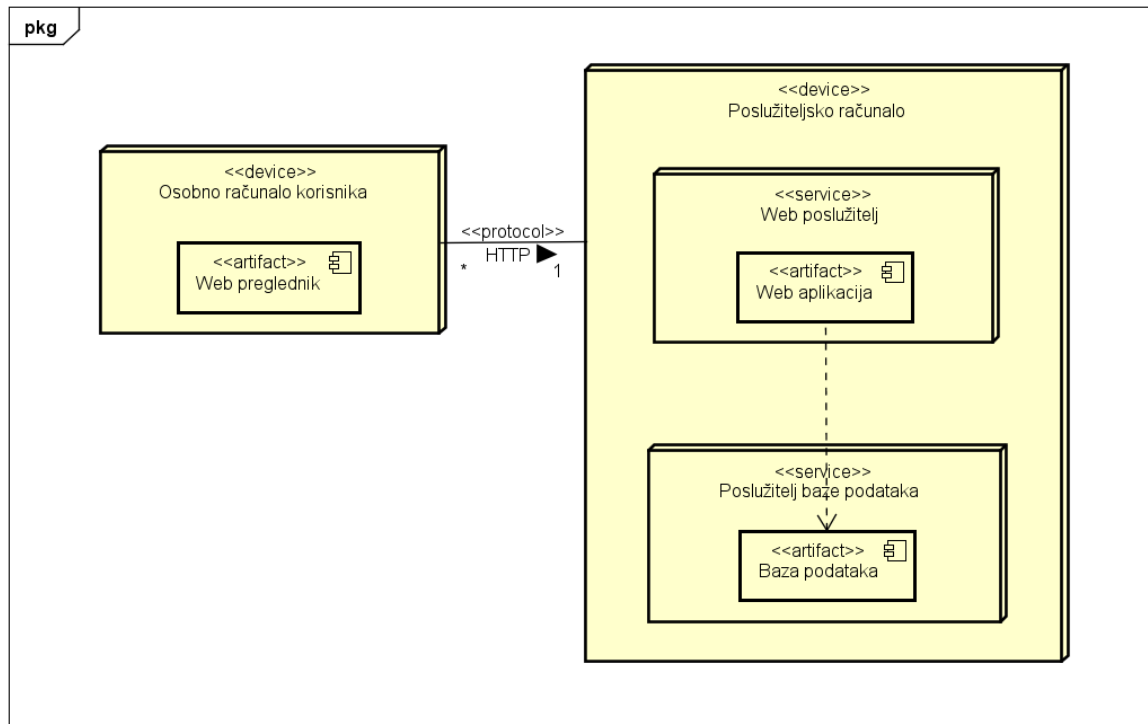
Za skladištenje i manipulaciju medijskim sadržajem u aplikaciji korištena je Cloudinary platforma kojoj se pristupa pomoću API-ja (engl. application programming interface). Glavne funkcionalnosti Cloudinary platforme koje su korištene za aplikaciju su skladištenje i dohvat fotografija. Platforma osigurava sigurnost i podržava kontroliran pristup budući da se pri konfiguraciji servisa koriste Cloud name (ime oblaka), API key (API ključ) i API secret (API tajna) koji se provjeravaju pri svakom pristupu.

3.1.4 EmailJS

Integracija s Gmail servisom ostvarena je pomoću EmailJS servisa koji omogućava direktno slanje emailova iz JavaScript aplikacija bez potrebe za poslužiteljem. Na ovaj način omogućeno je učinkovito slanje mailova direktno iz aplikacije. Nakon konfiguracije, servis brine o komunikaciji i autentificiranom pristupu svojim uslugama.

3.2 Dijagram razmještaja sustava

Na slici 4.1. možemo vidjeti dijagram razmještaja sustava. Za arhitekturu odabrana je arhitektura poznata po nazivu "klijent-poslužitelj". Glavna karakteristika spomenute arhitekture je razdvojenog funkcionalnosti na usluge koje pruža poslužitelj i usluge koje pruža klijent. Korisnici web aplikacije pristupaju aplikaciji putem web preglednika na vlastitom računalu. Korisničko računalo ima ulogu klijenta koje u ime korisnika obavlja zahtjeve prema poslužitelju. Web poslužitelj i poslužitelj baze podataka nalaze se na zasebnom računalu. Interakcija između korisničkog i poslužiteljskog računala odvija se putem HTTP (HyperText Transfer Protocol) protokola. Osnovne karakteristike HTTP protokola su komunikacija putem zahtjeva i odgovora te neovisnost novog zahtjeva o prethodno poslanim zahtjevima.



Slika 3.1: Dijagram razmještaja sustava

3.2.1 Klijent

Klijentska strana zasnovana je na funkcijskim komponentama pomoću kojih se gradi izgled cijele aplikacije. React koristi virtualni DOM (engl. Data Object Model) za poboljšanje svojih performansi. Kada se stanje aplikacije promijeni, React prvo ažurira virtualni DOM koji onda uspoređuje s pravim modelom kako bi minimalno ažurirao stvarni model. App.js je ključna komponenta u kojoj su definirane korisničke rute i komponente koje se koriste za izgradnju pojedinih sučelja. Sučelje koje se prikazuje korisniku ovisi u njegovoj ulozi koja se sprema u memoriju sjednice. Svaka komponenta ima svoje unutarnje stanje koje određuje kako se komponenta prikazuje i ponaša. Za očuvanje stanja i ažuriranje komponenti koriste se dvije posebne funkcionalnosti Reacta: useState i useEffect hook-ovi. Hook-ovi služe za baratiranje s popratnim efektima pojedine komponente. useState hook služi za održavanje lokalnog stanja funkcijskim komponentama, a useEffect hook služi za ažuriranje komponente, odnosno njenih podataka. Kako bi se moglo navigirati između različitih komponenti, korišten je React Router koji omogućava definiranje višestrukih ruta unutar aplikacije.

3.2.2 Poslužitelj

Aplikacija je izgrađena korištenjem objektno orijentirane paradigme i MVC (eng. Model-View- Controller) arhitekturnog obrasca koji razdvaja prezentaciju podataka, njihov do-

hvat i manipulaciju. Implementacija poslužiteljske strane zasnovana je na reprezentacijskom prijenosu stanja, odnosno REST (Representational State Transfer) arhitekturnom stilu čija glavna karakteristika manipulacija podacima korištenjem standardnih HTTP metoda kao što su GET, POST, PUT, i DELETE. Anotacija @Entity označava komponentu koja reprezentira informacije potrebne za rad aplikacije. Svaki entitet predstavlja podatke koji se prezentiraju kao tablica u bazi podataka. Korištenjem anotacije @Controller označena je kontrolerska komponenta koja je odgovorna za obradu zahtjeva koji dolaze od klijentske strane. Servisni sloj sadrži poslovnu logiku aplikacije te obavljaju sve potrebne operacije nad podacima. Servisi su zasebne komponente odvojene od kontrolera te se označavaju anotacijom @Service. Za podatkovni sloj zaduženo je standardizirano sučelje prema podacima pod nazivom JPA (Java Persistence API). Podatkovno sučelje naznačeno je anotacijom @Repository.

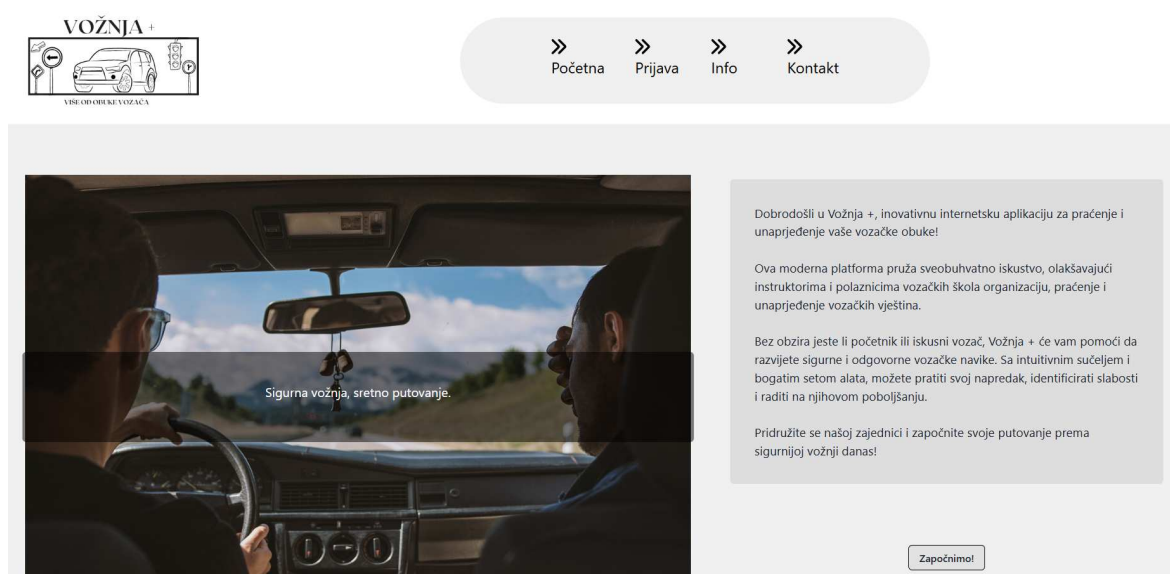
3.3 Sigurnost

Za sigurnost sustava, odnosno implementaciju autentifikacije i autorizacije korisnika korišten je JSON Web Token (JWT). Korištenje tokena standardan je način sigurnog prijenosa podataka između dvaju partija, u ovom slučaju su to klijent i poslužitelj. Pri prijavi u sustav pomoću korisničkih podataka, korisnik, odnosno njegov web preglednik kao odgovor od poslužitelja dobiva izgeneriran token koji pohranjuje u sjednicu i koristi pri svakoj interakciji s poslužiteljem do kraja korištenja aplikacije. S druge strane, pri svakom zahtjevu poslužitelj iz tokena izvlači e-mail te provjerava postoji li korisnik u bazi te u skladu s provjerom obrađuje poslani zahtjev. Kako bi se spriječio neovlašten pristup određenim stranicama, pored autentifikacije putem JWT tokena, svakom korisniku u sustavu dodeljena je uloga i u skladu s njom definirane korisničke ovlasti. Pri slanju zahtjeva klijent obavezno uključuje token u autorizacijsko zaglavlje svakog HTTP zahtjeva kako bi mogao pristupiti zaštićenim resursima.

4. Sučelja web aplikacije

Sučelja anonimnog korisnika

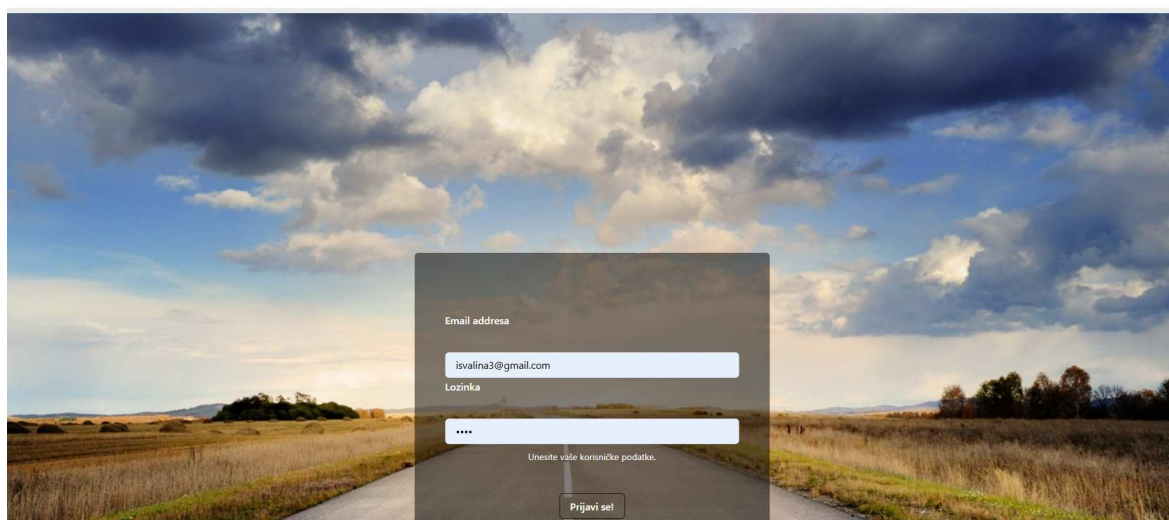
Na slici 4.1. možemo vidjeti sučelje koje se prikazuje pri pokretanju aplikacije. Svakom anonimnom korisniku su u navigacijskoj traci omogućene opcije za povratak na početnu stranicu, prijavu u aplikaciju, pristup stranici s informacijama i pristup kontaktnoj formi. Klikom na gumb "Započnimo" korisnika se također preusmjerava na stranicu za prijavu



Slika 4.1: Početna stranica

Da bi se korisnik prijavio u sustav, u navigacijskoj traci odabire opciju za prijavu. Sučelje koje se prikazuje može se vidjeti na slici 4.2.

Nemaš potrebne podatke za prijavu? Kontaktiraj autoškolu putem kontaktnog obrasca klikom na "Kontakt".



Slika 4.2: Stranica za prijavu

Ako korisnik želi dobiti više informacija o autoškoli i aplikaciji, može dabrati opciju "Info" u navigacijskoj traci. Sučelje koje mu se prikazuje je na slici 4.3. i 4.4.



» Početna » Prijava » Info » Kontakt

Autoškola "Vožnja +" nije samo obrazovna institucija: to je simbol pouzdanosti, kvalitete i posvećenosti sigurnosti u prometu već više od dva desetljeća. Naša povijest obiluje uspjesima, izazovima i kontinuiranim prilagodbama kako bismo osigurali da naši polaznici dobiju najbolju moguću obuku.

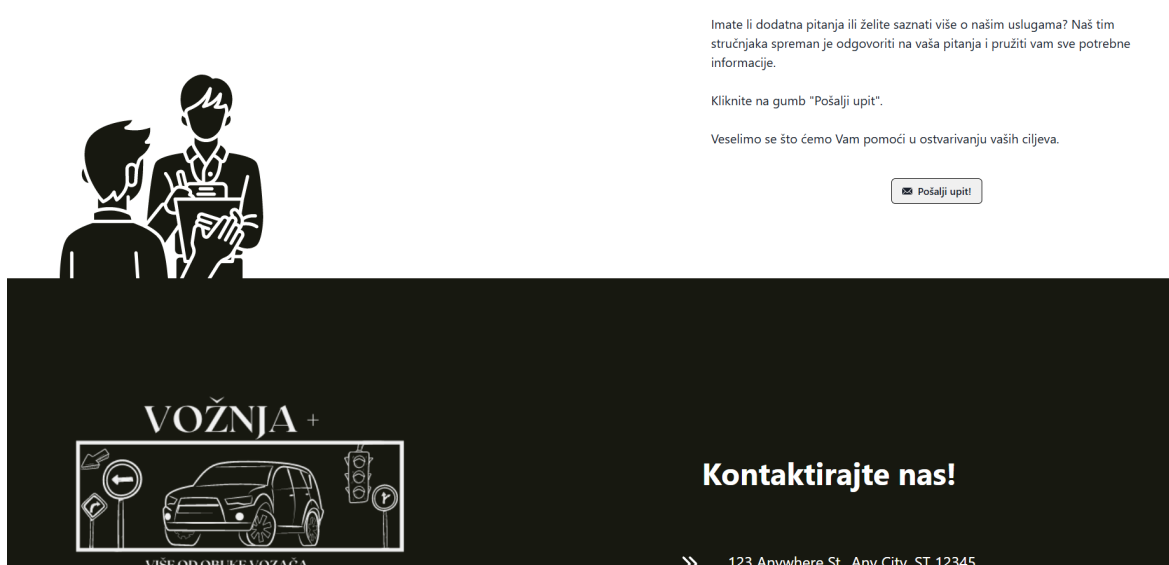
Kroz godine, naša autoškola postala je sinonim za vrhunsku obuku vozačkih vještina i prometne sigurnosti. Osnovana s jasnim ciljem - pružiti vrhunsku obuku i stvoriti sigurne i odgovorne vozače - postali smo priznata institucija koja je iznjedrila mnoge vozače koji voze s povjerenjem i sigurnošću.

Neprestano se prilagođavamo promjenjivim potrebama i zahtjevima suvremenog prometa. Naša obuka se kontinuirano usavršava kako bismo osigurali da naši polaznici dobiju najnovije informacije, tehnike i vještine potrebne za sigurno upravljanje vozilima u različitim prometnim situacijama.

Ponosimo se našom prošalošću i tradicijom, ali istovremeno gledamo prema budućnosti s entuzijazmom i predanošću. Uvijek težimo biti korak ispred, uvodeći nove metode obuke, tehnologije i pristupe kako bismo osigurali da naši vozači budu najbolje pripremljeni za izazove na cestama.

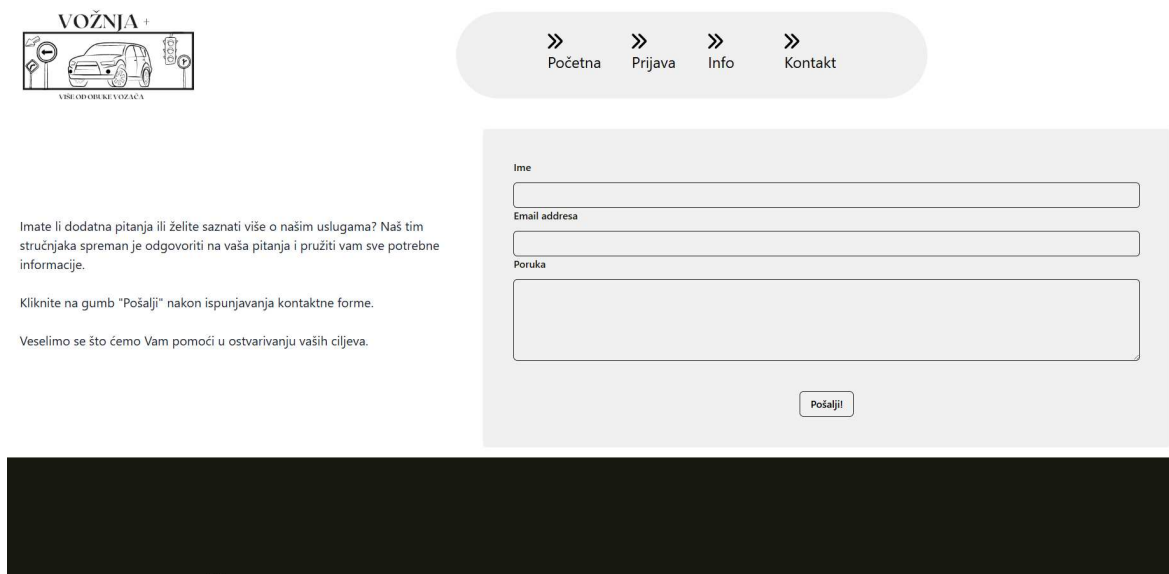
Autoškola "Sigurna Vožnja" ostaje predana svojoj misiji - osposobiti sigurne, odgovorne i samostalne vozače koji će doprinijeti sigurnosti i kvaliteti života na cestama.

Slika 4.3: Stranica s dodatnim informacijama



Slika 4.4: Stranica s dodatnim informacijama

Za slanje kontaktne forme korisnik odabire opciju "Kontakt" u navigacijskoj traci ili "Pošalji upit" te mu se prikazuje sučelje na slici 4.5.



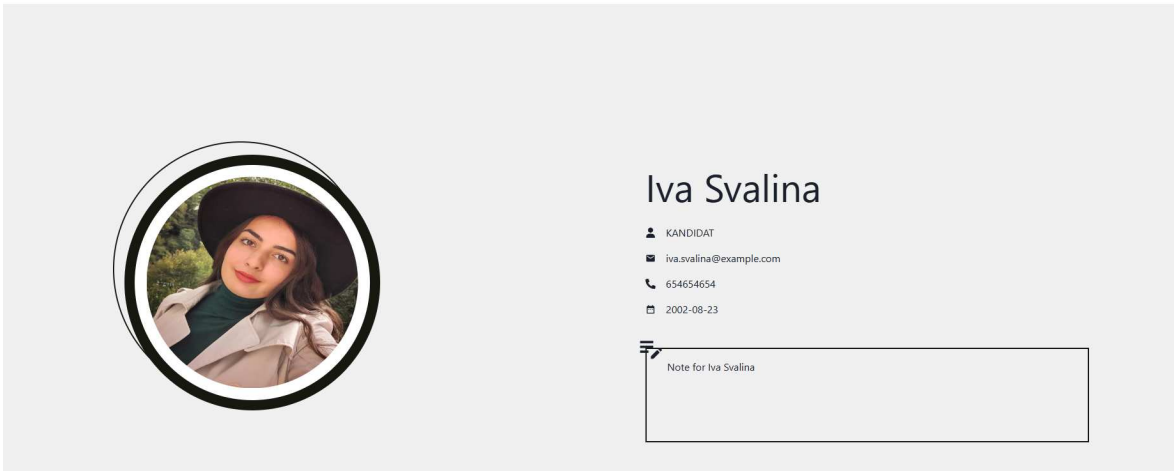
Slika 4.5: Stranica s kontakt formom

Sučelja polaznika autoškole

Nakon što se kandidat prijavi u aplikaciju, pored osnovnih, omogućen mu je pristup dodatnim web stranicama. Na slici 4.6. možemo vidjeti kako izgleda stranica s vlastitim korisničkim podacima kojoj može pristupiti svaki autentificiran korisnik.



- » Početna
- » Info
- » Profil
- » Kalendar
- » Napredak
- » Odjava



Slika 4.6: Stranica s kontakt formom

Odabirom opcija "Kalendar" u navigacijskoj traci korisniku se prikazuje sučelje na slici 4.7. Korisnik događaje u kalendaru može dodavati, uređivati i brisati, što je prikazano na slici 4.8.



- » Početna
- » Info
- » Profil
- » Kalendar
- » Napredak
- » Odjava

Upute za korištenje

U kalendar možete unijeti vremenski period u kojem ste dostupni za održavanje obuke. U označenom periodu instruktor će zakazati termin koji će biti vidljiv u kalendaru.

DODAVANJE

Odaberite vremenski period u kojem želite dodati događaj. To može biti jedan dan ili raspon više dana. U formi da dodavanje događaja unesite potrebne podatke. Svoj odabir potvrdite klikom na gumb "Spremi".

UREDIVANJE

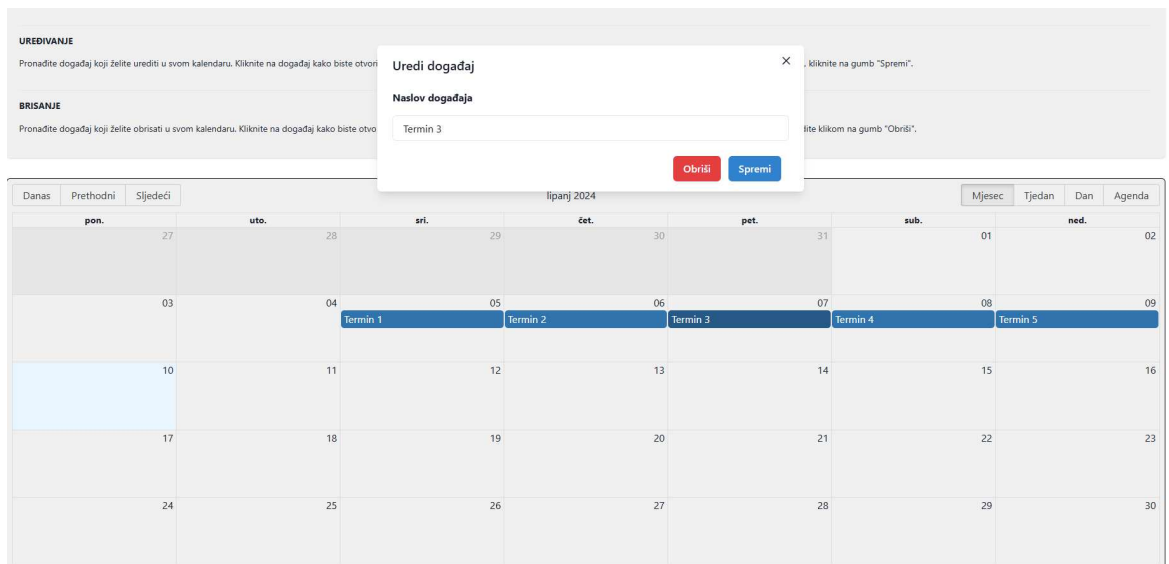
Pronađite događaj koji želite urediti u svom kalendaru. Kliknite na događaj kako biste otvorili formu za uređivanje koja Vam omogućuje izmjenu postojećih detalja događaja. Kada završite s uređivanjem, kliknite na gumb "Spremi".

BRISANJE

Pronađite događaj koji želite obrisati u svom kalendaru. Kliknite na događaj kako biste otvorili formu za uređivanje/brisanje koja Vam omogućuje izmjenu postojećih detalja događaja. Svoj odabir potvrdite klikom na gumb "Obrisi".

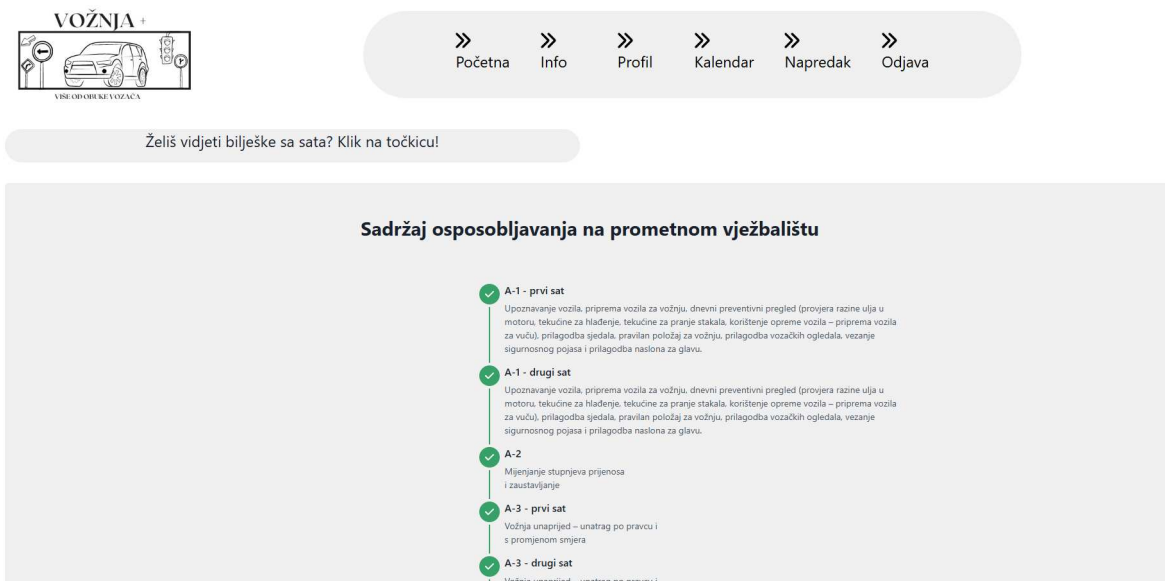
Danas	Prethodni	Sjedeći	lipanj 2024							Mjesec	Tjedan	Dan	Agenda	
	pon.	27	uto.	28	sri.	29	čet.	30	pet.	31	sub.	01	ned.	02
		03		04	termin 1	termin 2	termin 3	termin 4	termin 5					09

Slika 4.7: Stranica s kalendarom

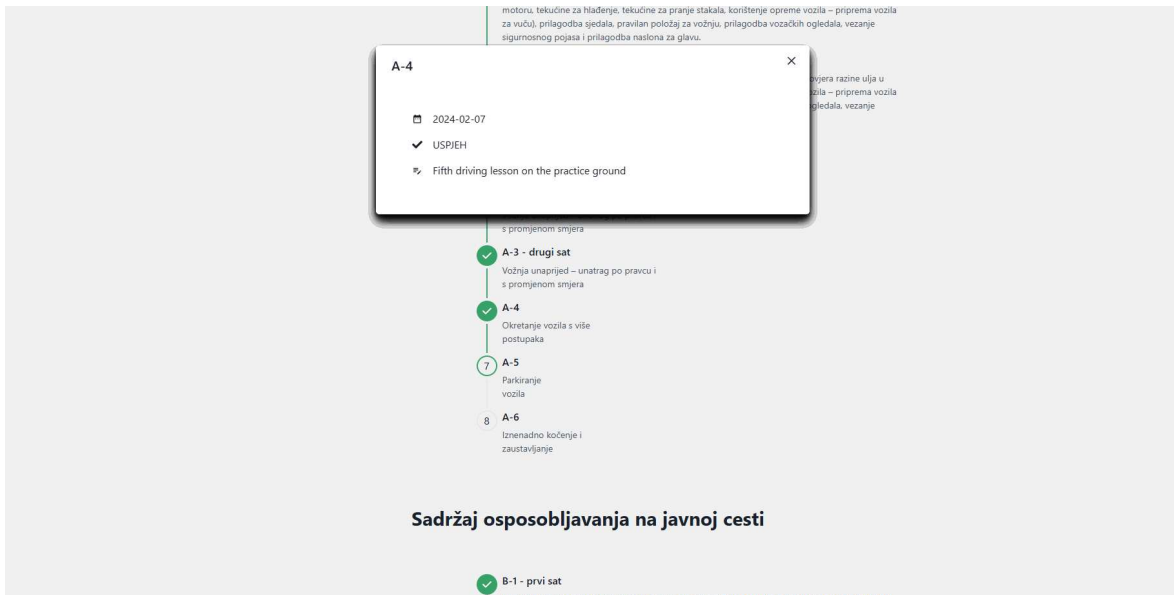


Slika 4.8: Uređivanje događaja u kalendaru

Za prikaz stranice napredak na slici 4.9. korisnik odabire opciju "Napredak" u navigacijskoj traci. Svaki predviđeni sat vožnje ima pripadajuću bilješku koja se može vidjeti na slici 4.10.



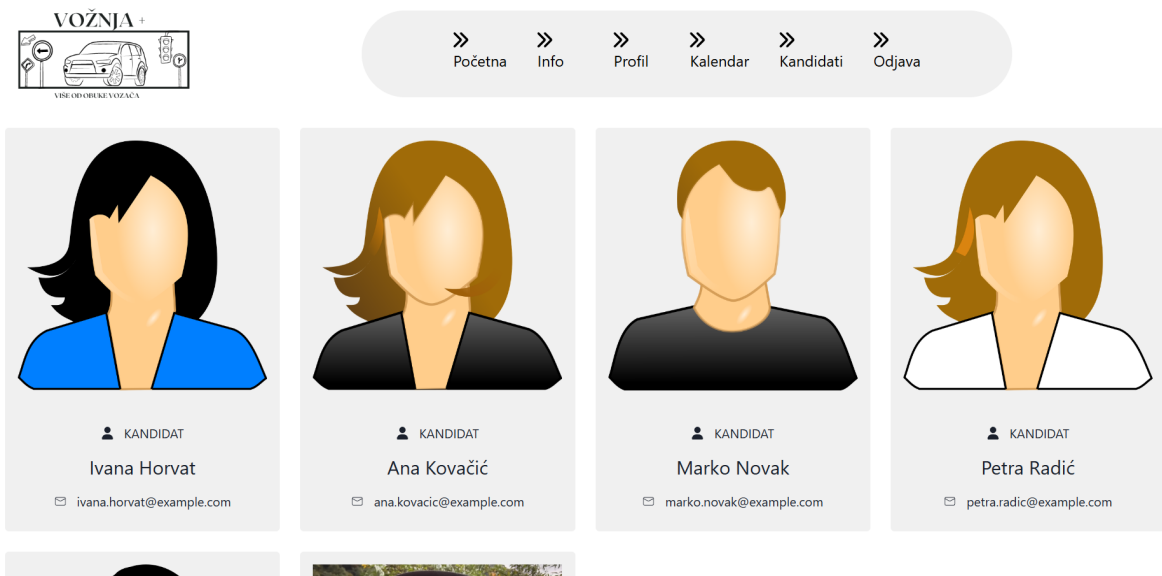
Slika 4.9: Napredak stranica



Slika 4.10: Bilješka za sat

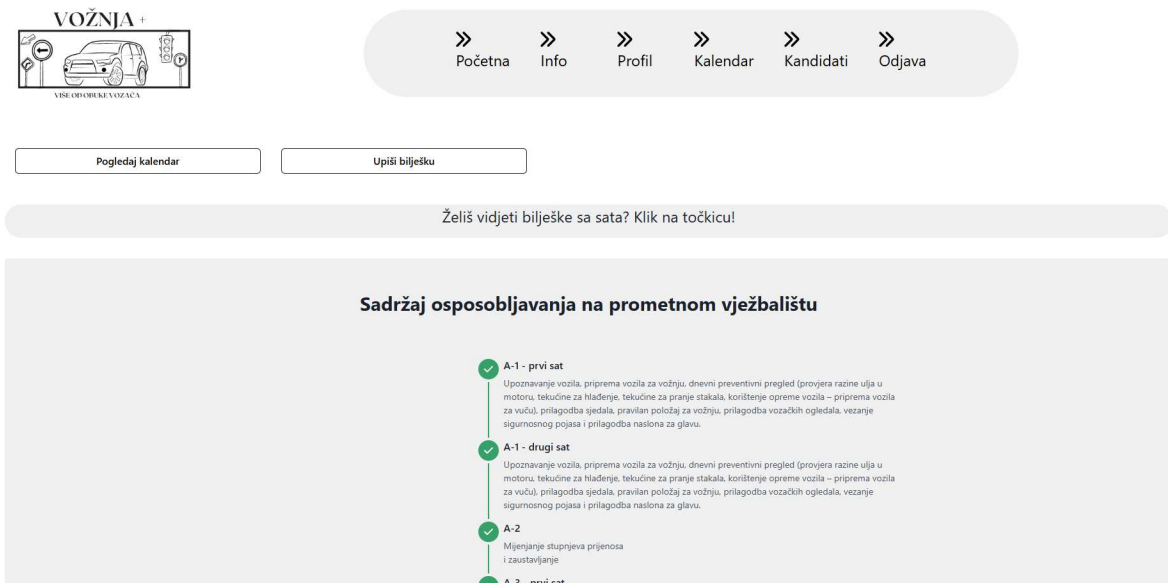
Sučelja instruktora autoškole

Nakon što se instruktor prijavi u aplikaciju, pored osnovnih, omogućen mu je pristup dodatnim web stranicama. Na slici 4.11. možemo vidjeti kako izgleda stranica sa svim kandidatima kojoj mogu pristupiti instruktori i administratori.

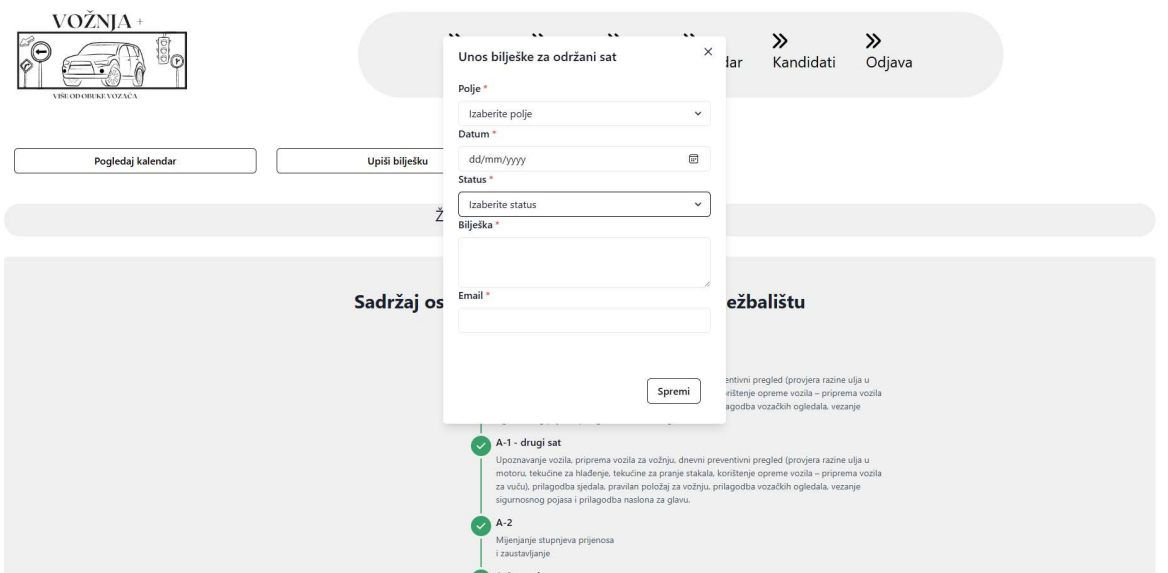


Slika 4.11: Prikaz svih kandidata

Nakon odabira na određenog kandidata iz prikaza prikazuje se stranica na slici 4.12. Putem prikazanog sučelja instruktor može pregledati kandidatov kalendar te u njega upisati, urediti ili obrisati termin. Osim toga, instruktor unosi bilješke za održani sat putem sučelja prikazanog na slici 4.13.



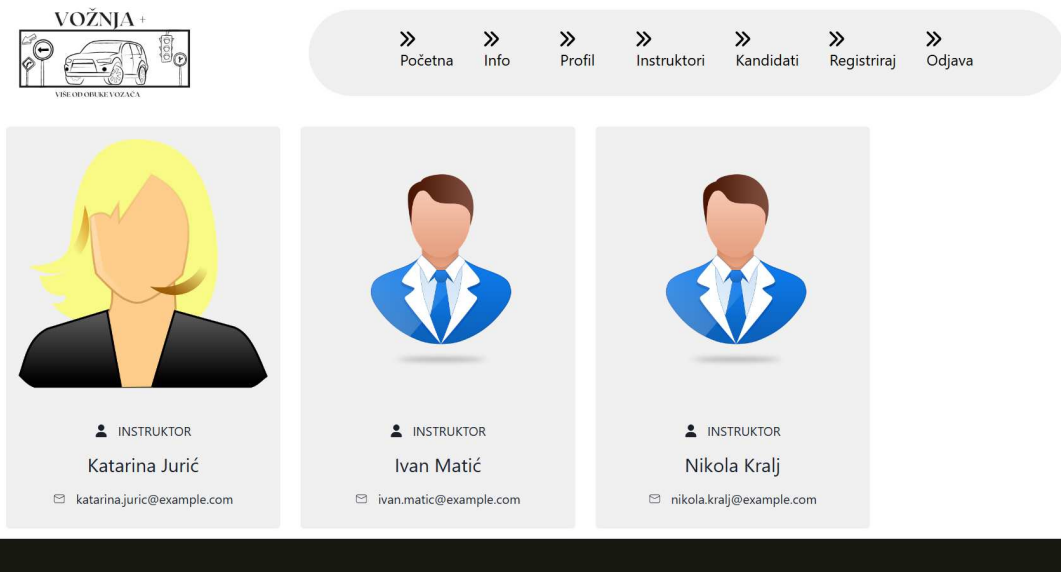
Slika 4.12: Prikaz stranice za napredak



Slika 4.13: Unos bilješke sa sata

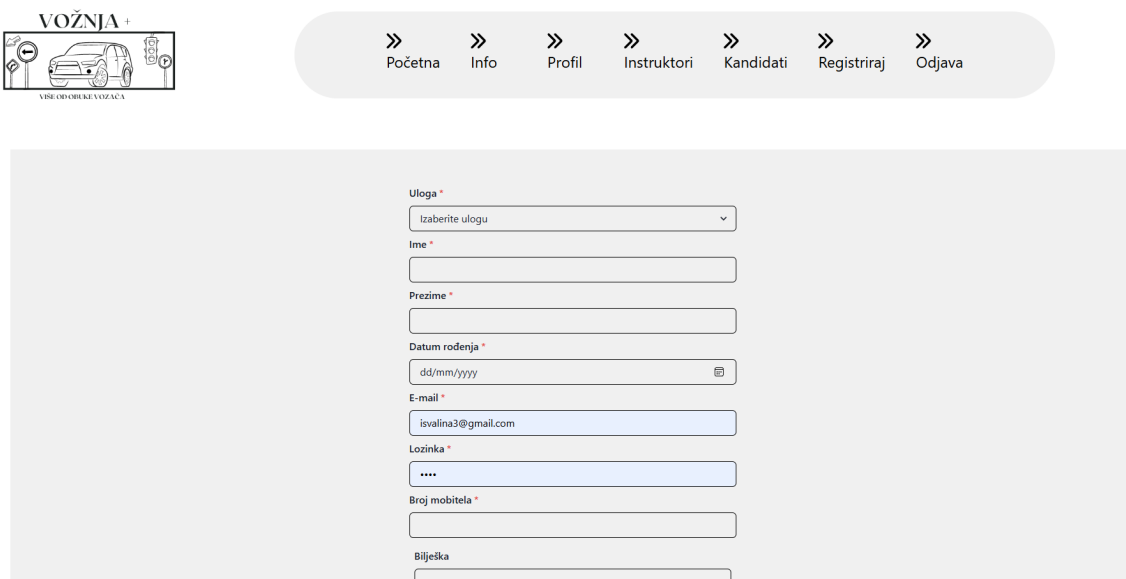
Sučelja administratora autoškole

Nakon što se administrator prijavi u aplikaciju, pored osnovnih, omogućen mu je pristup dodatnim web stranicama. Na slici 4.14. možemo vidjeti kako izgleda stranica koja prikazuje sve instruktore. Spomenutoj stranici može pristupiti samo administrator. Klikom na određenog instruktora prikazuje se stranica s kalendarom odabranog korisnika.



Slika 4.14: Prikaz instruktora

Na kraju, na slikama 4.15 i 4.16. možemo vidjeti prikaz forme za registraciju novih korisnika kojoj također može pristupiti isključivo korisnik s ulogom administratora. Na slici 4.16 vidi se i univerzalno podnožje svake stranice na kojem se mogu naći kontakt podaci.




Slika 4.15: Registracija novog korisnika

Korisnička fotografija *

Choose File No file chosen

Registriraj korisnika

VOŽNJA +



VIŠE OD OBUKE VOZAČA

Kontaktirajte nas!

>> 123 Anywhere St., Any City, ST 12345

✉ hello@reallygreatsite.com

☎ +123-456-7890

Slika 4.16: Registracija novog korisnika i kontakt podaci

Zaključak

U završnom radu izrađena je web aplikacija "Vožnja +" koja služi za organizaciju i kontinuirano praćenje obuke za vozača B kategorije. Glavni cilj rada bio je cjeloviti razvoj aplikacije koji se može podijeliti na tri glavna dijela: početak u kojem je dominantno bilo upoznavanje s novim tehnologijama s naglaskom na React, sredina u kojoj se primarno radilo na dizajnu sustava i odabirom arhitekture te implementacijski dio koji predstavlja najizazovniji dio cijelog projekta.

Uspješnoj realizaciji implementacijskog dijela znatno su doprinijeli kolegiji "Baze podataka", "Objektno orijentirano programiranje", "Razvoj programske potpore za web", ali i široka programerska zajednica koja je ključna za dijeljenje znanja i resursa, ali motivacije i podrške. Najvažnije funkcionalnosti aplikacije su praćenje obuke kandidata bilježenjem svakog održanog sata, ali i kalendar koji ima mogućnost sinkronizacije između instruktora i kandidata što doprinosi boljoj organizaciji obuke, ali i praćenju radnog opterećenja instruktora. Osnovne funkcionalnosti aplikacije obogaćene su integracijom servisa Cloudinary i EmailJS koje omogućuju rad s multimedijским sadržajem i komunikaciju elektroničkom poštom.

Aplikacija ima potencijal za brojna unapređenja od kojih se posebno izdvaja integracija s postojećom aplikacijom za savladavanje teorijskog dijela obuke. Riječ je o aplikaciji "Autoškola" koja služi za treniranje kandidata za polaganje ispita iz prometnih propisa. Slanje automatskih podsjetnika za termine u kalendaru putem elektroničke pošte također bi bilo značajno poboljšanje. Ako bi se sustav koristio u praksi, aplikacija ima mogućnost unapređenja sigurnosti sustava. Osim toga, postoji i mjesto za napredak u izgledu aplikacije i poboljšanju arhitekture cjelokupne aplikacije.

Literatura

1. Astah Community, *Astah Community*, (n.d.). Poveznica: <http://astah.net/editions/uml-new>; pristupljeno u lipnju 2024.
2. Chakra UI, *Getting Started with Chakra UI*, (n.d.). Poveznica: <https://v2.chakra-ui.com/getting-started>; pristupljeno u svibnju 2024.
3. EmailJS, *EmailJS ReactJS*, (n.d.). Poveznica: <https://www.emailjs.com/docs/examples/reactjs/>; pristupljeno u svibnju 2024.
4. JSON Web Tokens (JWT), *Introduction to JSON Web Tokens*, (n.d.). Poveznica: <https://jwt.io/>; pristupljeno u svibnju 2024.
5. Nastava iz predmeta Upravljanje vozilom, *Instruktor vožnje*, (n.d.). Poveznica: https://www.instruktor-voznje.com.hr/nastava_iz_predmeta_upravljanje_vozilom/; pristupljeno u travnju 2024.
6. Spring Boot Project, *Spring Boot Project*, (n.d.). Poveznica: <https://spring.io/projects/spring-boot>; pristupljeno u svibnju 2024.
7. Spring Boot H2 Database, *Spring Boot H2 Database*, (n.d.). Poveznica: <https://www.baeldung.com/spring-boot-h2-database>; pristupljeno u svibnju 2024.
8. Programsko inženjerstvo, *FER ZEMRIS*, (n.d.). Poveznica: <https://www.fer.unizg.hr/predmet/proinz>; pristupljeno u svibnju 2024..
9. React Big Calendar, *React Big Calendar*, (n.d.). Poveznica: <https://github.com/jquense/react-big-calendar>; pristupljeno u lipnju 2024..
10. React Documentation (Legacy), *React Documentation (Legacy)*, (n.d.). Poveznica: <https://legacy.reactjs.org/>; pristupljeno u lipnju 2024.

Sažetak

Ovaj završni rad sastoji se od web aplikacije "Vožnja +" i priložene dokumentacije.

Izrada web aplikacije za autoškole realizirana je kroz tri faze: definiranje korisničkih zahtjeva i stjecanje kompetencija, dizajniranje sustava i odabir arhitekture aplikacije te implementacija u razvojnim okvirima Spring Boot i React. Glavne funkcionalnosti aplikacije su organiziranje obuke pomoću kalendara i praćenje napretka pomoću unaprijed definirane strukture obuke i bilješki koje se unose za održane sate obuke. Cloudinary servis za manipulaciju s medijskim sadržajem integriran je u backend dio aplikacije, dok je slanje elektroničke pošte omogućeno pomoću servisa EmailJS u frontend dijelu aplikacije. Sloj baze podataka implementiran je pomoću H2 relacijske baze podataka. Svaki korak detaljno je dokumentiran te je prikazan konačni izgled aplikacije.

Rezultat rada je učinkovit organizacijski sustav koji olakšava instruktorima i polaznicima autoškole organizaciju i praćenje napretka obuke za vozača B kategorije. Aplikacija ima potencijal s brojna unapređenja od koji se posebno izdvaja integracija različitog edukacijskog sadržaja za samostalno učenje poput postojeće aplikacije "Autoškola" koja služi za savladavanje teorijskog dijela obuke. Također, moguće je proširiti funkcionalnosti davanjem obuke za ostale kategorije vozila. Iako je aplikacija usko specijalizirana na autoškole, moguće ju je razviti u općeniti sustav za organizaciju i praćenje napretka tijekom učenja.

Ključne riječi: autoškola, Spring Boot, React, H2, Cloudinary, EmailJS

Summary

This thesis consists of the "Vožnja +" web application and the accompanying documentation.

The development of the web application for driving schools was realized through three phases: defining user requirements and acquiring competencies, designing the system and selecting the application architecture, and implementation within the Spring Boot and React frameworks. The main functionalities of the application are organizing training sessions using a calendar and tracking progress through a predefined training structure and notes entered for each session. The Cloudinary service for media content manipulation is integrated into the backend part of the application, while sending emails is enabled through the EmailJS service in the frontend part of the application. The database layer is implemented using the H2 relational database. Each step is thoroughly documented, and the final appearance of the application is presented.

The result of the work is an efficient organizational system that facilitates instructors and driving school students in organizing and tracking the progress of driver training for category B. The application has the potential for numerous enhancements, particularly the integration of various educational content for self-learning, similar to the existing "Autoškola" application used for mastering the theoretical part of the training. Additionally, it is possible to expand its functionalities by including training for other vehicle categories. Although the application is specifically tailored for driving schools, it can be developed into a general system for organizing and tracking progress during learning.

Keywords: driving school, Spring Boot, React, H2, Cloudinary, EmailJS