

Programsko inženjerstvo
Ak. god. 2021./2022.

Tectonic HR
Dokumentacija, Rev. 2

Grupa: *BitByBit*
Voditelj: *Maja Jurić*

Datum predaje: 14.1.2022.

Nastavnik: *Daria Primorac*

Sadržaj

1 Dnevnik promjena dokumentacije	3
2 Opis projektnog zadatka	5
2.1 Vrste korisnika	7
2.2 Usporedba s već postojećim rješenjima	8
2.3 Mogućnosti nadogradnje	9
3 Specifikacija programske potpore	10
3.1 Funkcionalni zahtjevi	10
3.1.1 Obrasci uporabe	12
3.1.2 Sekvencijski dijagrami	26
3.2 Ostali zahtjevi	29
4 Arhitektura i dizajn sustava	30
4.1 Baza podataka	31
4.1.1 Opis tablica	31
4.1.2 Dijagram baze podataka	34
4.2 Dijagram razreda	35
4.3 Dijagram stanja	38
4.4 Dijagram aktivnosti	41
4.5 Dijagram komponenti	43
5 Implementacija i korisničko sučelje	44
5.1 Korištene tehnologije i alati	44
5.2 Ispitivanje programskog rješenja	46
5.2.1 Ispitivanje komponenti	46
5.2.2 Ispitivanje sustava	49
5.3 Dijagram razmještaja	58
5.4 Upute za puštanje u pogon	59
5.4.1 Lokalno pokretanje	59
5.4.2 Puštanje u pogon na udaljenom serveru	59

6 Zaključak i budući rad	60
Popis literature	62
Indeks slika i dijagrama	63
Dodatak: Prikaz aktivnosti grupe	64

1. Dnevnik promjena dokumentacije

Rev.	Opis promjene/dodataka	Autori
0.1	Napravljen predložak.	M.Jurić
0.2	Dodani funkcionalni zahtjevi anonimnog korisnika	A.Engler
0.5	Dodane vrste korisnika i njihove uloge u opisu projektnog zadatka	M.Jurić
0.6	Dodani funkcionalni zahtjevi administratora	D.Čemeljić
0.8	Dodan opis projektnog zadatka	K.Iličić
0.9	Dodani funkcionalni zahtjevi znanstvenika (seizmologa)	M.Ćurković
0.10	Dodani ostali zahtjevi	D.Čemeljić
0.11	Dodani opis	K.Iličić
0.12	Dodani neki obrasci uporabe	A.Engler
0.13	Promijenjeni dionici	M.Ćurković
0.14	Dodani još neki obrasci uporabe	K.Iličić
0.15	Dodan ostatak obrazaca uporabe	M.Jurić
0.16	Dodani dijagrami obrazaca uporabe	A.Engler, K.Iličić, M.J.
0.17	Dodani sekvencijski dijagrami	K.Iličić, M.Jurić
0.18	Arhitektura sustava, baza podataka i dijagram razreda	D.Čemeljić, F.Zekan
1.0	Ispravljanje pogrešaka u dokumentaciji	M.Jurić
1.3	Dodana dva obrasca uporabe i ispravljene greške	A.Engler, K.Iličić, M.J.
1.4	Dodana literatura	M.Jurić
1.5	Dodan prvi dio zaključka	M.Jurić

Nastavljeno od prethodne stranice

Rev.	Opis promjene/dodatka	Autori
7.1.6	Dodan dijagram komponenti i razmještaja	D.Čemeljić, A.Engler
1.7	Dodan dijagram aktivnosti	M.Jurić
1.8	Dodan dijagram stanja	K.Iličić
1.9	Dodane upute za puštanje u pogon	D.Čemeljić, F.Zekan
2.0	Predana druga verzija	

2. Opis projektnog zadatka

Ovaj se projekt bavi razvojem programske podrške za web aplikaciju "TectonicHR". Cilj aplikacije "TectonicHR" olakšano je prikupljanje podataka o intenzitetu potres te olakšani vizualni pristup informacijama. Aplikacija je namijenjena znanstvenoj zajednici, ali i općoj populaciji.

Znanstvenoj zajednici, seismologima, bit će olakšan pristup informacijama i njihovo prikupljanje. Opća populacija, građani, imat će mogućnost unosa novog potresa kojega su osjetili te pregled već zabilježenih potresa.

Potrese smo podijelili na aktualne i arhivirane potrese. Aktualni potresi su oni potresi za koje građani još mogu ispuniti upitnik, a arhivirani potresi su svi oni za koje to nije moguće.

Prilikom otvaranja aplikacije građanima se prikazuje početna stranica na kojoj se nude tri opcije:

- Novi upitnik?
- Aktualni potresi
- Arhivirani potresi

Na početnoj stranici također se nalazi i preliminarna karta Hrvatske na kojoj su označeni aktualni i arhivirani potresi. Boje oznaka potresa različite su za ove dvije kategorije potresa.

Odabirom opcije „Aktualni potresi“ prikazuje se stranica na kojoj se može pregledati karta aktualnih potresa, mogu se pregledati podaci o aktualnim potresima te se može ispuniti upitnik za neki od navedenih aktualnih potresa.

Odabirom opcije „Arhivirani potresi“ prikazuje se stranica na kojoj se može pregledati karta intenziteta arhiviranih potresa i na kojoj se mogu pregledati podaci o arhiviranim potresima.

Na kartama na stranicama „Aktualni potresi“ i „Arhivirani potresi“ zvjezdicom je označen epicentar te kružićem određene boje označen je intenzitet potresa. Hladnije plave nijanse označavaju slabiji intezitet, a tamnije crvene jači intezitet. Pored svake od karta nalazi se tablica s podacima o zadnjem i starijim potresima.

Pri pregledu arhiviranih i aktualnih potresa moguće je filtrirati potrese prema mjestu, vremenu ili intezitetu.

Ako se posljednji prijavljeni potres ne poklapa s opažanjima građana, preko treće opcije „Novi potres?” građanin može prijaviti novi potres koji je osjetio. Građanin prijavljuje novi potres ispunjavanjem i predajom upitnika.

Upitnik se sastoji od pitanja iz kojih znanstvenici mogu dobiti vrijedne informacije o tome kakvi su bili učinci potresa. Pitanja se odnose na to koliko se potres osjetio, koliku je štetu napravio na malim predmetima, kućama, zgradama i zemlji te kako su ljudi reagirali. Odgovori na ta pitanja pomažu pri računanju intenziteta potresa.

Na početnoj stranici u gornjem desnom kutu nalazi se gumb za prijavu u sustav. Tu funkcionalnost koriste seismolozi koji prijavom mogu preuzeti podatke o potresu i odgovore građana na upitnik. Pri prijavi upisuju svoje e-mail i lozinku.

Znanstvenike u sustav mora registrirati administrator. Pri registraciji, administrator upisuje:

- e-mail seismologa
- lozinku.

Nakon što administrator registrira seismologa u sustav, seismolog će e-mailom primiti obavijest o registraciji te time dobiva mogućnost prijave u sustav. Pregled svih registriranih seismologa omogućen je samo administratoru. Administrator se također mora prijaviti pri dolasku na stranicu kako bi imao sve ovlasti.

Program treba sam računati intenzitet potresa pomoću prethodno preuzetih upitnika. Prema odgovorima iz upitnika treba se automatski izračunati intenzitet potresa. Vrijednost intenziteta na pojedinoj lokaciji odgovara srednjoj vrijednosti intenziteta svih upitnika ispunjenih za potres na istoj lokaciji. Mjesto epicentra aproksimira se koristeći lokacije jednakog intenziteta, a intenzitet potresa se u epicentru (predstavljen bojom) određuje koristeći *Koevesligethyjevu jednadžbu*:

$$I_0 = I_{max} + 3 \log \frac{r}{h} + 3\mu\alpha(r - h) \quad (2.1)$$

- I_{max} - procijenjeni intenzitet potresa na udaljenosti r od hipocentra
- $h = 10 \text{ km}$
- $\mu = 0.4343$
- $\alpha = 0.005 \text{ km}^{-1}$

2.1 Vrste korisnika

Postoje tri vrste korisnika, a to su:

- anonimni korisnik (građanin)
- znanstvenik (seizmolog)
- administrator

Neregistriranom (anonimnom) korisniku otvaranjem aplikacije prikazuje se izbornik u kojem može odabratи želi li pregledati aktualne potrese („Aktualni potresi“), pregledati arhivirane potrese („Arhivirani potresi“) ili ispuniti upitnik ako je osjetio novi potres („Novi potres?“). Klikom na „Aktualni potresi“ prikazuju mu se karta i popis potresa koje administrator još nije arhivirao. Anonimni korisnik može pretraživati te potrese i odabratи jedan potres te za njega ispuniti upitnik. Na početnoj stranici, klikom na „Arhivirani potresi“ pokaže mu se karta i ispod nje popis potresa koje je administrator arhivirao. Prelaženjem kurzorom preko oznake kojom je označen taj potres, prikazuju se osnovne informacije o potresu (datum, vrijeme, lokacija i intenzitet potresa).

Seizmolog (znanstvenik) se prijavljuje e-mailom i lozinkom. Ima sve mogućnosti kao i anonimni korisnik (ispunjavanje upitnika za novi potres, pregled karte i drugih podataka o aktualnim i arhiviranim potresima) uz još dodatnu ovlast preuzimanja podataka o potresima u .csv formatu. Seizmolog na svom profilu može promjeniti svoje podatke i lozinku.

Administrator ima najveće ovlasti. Početna stranica izgleda mu isto kao i seizmologu, ali mu se na stranici profila ispod njegovih podataka nalaze i dva gumba, „Novi upitnici“ i „Dodaj novog seizmologa“. Klikom na „Novi upitnici“ pregledava ispunjene upitnike koje još nije svrstao u nijedan potres. Upitnike može pridijeliti nekom već imenovanom potresu ili može stvoriti, imenovati i potvrditi novi potres te ih pridijeliti tom novostvorenom potresu. Klikom na „Dodaj novog seizmologa“ otvara mu se obrazac za registraciju novog seizmologa. Mora upisati ime, prezime, e-mail i lozinku novog seizmologa. Administrator, kao i seizmolog, može promjeniti svoje podatke. S početne stranice može pristupiti stranici aktualnih potresa. Na toj stranici može odabratи jedan ili više potres te ih arhivirati. Na stranici arhiviranih i na stranici aktualnih potresa može pregledavati podatke o potresima te također, kao i seizmolog, preuzeti podatke .csv formatu.

Sustav treba podržavati rad više korisnika u stvarnom vremenu.

2.2 Usporedba s već postojećim rješenjima

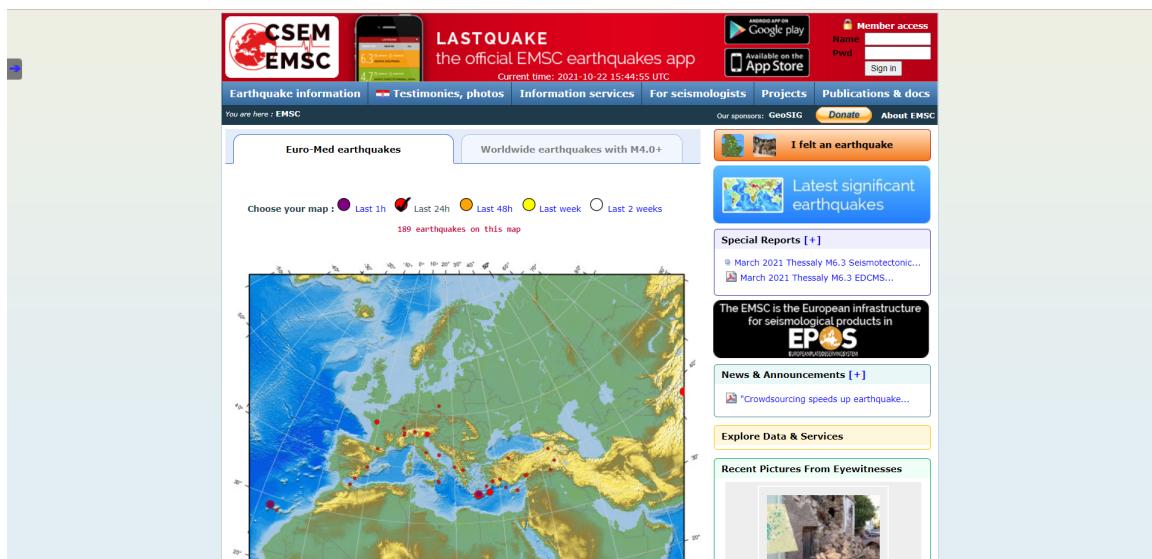
Od sličnih aplikacija koje već postoje, najpoznatija je EMSC. Aplikaciju je razvila organizacija European Mediterranean Seismological Centre.

Početna stranica sastoji se od karte Europe i Mediterana. Na karti možete birati želite li pregledati potrese u zadnjih sat vremena, 24 sata, 48 sati, tjedan dana ili dva tjedna. Osim karte Europe i Mediterana, postoji mogućnost otvaranja karte cijelog svijeta. Ispod karte nalazi se tablica s informacijama o potresima koji su označeni na karti.

Osim toga, stranica nudi funkcionalnosti ispunjavanja upitnika o doživljaju potresa i dodavanje slika koje prikazuju posljedice potresa.

Postoji posebni odjeljak za seizmologe, kao i odjeljci namijenjeni projektima organizacije i publikaciji.

Problem stranice je što je prilično neintuitivna za korištenje i zastarjelog dizajna. Nudi mnogo mogućnosti u raznim izbornicima te to stvara problem nesnalaženja. Većinu građana koji posjećuju tu stranicu radi prijave potresa ili traženja informacija o potresu kojega su možda osjetili zasigurno neće zanimati projekti EMSC-a ili znanstvena publikacija.



Slika 2.1: Slika 2.2: početna stranica EMSC-a

Prednost aplikacije „TectonicHR“ bila bi preglednost i mogućnost lakšeg snalaženja na karti. Mogućnost prijave seizmologa na stranicu omogućilo bi bolju prilagodbu stranice. Građanima pri korištenju ne bi smetali izbornici s mogućnostima koje

oni ne bi koristili, već bi im se pregledno prikazivale funkcionalnosti ispunjavanja upitnika te pregleda aktualnih i starijih potresa.

Interes za korištenjem aplikacije imat će i građani i znanstvena zajednica. Potresi uvelike utječu na psihičko stanje ljudi. Kada dođe do potresa, većina želi znati gdje je epicentar, kolike je magnitude potres bio, kakvu je štetu prouzročilo u ostalim dijelovima pogodjenog područja... Tako da bi ova jednostavna aplikacija ljudima pružila, za početak, osnovne informacije o doživljenom potresu te odgovor na neka od pitanja koja im se nameću nakon doživljenog potresa. Osim toga, korist od aplikacije imaju i znanstvenici koji putem opažanja građana mogu doći do vrlo vrijednih informacija.

2.3 Mogućnosti nadogradnje

Buduće verzije aplikacije mogle bi uključivati mogućnosti:

1. umetanje slika područja pogodjenih potresom
2. pregledavanje pojedinog potresa (informacija o potresu, galerije slika...)
3. prikaz područja s većim rizikom pojave potresa
4. donacije za obnovu područja pogodjenih potresom
5. preuzimanje karte s označenim potresima u tom trenutku
6. pregled potresa iz određenog vremenskog perioda ili s geografskog područja

Vidimo veliku korist i primjenjivost aplikacije te bi se nove funkcionalnosti dodavale nakon dobivanja povratne informacije od svih korisnika (i građana i seizmologa) u ovisnosti o njihovim potrebama.

3. Specifikacija programske potpore

3.1 Funkcionalni zahtjevi

Dionici:

1. Administrator
2. Seizmoloz
3. Građani (neregistrirani korisnici)
4. Razvojni tim
5. Geološki institut
6. Geofizički institut

Aktori i njihovi funkcionalni zahtjevi:

1. Neregistrirani/neprijavljeni korisnik (građanin) (inicijator) može:
 - (a) ispuniti upitnik ako je osjetio novi potres
 - (b) pregledati sve arhivirane potrese s:
 - i. interaktivnom kartom Hrvatske
 - ii. dodatnim informacijama (broj ispunjenih upitnika, datum, vrijeme, geografski položaj, dubina fokusa, magnituda i naziv područja)
 - (c) pristupiti aktualnim potresima i:
 - i. ispuniti upitnik
 - ii. pregledati već analizirane podatke (karta, datum, vrijeme, itd.)
2. Seismolog/znanstvenik (inicijator) može:
 - (a) sve što i neregistrirani/neprijavljeni korisnik može
 - (b) pristupiti aplikaciji nakon prijave (e-mail i lozinka)
 - (c) pregledavati podatke o:
 - i. arhiviranim potresima
 - ii. aktualnim potresima
 - (d) u tekstualnom formatu preuzeti podatke o potresima

(e) izmijeniti svoje podatke

3. Administrator (inicijator) može:

- (a) sve što i seismolog
- (b) upravljati upitnicima
 - i. stvoriti novi potres od jednog ili više upitnika
 - ii. dodati upitnik nekom od postojećih potresa
 - iii. brisati upitnike
- (c) upravljati seismolozima
 - i. registrirati nove seismologe
 - ii. pregledati registrirane seismologe
 - iii. ukloniti registrirane seismologe
- (d) izmijeniti svoje podatke

4. Baza podataka (sudionik):

- (a) pohranjuje sve podatke o potresima
- (b) pohranjuje sve podatke o registriranim korisnicima (seismolozima i administratoru)
- (c) pohranjuje sve odgovore na upitnik

3.1.1 Obrasci uporabe

Opis obrazaca uporabe

UC1 - Prijava

- **Glavni sudionik:** Korisnik (seizmolog, administrator)
- **Cilj:** Dobiti pristup odgovarajućem sučelju na temelju uloge i ovlastima
- **Sudionici:** Baza podataka
- **Preduvjet:** Registracija
- **Opis osnovnog tijeka:**
 1. Korisnik odabire opciju za prijavu
 2. Pokazuje se obrazac za prijavu
 3. Korisnik unosi korisničke podatke (e-mail, lozinka)
 4. Provjera ispravnosti unesenih podataka
 5. Pristup odgovarajućim funkcijama
- **Opis mogućih odstupanja:**
 - 4.a Neispravan email ili lozinka
 1. Sustav obavještava korisnika o netočnim informacijama
 2. Sustav vraća korisnika na obrazac za prijavu

UC2 - Registracija novih seizmologa

- **Glavni sudionik:** Administrator
- **Cilj:** Registrirati nove seizmologe
- **Sudionici:** Seizmolog, Baza podataka
- **Preduvjet:** Administrator prijavljen
- **Opis osnovnog tijeka:**
 1. Administrator na svom profilu odabire opciju "Novi korisnik"
 2. Pokazuje se obrazac za registraciju
 3. Administrator unosi korisničke podatke (e-mail, lozinka)
 4. Administrator potvrđuje unos
 5. Baza podataka se ažurira
- **Opis mogućih odstupanja:**
 - 3.a Unos neispravnog e-maila
 1. Sustav obavještava administratora o neispravnom upisu e-maila i vraća ga na stranicu za registraciju
 2. Administrator mijenja potrebne podatke te završava unos ili odustaje od registracije

3.b Unos e-maila koji već postoji u sustavu

1. Sustav obaveštava administratora o upisu e-maila seizmologa koji je već registriran u sustavu i vraća ga na stranicu za registraciju
2. Administrator mijenja potrebne podatke te završava unos ili odustaje od registracije

3.c Unos lozinke koja ne odgovara ograničenjima

1. Sustav obaveštava administratora o upisu nevaljane lozinku i ostaje na stranici za registraciju
2. Administrator mijenja potrebne podatke te završava unos ili odustaje od registracije

UC3 - Pregled korisničkih podataka

- **Glavni sudionik:** Seizmolog, Administrator
- **Cilj:** Pregled podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik u kutu početne stranice odabire opciju za pregled profila
 2. Korisnik pregledava podatke

UC4 - Promjena korisničkih podataka

- **Glavni sudionik:** Administrator, Seizmolog
- **Cilj:** Promjena podataka
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav i odabrao opciju za pregled profila
- **Opis osnovnog tijeka:**
 1. Pregled profila
 2. Odabir opcije za promjenu podataka
 3. Korisnik mijenja željene podatke
 4. Korisnik potvrđuje promjenu
 5. Ažurira se baza podataka
- **Opis mogućih odstupanja:**
 - 3.a Unos nevaljane lozinke
 1. Sustav obaveštava korisnika o upisu nevaljane lozinke i vraća ga na stranicu za promjenu podataka

2. Korisnik mijenja potrebne podatke te završava unos ili odustaje od promjene podataka

UC5 - Pregled seizmologa

- **Glavni sudionik:** Administrator
- **Cilj:** Pregled seizmologa koji su registrirani u sustavu
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Administrator u izborniku na početnoj stranici odabire opciju za pregled registriranih korisnika
 2. Sustav prikazuje popis registriranih korisnika (seizmologa)

UC6 - Uklanjanje seizmologa

- **Glavni sudionik:** Administrator
- **Cilj:** Uklanjanje registriranih seismologa iz sustava
- **Sudionici:** Baza podataka, Seismolog
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje barem jednog registriranog seismologa
- **Opis osnovnog tijeka:**
 1. Pregled seismologa
 2. Odabir opcije "Ukloni" pored imena seismologa
 3. Prikazuje se potvrđni okvir
 4. Administrator potvrđuje uklanjanje
 5. Ažuriranje baze podataka
- **Opis mogućih odstupanja:**
 - 3.a Administrator odustaje od uklanjanja seismologa
 1. Sustav vraća administratora na pregled seismologa

UC7 - Pregled upitnika

- **Glavni sudionik:** Administrator
- **Cilj:** Pregled pristiglih upitnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Administrator prijavljen u sustav

- **Opis osnovnog tijeka:**
 1. Administrator u izborniku na početnoj stranici odabire opciju za pregled potresa
 2. Sustav prikazuje popis pristiglih upitnika
- **Opis mogućih odstupanja:**
 - 2.a Nema novih upitnika
 1. Prikazuje se poruka da nema upitnika

UC8 - Registracija novog potresa

- **Glavni sudionik:** Administrator
- **Cilj:** Stvoriti novi aktualni potres
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje upitnika
- **Opis osnovnog tijeka:**
 1. Pregled pristiglih upitnika
 2. Odabir jednog ili više upitnika
 3. Odabir „Novi potres“ u izborniku
 4. Otvara se obrazac za dodavanje novog potresa
 5. Administrator unosi naziv potresa
 6. Administrator odabire opciju "Potvrди" čime potvrđuje unos novog potresa
 7. Ažurira se baza podataka
 8. Sustav obavještava administratora o uspješnosti dodavanja novog potresa

UC9 - Dodavanje upitnika postojićem aktualnom potresu

- **Glavni sudionik:** Administrator
- **Cilj:** Srvstavanje upitnika već postojećim potresima u svrhu upotpunjavanja informacija
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje upitnika

- Postojanje aktualnog potresa
- **Opis osnovnog tijeka:**
 1. Pregled pristiglih upitnika
 2. Odabir jednog ili više upitnika
 3. Odabir "Aktualni potres"
 4. Odabir nekog imenovanog potresa iz izbornika
 5. Baza podataka se ažurira
 6. Sustav obavještava administratora o uspješnosti dodavanja upitnika pos-tojećem potresu

UC10 - Brisanje upitnika

- **Glavni sudionik:** Administrator
- **Cilj:** Ukloniti nerelevantne upitnike
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje upitnika
- **Opis osnovnog tijeka:**
 1. Administrator u izborniku na početnoj stranici odabire opciju za pregled profila
 2. Administrator odabire upitnike za brisanje
 3. Administrator na profilu odabire opciju brisanja upitnika
 4. Prikazuje se potvrđni okvir
 5. Administrator potvrđuje uklanjanje
 6. Ažuriranje baze podataka
- **Opis mogućih odstupanja:**
 - 5.a Administrator odabire opciju "Odustani"
 1. Sustav vraća administratora na pregled upitnika

UC11 - Pregled aktualnih potresa

- **Glavni sudionik:** Administrator
- **Cilj:** Pregledati aktualne potrese
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**
 1. Na početnoj stranici odabir opcije „Aktualni potresi“

2. Prilikom učitavanja stranice prikazana je preliminarna karta intenziteta i popis aktualnih potresa
- **Opis mogućih odstupanja:**
 - 2.a Nema aktualnih potresa
 1. Prikazuje se odgovarajuća poruka

UC12 - Arhiviranje aktualnih potresa

- **Glavni sudionik:** Administrator
- **Cilj:** Prebaciti aktualne potrese u kategoriju arhiviranih
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje aktualnih potresa
- **Opis osnovnog tijeka:**
 1. Pregled aktualnih potresa
 2. Odabir jednog ili više aktualnih potresa
 3. Odabir „Arhiviraj“
 4. Baza podataka se ažurira
 5. Sustav obavještava korisnika o uspješnosti arhiviranja potresa

UC13 - Odjava iz sustava

- **Glavni sudionik:** Seizmolog, Administrator
- **Cilj:** Odjaviti se iz sustava
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Odabir opcije za odjavu
 2. Korisnik se odjavljuje iz sustava
 3. Prikazuje se početna stranica za neregistriranog korisnika

UC14 - Preuzimanje podataka o arhiviranim potresima

- **Glavni sudionik:** Seizmolog, Administrator
- **Cilj:** Preuzimanje podataka o potresima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav

- **Opis osnovnog tijeka:**
 1. Pregled i pretraživanje/filtriranje arhiviranih potresa
 2. Odabir željenog potresa
 3. Odabir opcije „Preuzmi“
 4. Preuzimanje podataka

UC15 - Preuzimanje podataka o aktualnim potresima

- **Glavni sudionik:** Seizmolog, Administrator
- **Cilj:** Preuzimanje podataka o potresima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Pregled i pretraživanje/filtriranje arhiviranih potresa
 2. Odabir željenog potresa
 3. Odabir opcije „Preuzmi“
 4. Preuzimanje podataka

UC16 - Pregled arhiviranih potresa

- **Glavni sudionik:** Korisnik (neregistrirani građanih, seismolog, administrator)
- **Cilj:** Pregledati arhivirane potrese
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**
 1. Na početnoj stranici odabir opcije „Arhivirani potresi“
 2. Prilikom učitavanja stranice prikazana je preliminarna karta intenziteta i popis arhiviranih potresa
- **Opis mogućih odstupanja:**
 - 2.a Nema arhiviranih potresa
 1. Prikazuje se odgovarajuća poruka da još nema arhiviranih potresa

UC17 - Pretraživanje/filtriranje arhiviranih potresa

- **Glavni sudionik:** Korisnik (neregistrirani građanin, seismolog, administrator)
- **Cilj:** Pretražiti i filtrirati arhivirane potrese
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**

1. Na početnoj stranici odabir opcije „Arhivirani potresi“
 2. Prilikom učitavanja stranice prikazana je preliminarna karta intenziteta i popis arhiviranih potresa
 3. Korisnik pretražuje ili filtrira po mjestu, vremenu i intenzitetu
 4. Pregled potresa s traženim svojstvima
- **Opis mogućih odstupanja:**
 - 3.a Korisnik pri pretraživanju upisuje podatke koji nisu važeći ni za jedan arhivirani potres
 1. Ne prikazuje ni jedan arhivirani potres na karti ni na popisu
 2. Ispisuje se poruka da nije pronađen ni jedan potres

UC18 - Pretraživanje/filtriranje aktualnih potresa

- **Glavni sudionik:** Korisnik (neregistrirani građanin, seismolog, administrator)
- **Cilj:** Pretražiti i filtrirati aktualne potrese
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**
 1. Na početnoj stranici odabir opcije „Aktualni potresi“
 2. Prilikom učitavanja stranice prikazana je preliminarna karta intenziteta i popis aktualnih potresa
 3. Korisnik pretražuje ili filtrira po mjestu, vremenu i intenzitetu
 4. Pregled potresa s traženim svojstvima
- **Opis mogućih odstupanja:**
 - 3.a Korisnik pri pretraživanju upisuje podatke koji nisu važeći ni za jedan aktualni potres
 1. Ne prikazuje ni jedan aktualni potres na karti ni na popisu
 2. Ispisuje se poruka da nije pronađen ni jedan potres

UC19 - Prijava novog potresa

- **Glavni sudionik:** Korisnik (neregistrirani građanin, seismolog, administrator)
- **Cilj:** Ispuniti upitnik kako bi se prijavio novi potres
- **Sudionici:** Baza podataka
- **Opis osnovnog tijeka:**
 1. Na početnoj stranici odabir opcije „Novi potres?“
 2. Otvorena se upitnik

3. Sustav pita korisnika za dozvolu korištenja korisnikove lokacije
 4. Korisnik dopušta korištenje lokacije
 5. Korisnik ispunjava upitnik
 6. Odabir opcije "Predaj"
 7. Dodaje se novi potres u bazu podataka
 8. Korisnik dobiva povratnu informaciju da je uspješno ispunio upitnik
- **Opis mogućih odstupanja:**
 - 4.a Korisnik ne dozvoljava korištenje lokacije
 1. Korisnik iz padajućeg izbornika bira jednu od ponuđenih lokacija
 - 5.a Korisnik nije odgovorio na obvezno pitanje
 1. Sustav obavještava korisnika o neuspjeloj predaji te ga vraća na stranicu s upitnikom
 - 6.a Odabir opcije "Odustani"
 1. Podaci nisu sačuvani
 2. Sustav vraća korisnika na početnu stranicu

UC20 - Ispunjavanje upitnika za aktualni potres

- **Glavni sudionik:** Korisnik (neregistrirani građanin, seismolog, administrator)
- **Cilj:** Ispuniti upitnik za aktualni potres
- **Sudionici:** Baza podataka
- **Preduvjet:** Postojanje aktualnog potresa
- **Opis osnovnog tijeka:**
 1. Na početnoj stranici odabir opcije „Aktualni potresi“
 2. Pregled aktualnih potresa
 3. Odabir želenog aktualnog potresa
 4. Odabir opcije "Ispuni upitnik"
 5. Otvara se upitnik s popunjениm mjestom, datumom i vremenom odbaranog aktualnog potresa
 6. Korisnik ispunjava ostatak upitnika
 7. Odabir opcije "Predaj"
 8. Dodavanje informacija iz upitnika u bazu podataka
 9. Ažuriranje podataka o aktualnom potresu
 10. Korisnik dobiva povratnu informaciju da je uspješno ispunio upitnik
- **Opis mogućih odstupanja:**
 - 6.a Korisnik nije odgovorio na obvezno pitanje

1. Sustav obavještava korisnika o neuspjeloj predaji te ga vraća na stranicu s upitnikom
- 7.a Odabir opcije "Odustani"
 1. Podaci nisu sačuvani
 2. Sustav vraća korisnika na stranicu aktualnih potresa

UC21 - Brisanje potresa i pripadajućih upitnika

- **Glavni sudionik:** Administrator
- **Cilj:** Brisanje potresa
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje potresa
- **Opis osnovnog tijeka:**
 1. Administrator u izborniku na početnoj stranici odabire opciju za pregled potresa
 2. Administrator odabire opciju "Obriši" za potres koji želi obrisati
 3. Administrator na profilu odabire opciju brisanja potresa
 4. Prikazuje se potvrdni okvir
 5. Administrator označava da želi obrisati i pripadajuće upitnike
 6. Administrator upisuje ime potresa kojeg želi obrisati
 7. Administrator potvrđuje uklanjanje
 8. Ažuriranje baze podataka
- **Opis mogućih odstupanja:**
 - 7.a Administrator odabire opciju "Odustani"
 1. Sustav vraća administratora na pregled potresa

UC22 - Brisanje potresa bez brisanja upitnika

- **Glavni sudionik:** Administrator
- **Cilj:** Brisanje potresa
- **Sudionici:** Baza podataka
- **Preduvjet:**
 - Administrator prijavljen u sustav
 - Postojanje potresa
- **Opis osnovnog tijeka:**

1. Administrator u izborniku na početnoj stranici odabire opciju za pregled potresa
 2. Administrator odabire opciju "Obriši" za potres koji želi obrisati
 3. Administrator na profilu odabire opciju brisanja potresa
 4. Prikazuje se potvrđni okvir
 5. Administrator označava da ne želi obrisati pripadajuće upitnike
 6. Administrator upisuje ime potresa kojeg želi obrisati
 7. Administrator potvrđuje uklanjanje
 8. Administratoru se prikazuju upitnici unutar kojih su i upitnici koji su prije brisanja pripadali potresu
 9. Ažuriranje baze podataka
- **Opis mogućih odstupanja:**
 - 7.a Administrator odabire opciju "Odustani"
 1. Sustav vraća administratora na pregled potresa

UC23 - Preuzimanje pitanja iz upitnika

- **Glavni sudionik:** Korisnik (seismolog, administrator)
- **Cilj:** Preuzeti pitanja koja se javljaju u upitnicima zajedno sa svim mogućim odgovorima i intenzitetima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik u izborniku na početnoj stranici odabire opciju za izvoz podataka
 2. Korisnik odabire opciju "Preuzmi pitanja upitnika"
 3. Preuzimanje podataka u .csv formatu

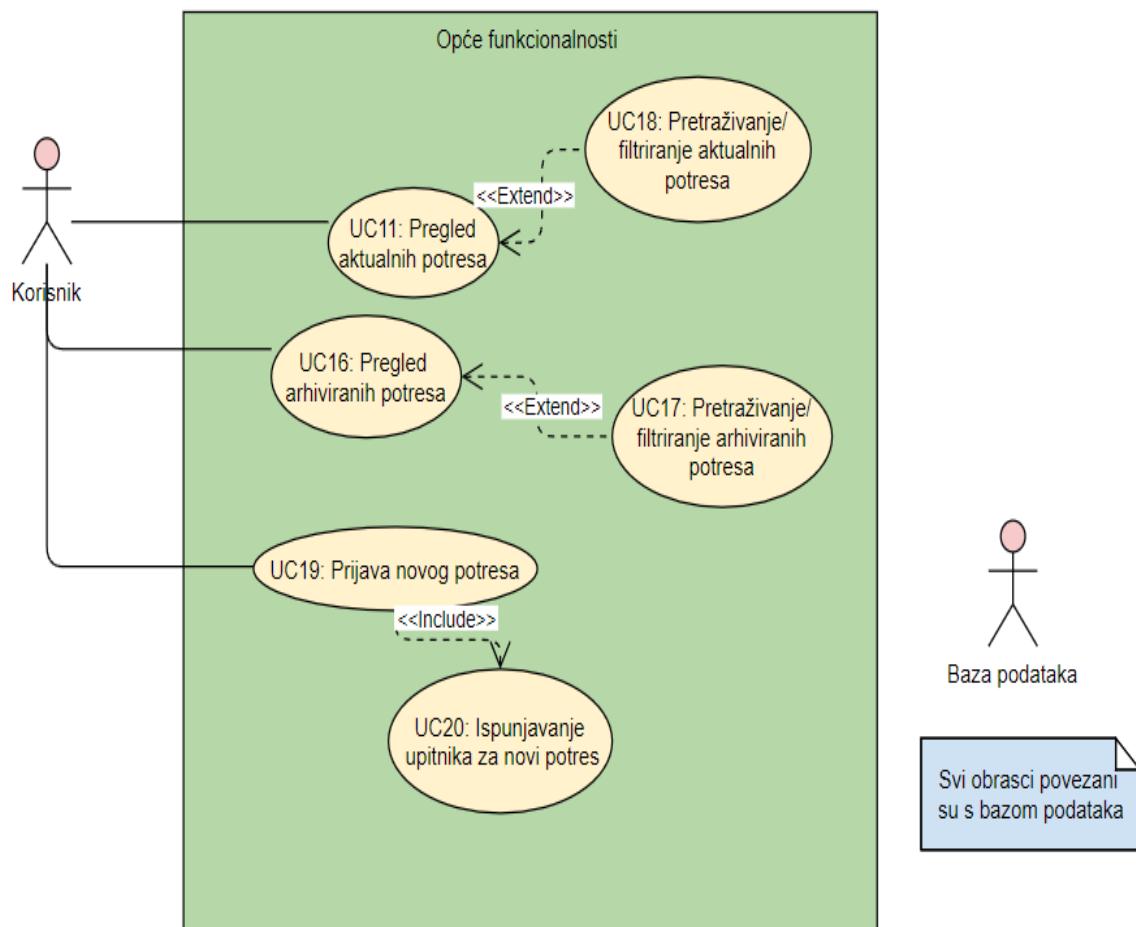
UC24 - Preuzimanje svih potresa s upitnicima

- **Glavni sudionik:** Korisnik (seismolog, administrator)
- **Cilj:** Preuzeti sve potrese zajedno s popunjениm upitnicima
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik u izborniku na početnoj stranici odabire opciju za izvoz podataka
 2. Korisnik odabire opciju "Preuzmi i upitnike"

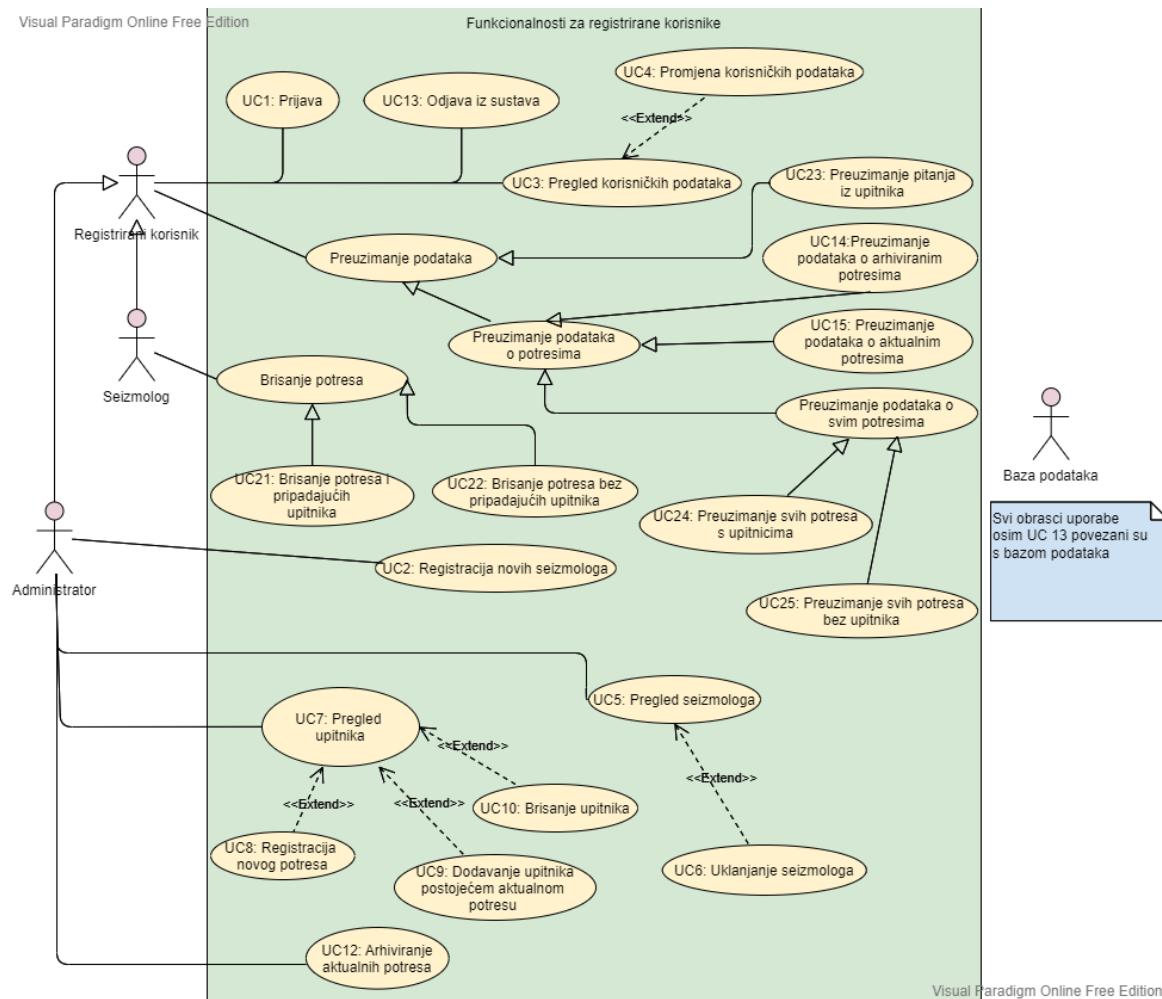
3. Korisnik odabire opciju "Preuzmi potrese"
4. Preuzimanje podataka u .csv formatu

UC25 - Preuzimanje svih potresa bez upitnika

- **Glavni sudionik:** Korisnik (seismolog, administrator)
- **Cilj:** Preuzeti sve potrese bez popunjениh upitnika
- **Sudionici:** Baza podataka
- **Preduvjet:** Korisnik prijavljen u sustav
- **Opis osnovnog tijeka:**
 1. Korisnik u izborniku na početnoj stranici odabire opciju za izvoz podataka
 2. Korisnik odabire opciju "Izuzmi upitnike"
 3. Korisnik odabire opciju "Preuzmi potrese"
 4. Preuzimanje podataka u .csv formatu

Dijagrami obrazaca uporabe

Slika 3.1: Dijagram obrazaca uporabe, funkcionalnosti koje imaju svi korisnici

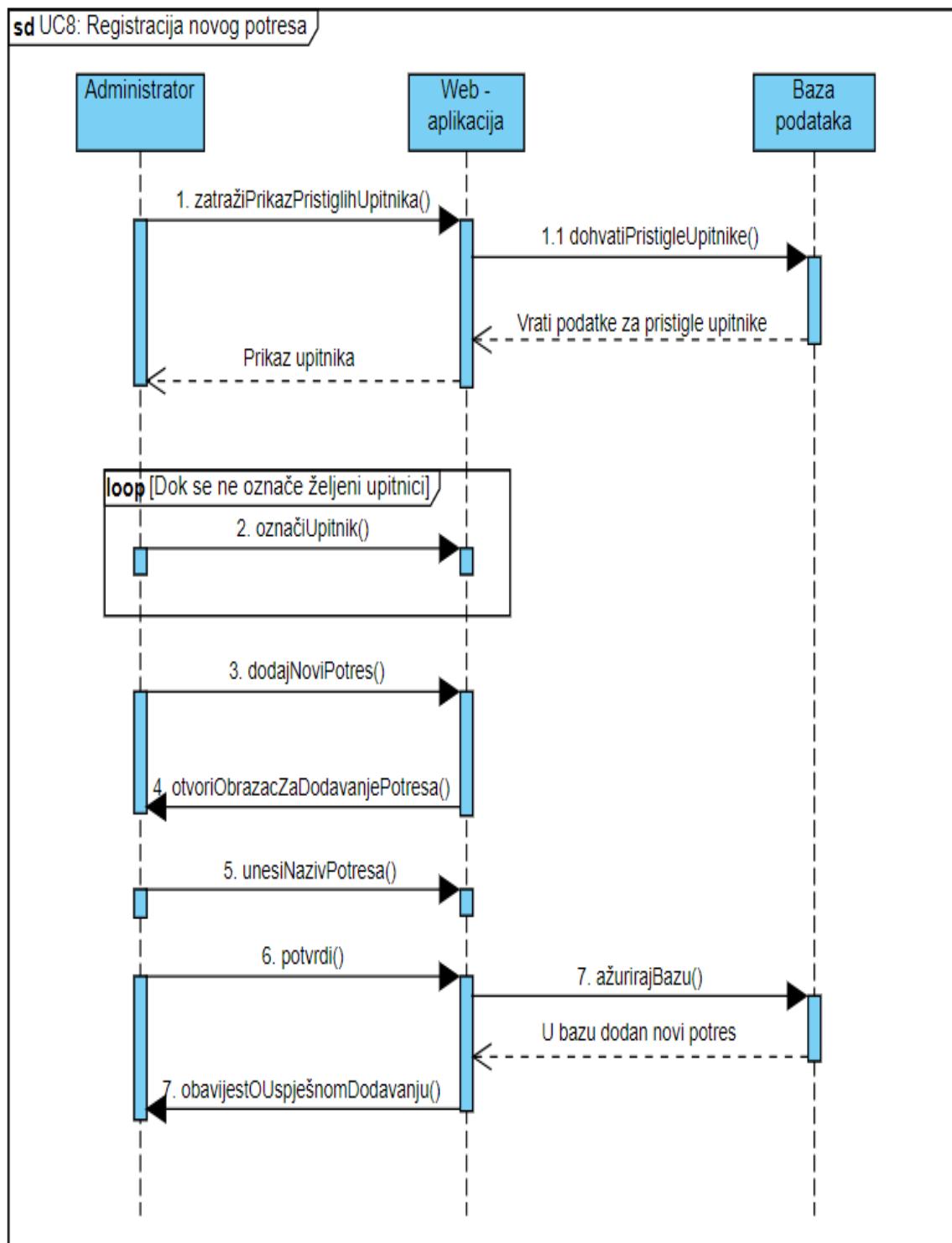


Slika 3.2: Dijagram obrazaca uporabe, funkcionalnosti koje imaju regestrirani korisnici

3.1.2 Sekvencijski dijagrami

UC8: Registracija novog potresa

Administrator odabire opciju "Pristigli upitnici". Poslužitelj potom šalje zahtjev bazi podataka kako bi dohvatio sve pristigle upitnike. Prihvata se zahtjev te se na klijentu prikazuju svi pristigli upitnici. Administrator označuje upitnike te ih na taj način grupira. Kada grupira sve upitnike koji se odnose na isti potres, odabire opciju "Dodaj novi potres". Potom imenuje taj potres i odabire opciju "potvrdi". U bazu podataka se upisuje novi potres. Sustav obavijesti administratora o uspješnom dodavanju novog potresa.

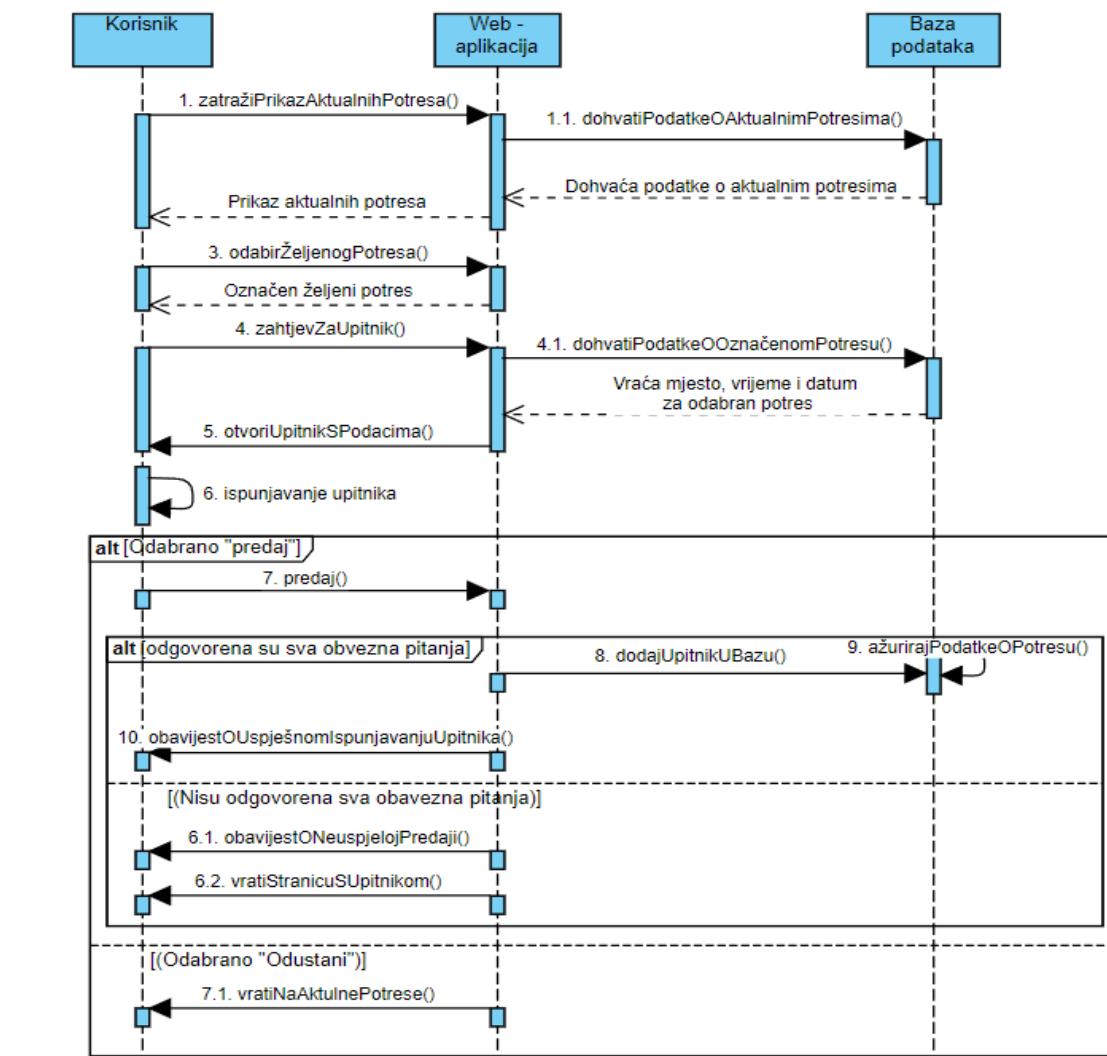


Slika 3.3: sekvenički dijagram za UC8

UC20: Ispunjavanje upitnika za aktualni potres

Korisnik odabire opciju "Aktualni potresi". Poslužitelj potom šalje zahtjev bazi

podataka kako bi dohvatio sve aktualne potrese. Prihvata se zahtjev te se na klijentu prikazuju svi aktualni potresi. Korisnik odabire potres za koji želi ispuniti upitnik. S klijenta se bazi podataka šalje zahtjev za dohvaćanjem podataka o potresu koji se automatski popunjavaju u upitniku. Korisniku se prikazuje upitnik s djelomično popunjanim odgovorima. Korisnik popunjava upitnik. Kada pritisne opciju "Predaj", upitnik se šalje na validaciju. Ako su odgovorena sva pitanja, odgovori se pohranjuju u bazu te se na temelju odgovora računa intezitet potresa pa se i to ažurira u bazi. Kada je upitnik uspješno predan, sustav o tome obavijesti korisnika. Ako nije sve odgovoreno, korisnik dobiva obavijest o neuspjeloj predaji te ga sustav vraća na stranicu s upitnikom. Ako je korisnik odabrao "Odustani" umjesto "Predaj" sustav ga vraća na stranicu "Aktualni potresi".



Slika 3.4: sekvenčni dijagram za UC20

3.2 Ostali zahtjevi

- Aplikacija treba podržavati rad više korisnika u stvarnom vremenu.
- Sustav treba biti implementiran kao web aplikacija koristeći objektno orijentirane jezike.
- Korisničko sučelje mora biti pisano na hrvatskom jeziku i podržavati dijakritičke znakove.
- Upitnik o potresu mora imati barem 10 pitanja pomoću kojih se može odrediti intenzitet potresa.
- Aplikacija mora jednostavno prikazivati informacije o potresima korisnicima.
- Karta treba prikazivati sve gradove i mjesta u kojima su upitnici popunjeni.
- Lozinka administratora i seismologa mora imati minimalno 6 znakova.

4. Arhitektura i dizajn sustava

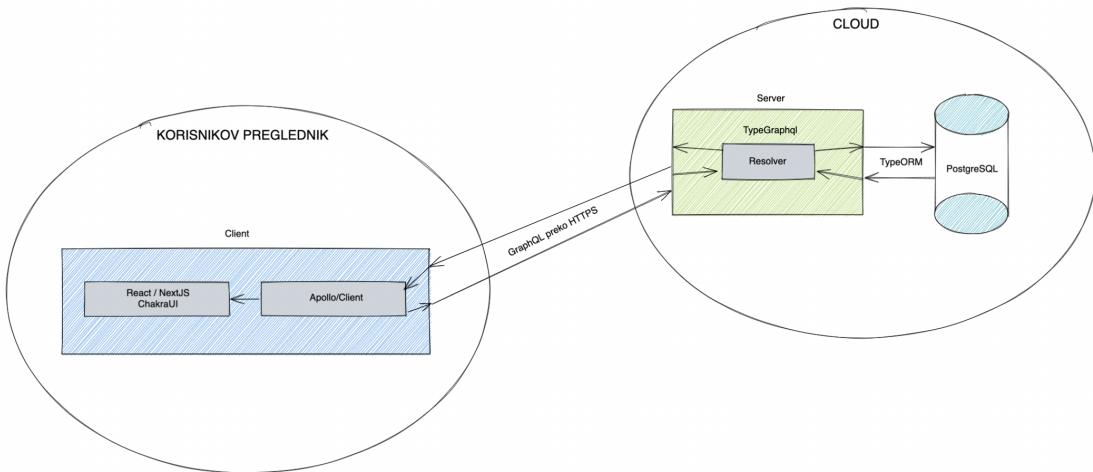
Arhitektura naše aplikacije, može se podijeliti na sljedeće komponente

- Server strana (*backend*)
- Klijent strana (*frontend*)
- Baza podataka (*db*)

Serverska strana glavna je komponenta naše aplikacije. U njoj se nalazi programska logika koja se brine da se korisniku isporuči ono što je zatražio i da aplikacija radi u skladu sa zahtjevima. Komunikacija prema klijentskoj strani ostvarena je korištenjem GraphQL-a (za manipulaciju s API-jem), a komunikacija prema bazi podataka ORM-om (Object-Relational Mapping). Također smo koristili razred Prisma kojem se predaju parametri SQL upita. Prisma je biblioteka sa serverske strane koja pomaže pri čitanju i pisanju podataka u i iz baze podataka na siguran način (štiti od SQL injekcija).

Baza podataka koju smo koristili je PostgreSQL. Kako bi izbjegli programske greške u SQLu te potencijalne ranjivosti (npr. SQL injekcija), za komunikaciju s bazom koristili smo *TypeORM* kako bi programer izbjegao greške u SQL jeziku. *TypeOrm* je podvrsta radnog okvira ORM prilagođen TypeScriptu i JavaScriptu. Kroz *TypeORM* dobili smo i sustav za verzioniranje baze podataka "migracije" tako da možemo lakše i sigurnije raditi izmjene na bazi te dijeliti te izmjene između programera u timu i produkcijskih servera.

Klijentska strana je pruža grafičko sučelje za olakšan i intuitivan pristup uslugama aplikacije. Ovu komponentu možemo zamisliti kao posrednika u komunikaciji između korisnika i serverske strane aplikacije. U našem slučaju klijentska strana implementirana je kao SPA (Single Page Application) s *React*-om. *React* je JavaScript knjižnica koja olakšava oblikovanje korisničkog sučelja.



Slika 4.1: Model arhitekture

4.1 Baza podataka

U našem sustavu korisimo PostgreSQL relacijsku bazu podataka za spremanje informacija o korisnicima (administratori i seizmolozi/znanstvenici), potresima i upitnicima te odgovorima iz tih upitnika. Zadaća baze podataka brza je i jednostavna pohrana, izmjena i dohvata podataka za daljnju obradu. Podaci će biti pohranjeni u tablice koje imaju svoje ime i atributte te će svaki entitet imati svoj primarni ključ, jedinstvenu oznaku svake n-torce koja ne smije imati vrijednost "null". U bazi podataka imamo sljedeće entitete:

- users
- earthquakes
- surveys
- survey_responses

4.1.1 Opis tablica

Tablica u kojoj se spremaju pristupni podaci te uloge korisnika u sustavu. Spremamo vrijeme kreiranja i izmjene podataka o korisniku. Svaki korisnik ima svoju e-mail adresu te hash svoje lozinke te ulogu - korisnik može biti ili administrator ili

seismolog/znanstvenik. U ovoj tablici također spremamo informaciju je li korisnik promijenio inicijalnu lozinku.

users		
id	INT	id korisnika
created_at	TIMESTAMP	vremenski žig kada je korisnik stvoren
updated_at	TIMESTAMP	vremenski žig kada je korisnik izmijenjen
email	VARCHAR	email adresa korisnika
password_hash	VARCHAR	lozinka korisnika spremljena kao hash
role	Enum:UserRole	uloga korisnika (admin i seismolog)
changedPassword	BOOLEAN	je li korisnik promijenio inicijalnu lozinku

Tablica u kojoj se spremaju prošli potresi, svaki potres može imati 0:n upitnika koje su građani ispunili. Spremamo vrijeme kreiranja i izmjene potresa. Podaci koje spremamo o potresu su naziv potresa, intenzitet, lokacija kao geografska širina i dužina te vrijeme kada se potres dogodio i kada je potres bio arhiviran.

earthquakes		
id	INT	id potresa
created_at	TIMESTAMP	vremenski žig kada je potres stvoren
updated_at	TIMESTAMP	vremenski žig kada je potres izmijenjen
name	VARCHAR	naziv potresa
strength	REAL	jačina potresa
epicenter_lat	REAL	geografska širina epicentra
epicenter_lng	REAL	geografska dužina epicentra
date	TIMESTAMP	vremenski žig kada se potres dogodio
archived_at	TIMESTAMP	vremenski žig kada je potres arhiviran

Tablica u kojoj se nalaze ispunjeni upitnici. Spremamo vrijeme kreiranja i izmjene upitnika. U bazi lokaciju korisnika spremamo kao geografsku širinu i dužinu odnosno dva broja, lat i long. Također spremamo jačinu potresa koju je

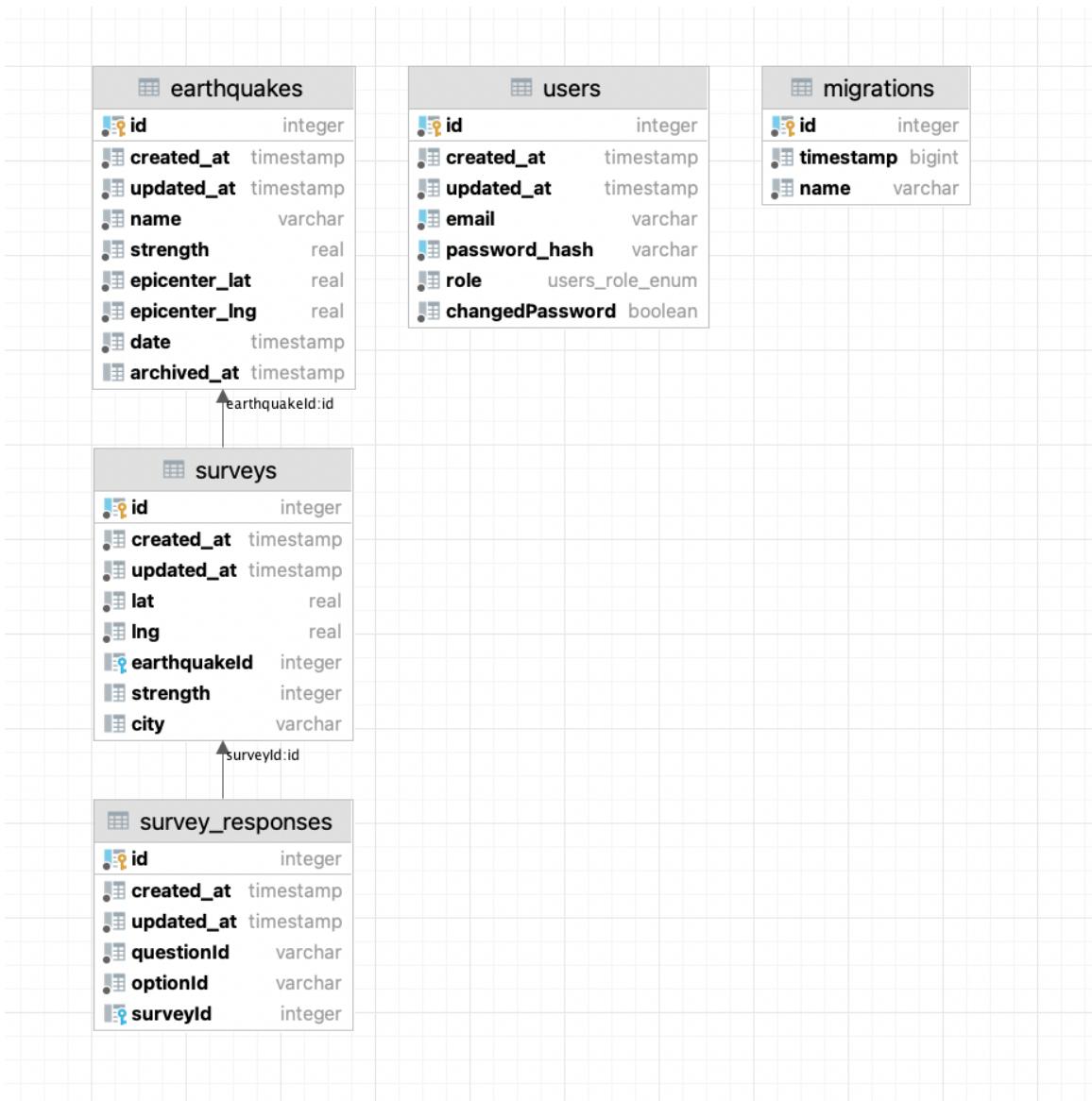
osoba osjetila te grad u kojem je upitnik ispunjen.

surveys		
id	INT	id upitnika
created_at	TIMESTAMP	vremenski žig kada je upitnik stvoren
updated_at	TIMESTAMP	vremenski žig kada je upitnik izmijenjen
lat	REAL	geografska širina upitnika
lng	REAL	geografska dužina upitnika
strength	INT	jačina koju je osoba koja je ispunila upitnik osjetila
city	VARCHAR	grad u kojem je upitnik ispunjen
earthquakeId	INT	id potresa za koji je ovaj upitnik ispunjen

Tablica u kojoj se nalaze odgovori pojedinačnog upitnika. Spremamo vrijeme kreiranja i izmjene svakog odgovora. U aplikaciji pitanja i odgovore imamo spremljene u listi tako da u ovoj tablici imamo indekse pitanja i odgovora.

survey_responses		
id	INT	id odgovora
created_at	TIMESTAMP	vremenski žig kada je odgovor stvoren
updated_at	TIMESTAMP	vremenski žig kada je odgovor izmijenjen
questionId	VARCHAR	oznaka pitanja
optionId	VARCHAR	oznaka odgovora
surveyId	INT	id upitnika za kojem pripada ovaj odgovor

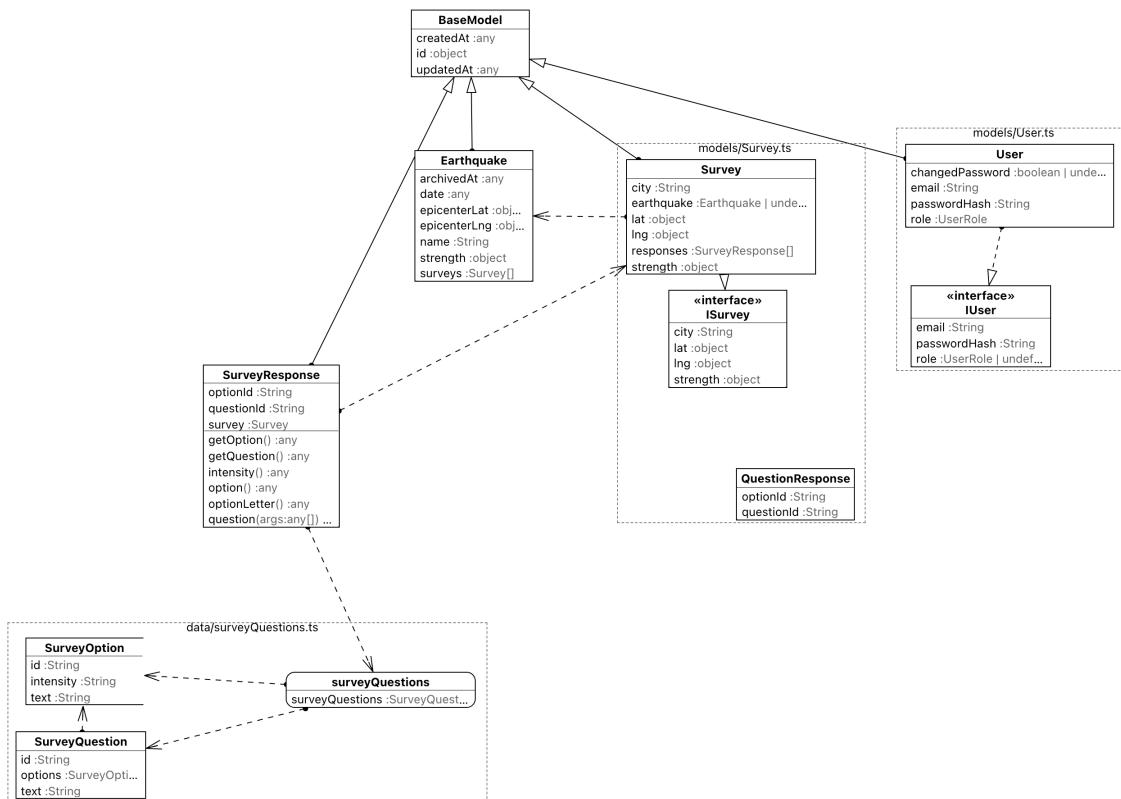
4.1.2 Dijagram baze podataka



Slika 4.2: Dijagram baze podataka

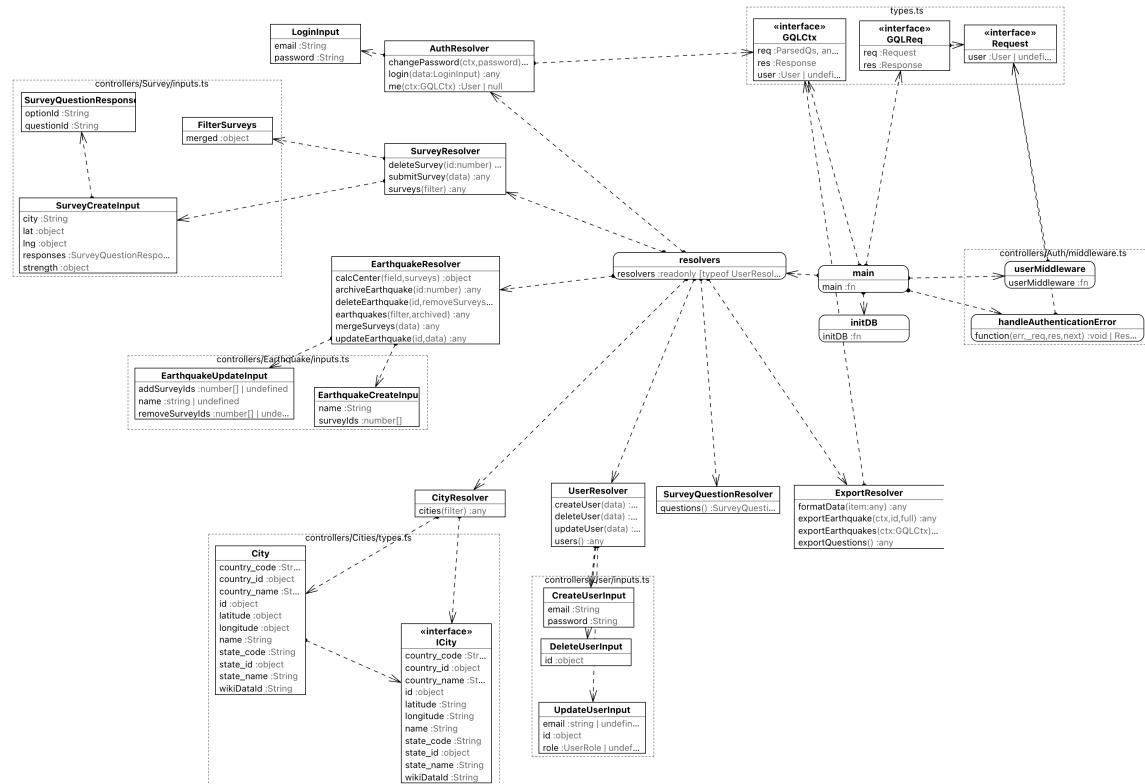
4.2 Dijagram razreda

Svi modeli iz baze podataka izvedeni su iz BaseModel razreda. Taj razred kao atributima ima oznaku (id) te vrijeme kreiranja i zadnje izmjene. Razred BaseModel koristimo zato što svi modeli sadržavaju te iste atributte. Razred User sadrži podatke o korisniku, e-mail, lozinka te je li korisnik administrator ili seismolog/znanstvenik. Razred Earthquake također sadrži atributte za vrijeme kada se potres dogodio, kada je arhivirat, koordinate epicentra, naziv i intenzitet te listu upitnika koje su korisnici ispunili za taj potres. Razred Survey opisuje upitnik atributima za ime grada u kojem je upitnik ispunjen (izračunato iz koordinati), potres za koji je upitnik ispunjen, koordinate na kojima je upitnik ispunjen (za računanje epicentra potresa), listu odgovora na pitanja te intenzitet potresa izračunat iz odgovora. SurveyResponse predstavlja odgovor na jedno pitanje u upitniku. Sadrži referencu na upitnik kojem pripada te oznake pitanja i odgovora. Popis pitanja i odgovora je spremlijen u surveyQuestions datoteci.

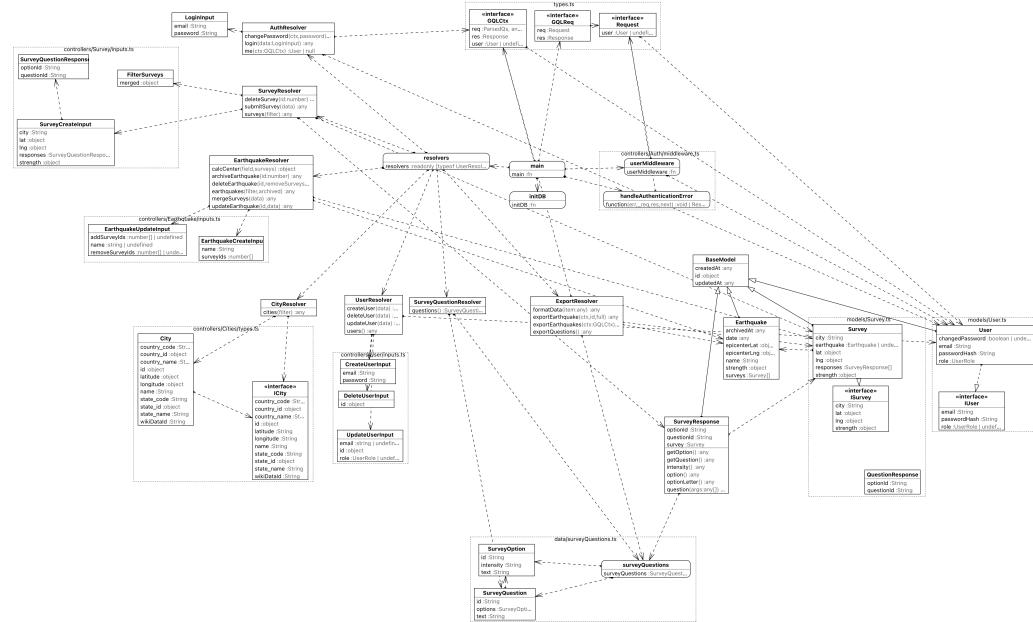


Slika 4.3: Dijagram razreda za modeliranje baze podataka

Dohvaćanje podataka te pisanje novih / modificiranje postojećih smo podijelili po svakom modelu u pojedine graphql resolvere. Za svaki resolver smo posebno napisali razrede koje validiraju inpute te ih koristili u samom resolveru. Svi resolveri osim Cities i SurveyQuestions ovise o bazi te komuniciraju s njom preko svojih modela (Surveys preko Survey modela, itd), dok Cities/SurveyQuestions direktno čitanju pitanja iz fileova. Svi resolveri su importani u resolvers.ts file iz kojeg se jedan array resolvera exporta te dodaje u inicijalizaciju servera koja se odvija u main.ts fileu. U njemu prije pokretanja servera inicijaliziramo konekciju s bazom podataka. Svaki request koji naš HTTP server prihvati prvo prolazi kroz user Middleware gdje se iz headera izvuče JWT token po kojem se dohvati korisnik te se po tome može znati im li korisnik prava pristupati nekim dijelovima aplikacije ili ne.



Slika 4.4: Dijagram razreda bez baze podataka



Slika 4.5: Dijagram razreda cijele aplikacije

4.3 Dijagram stanja

Dijagram stanja prikazuje stanja objekta te prijelaze iz jednog stanja u drugo temeljene na događajima. Na slici 4.6 prikazan je dijagram stanja za seizmologa koji je već prijavljen u sustav. Seizmolog je registrirani korisnik te ima više mogućnosti od građana koji nisu registrirani, ali manje od administratora.

Seizmologu se prvo, kao i ostalim korisnicima aplikacije, prikazuje Početna stranica. Na početnoj se stranici, kada joj netko pristupi, osvježe podatci na karti. Početna stranica seizmologu nudi pet opcija daljnog korištenja aplikacije te su one prikazane s pet stanja. U gornjem desnom kutu može odabrati mogućnost promjene svoje lozinke. Pri kliku na tu opciju, odmah mu se otvara obrazac za promjenu lozinke. Ako u obrazac upiše ispravnu novu lozinku te pritisne "Spremi", podatci će se ažurirati i vratiti seizmologa na početnu stranicu. Ako u obrazac ne upiše ispravnu novu lozinku, ostaje na istoj stranici i njegova lozinka se ne mijenja.

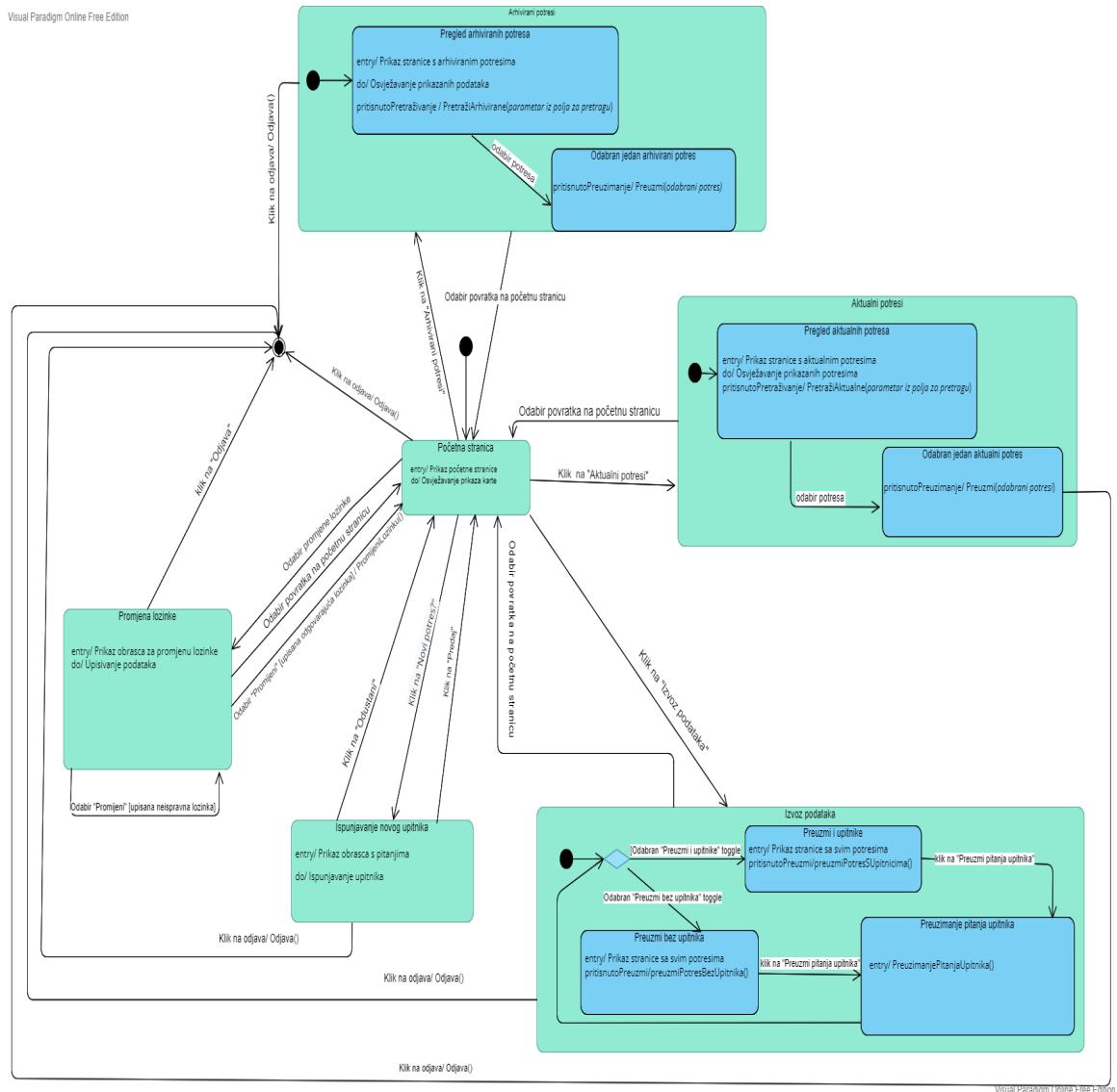
Iz stanja Početna stranica, seizmolog klikom na "Novi potres?" prelazi u stanje Ispunjavanje novog upitnika. U tom se stanju korisniku prikazuje obrazac s pitanjima o potresu kojeg korisnik želi prijaviti. Nakon što ispuni cijeli upitnih i pritisne "Spremi", aplikacija korisnika vraća u stanje Početna stranica. Ukoliko ipak odluči da ne želi ispuniti upitnik, utoliko će se povratkom na početnu stranicu ponovno naći u stanju Početna stranica.

Na početnoj stranici klikom na opciju "Arhivirani potresi", seizmolog prelazi u stanje Arhivirani potresi. U tom stanju odmah ulazi u podstanje Pregled arhiviranih potresa te mu se prikazuju svi arhivirani potresi te mogućnost pretraživanja tih potresa. Seizmolog ima mogućnost odabira i preuzimanja podataka o potresu (ispunjeni upitnici, lokacija, intezitet...). Preuzimanje podataka nalazi se u podstanju Odabran jedan arhivirani potres.

Iduće stanje u kojem se seizmolog može naći je Aktualni potresi. U stanje Aktualni potresi prelazi odabirom opcije "Aktuali potresi". Potom direktno prelazi u podstanje Pregled aktualnih potresa gdje mu se prikazuju aktualni potresi koje je, također, moguće filtrirati kao i arhivirane. Kao i u stanju Arhivirani potresi, seizmologu se i u stanju Aktualni potresi nudi mogućnost prelaska u podstanje Odabran jedan aktualni potres. U tom se podstanju provodi preuzimanje podataka o potresu. Iz stanja Aktualni potresi i Arhivirani potresi, seizmolog može prijeći u stanje Početna stranica.

Osim u podstanjima Odabran jedan aktualni potres i Odabran jedan arhivi-

rani potres, podaci se mogu preuzimati i u posebnom stanju Izvoz podataka. U to stanje seizmolog može doći klikom na "Izvoz podataka". Podstanje u kojem će se seismolog naći ovisi o tome je li "toggle" postavljen na "Preuzmi i upitnike" ili na "Izuzmi upitnike". U podstanju "Preuzmi i upitnike" prikazat će mu se stranica sa svim potresima te će klikom na "Preuzmi" preuzeti potrese zajedno sa pitanjima iz upitnika. Svi potresi se prikazuju i u podstanju "Preuzmi bez upitnika", no u tom će podstanju klikom na "Preuzmi" preuzeti samo podatke vezano uz potres, ali ne i pitanja. Iz oba podstanja klikom na "Preuzmi pitanja upitnika" prelazi se u podstanje "Preuzimanje pitanja upitnika". U tom podstanju preuzimaju se isključivo pitanja koja su u upitniku i svi mogući odgovori na njih. Iz tog postanja direktno se vraćamo na početak te, ovismo o tome je li "toggle" postavljen na "Preuzmi i upitnike" ili na "Izuzmi upitnike", odlazimo u jedno od tih podstanja. Iz stanja Izvoz podataka možemo prijeći u stanje Početna stranica. Završetak rada događa se klikom na "Odjavi se". Tu mogućnost seismolog ima u svim stanjima.



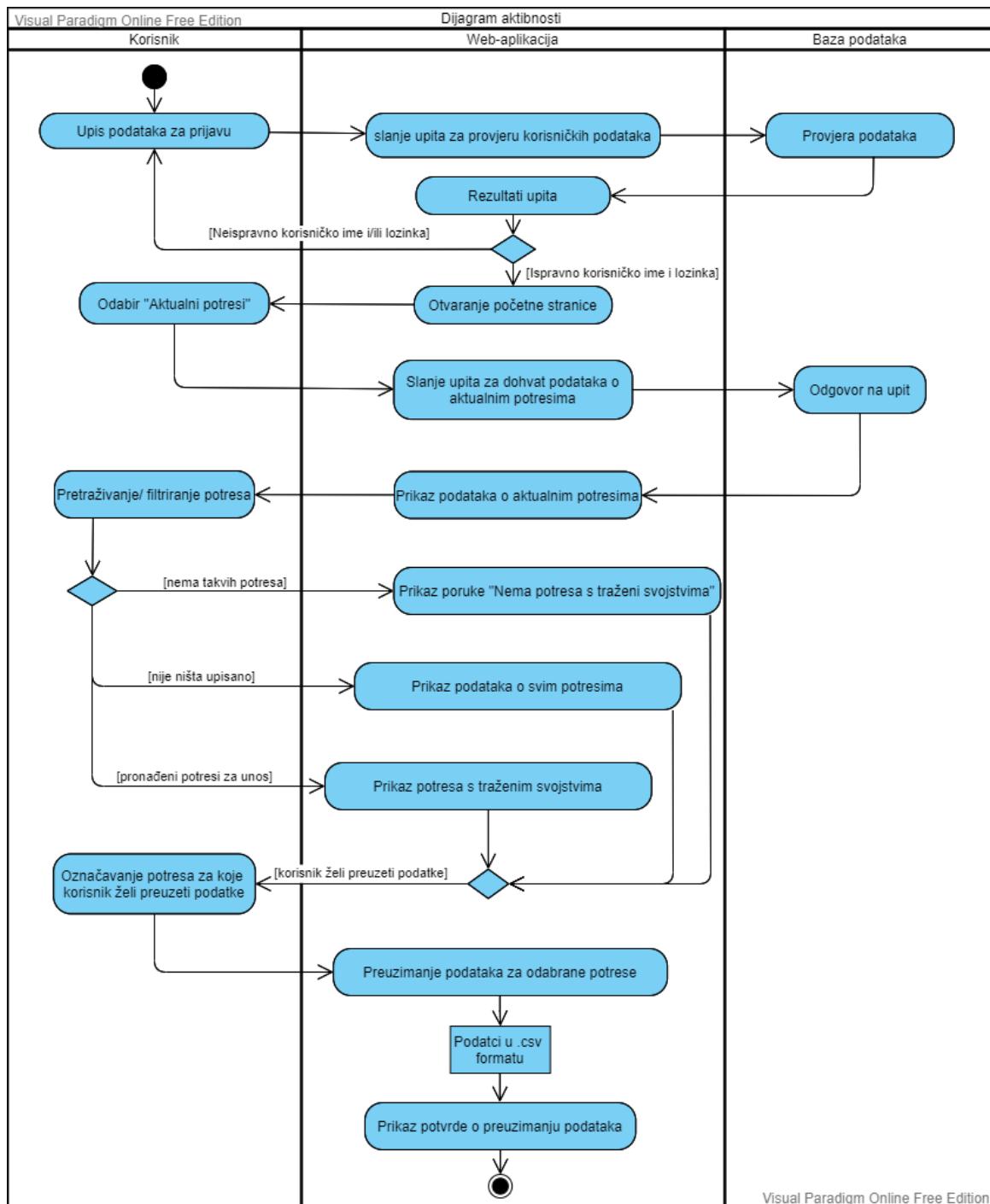
Slika 4.6: Dijagram stanja

4.4 Dijagram aktivnosti

Dijagram aktivnosti služi za modeliranje procesa. On opisuje ponašanje različitih elemenata modela toka upravljanja ili toka podataka. Ne upotrebljava se za modeliranje ponašanja poticanog događajima. Prilikom modeliranja toka upravljanja svaki novi korak poduzima se nakon završenog prethodnog. Na slici 4.7 prikazan je dijagram aktivnosti procesa pregleda aktualnih potresa, njihovog pretraživanja i preuzimanja podataka za njih. Na ovom dijagramu "Korisnik" može biti i seizmolog i administrator.

Prvo se korisnik prijavljuje u sustav. Nakon uspješne prijave, na početnoj stranici odabire "Aktualni potresi". Korisnik vidi prikaz stranice aktualnih potresa. Korisnik može pretraživati potrese prema nekom parametru. Ako nema potresa koji zadovoljavaju korisnikov upit, sustav javlja korisniku da takvih potresa nema. Potresi s traženim svojstvima prikazat će se ako postoje potresi koji zadovoljavaju upit (npr. pretraga po nazivu, mjestu, jačini...). Ako korisnik ne želi filtrirati potrese, jednostavno neće upisati nikakav upit u tražilicu te će se i dalje prikazivati lista svih aktualnih potresa.

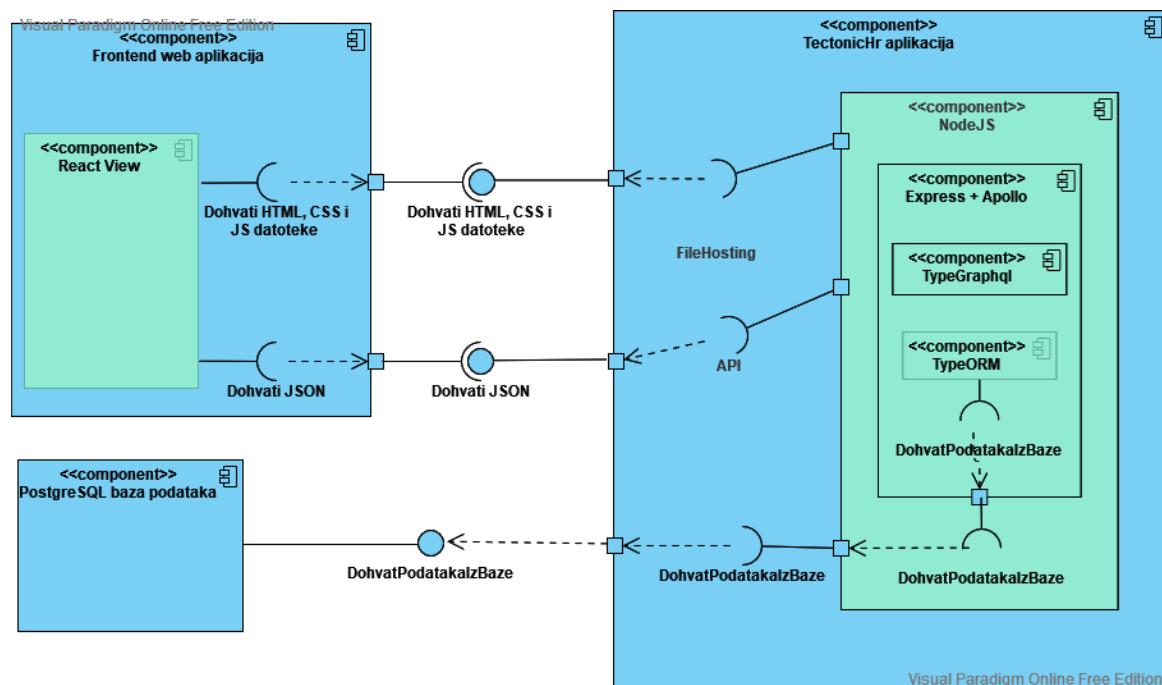
Kada je pronašao željeni potres te ga označio, klikom na "Preuzmi" korisnik pokreće preuzimanje podataka za odabrani potres. Podatci se preuzimaju u .csv formatu. Korisnika se obavještava o uspješnom preuzimanju podataka.



Slika 4.7: Dijagram aktivnosti

4.5 Dijagram komponenti

Dijagram komponenti prikazan na slici 4.8 prikazuje komponente aplikacije i njihovu međusobnu povezanost, strukture i odnos s okolinom. Aplikacija ima dvije pristupne točke sučelja: za dohvat HTML, CSS i JS datoteke ili dohvat JSON datoteke. Preko prvog sučelja pristupa se frontend dijelu aplikacije. Sve TypeScript datoteke koriste React biblioteku iz koje se mogu dohvatiti komponente za snalaženje na stranici, kao na primjer box, buttons, text itd. Drugo sučelje se koristi za pristup na backend, GraphQL komponentama. TypeORM razred se koristi za komunikaciju s bazom podataka i dohvatom SQL upita. Ti podaci se šalju onda u Services gdje se dalje vrše akcije s njima. React-view komponenta komunicira s TectonicHR aplikacijom preko poslanih sučelja, na kojima onda korisnik odreduje što želi vidjeti i raditi te ovisno o tome šalje zahtjeve aplikaciji.



Slika 4.8: Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

Pisana komunikacija u timu odvijala se putem aplikacije WhatsApp¹, a sastanci putem Google Meeta² i Discorda³. WhatsApp je besplatna mobilna aplikacija koja služi za razmjenu poruka, fotografija i videozapisa putem mobilnog interneta pametnim telefonom. Google Meet je usluga videokomunikacije koju je razvio Google, dok je Discord VoIP aplikacija koja omogućava komunikaciju glasom, videom i tekstrom.

UML dijagrami napravljeni su alatom Visual Paradigm Online⁴. Visual Paradigm Online je online alat za crtanje dijagrama i njihovu pohranu u web-pregledniku koji omogućava istovremeni rad više korisnika u stvarnom vremenu. Dijagrame možemo preuzeti u raznim formatima (.png, .jpeg, .pdf i dr.).

Za vizualizaciju stranice korištena je Figma⁵, besplatan alat za UI i UX dizajn, uređivanje i izradu prototipova te generiranje koda dostupan na webu ili u obliku desktop aplikacije.

Izvornim kodom upravljano je sustavom Git⁶. Git je distribuirani sustav za upravljanje različitim verzijama podataka (programskog koda, teksta i dr.). Sastoji se od udaljenog repozitorija koji se nalazi na nekoj Git platformi u oblaku i od lokalnih kopija tog repozitorija na računalima korisnika koji rade na projektu. Udaljeni repozitorij ovog projekta dostupan je na web platformi Gitlab⁷.

Kao razvojno okruženje korišten je Visual Studio Code⁸. Visual Studio Code je uređivač izvornog koda koji je razvio Microsoft za Linux, Windows i Mac OS platforme. Uključuje podršku za uklanjanje pogrešaka, isticanje sintakse, inteligentno dovršavanje koda, isječke, refaktoriranje koda i ugrađeni Git.

¹<https://www.whatsapp.com/>

²<https://meet.google.com/>

³<https://discord.com/>

⁴<https://online.visual-paradigm.com/>

⁵<https://www.figma.com/>

⁶<https://git-scm.com/>

⁷<https://about.gitlab.com/>

⁸<https://code.visualstudio.com/>

Osim VSCode-a, koristili smo i JetBrains WebStorm⁹ i JetBrains DataGrip¹⁰. WebStorm je integrirana razvojna okolina za JavaScript i povezane tehnologije, a DataGrip detektira moguće *buggove* u kodu i predlaže najbolje opcije za njihovo ispravljanje.

Cijeli sustav pisan je jezikom TypeScript¹¹ koji je proširenje jezika JavaScript (JavaScript s tipovima), skriptnog programskog jezika koji omogućava interakciju korisnika s web-stranicom.

Za frontend smo koristili React¹² i Chakra UI¹³. React je knjižnica koja služi za izgradnju korisničkog sučelja ili UI komponenti, a Chakra je jednostavna modularna biblioteka komponenata u kojoj se nalaze blokovi potrebnii za izgradnju React aplikacija.

PostgreSQL¹⁴ baza podataka spremljena je na serveru. PostgreSQL je besplatni i relacijski sustav upravljanja bazom podataka otvorenog koda dizajniran za upravljanje nizom radnih opterećenja, od pojedinačnih strojeva do skladišta podataka ili web-usluga s mnogim istovremenim korisnicima. Naš server nalazi se na Vultr VPS¹⁵ serveru.

⁹<https://www.jetbrains.com/webstorm/>

¹⁰<https://www.jetbrains.com/datagrip/>

¹¹<https://www.typescriptlang.org/>

¹²<https://reactjs.org/>

¹³<https://chakra-ui.com/>

¹⁴<https://www.postgresql.org/>

¹⁵<https://vultr.com>

5.2 Ispitivanje programskog rješenja

5.2.1 Ispitivanje komponenti

Testni Slučaj 1

```
const mocks = [
  {
    request: { query: me },
    result: {
      data: {
        me: null,
      },
    },
  },
];

describe("Actual", () => {
  it("renders a heading", () => {
    render(
      <MockedProvider mocks={mocks} addTypename={false}>
        <ActualEarthquakesPage />
      </MockedProvider>
    );
  });

  const heading = screen.getByText("Loading...");  

  expect(heading).toBeInTheDocument();
});
});
```

Testni Slučaj 2

```
const mocks = [
  {
    request: { query: me },
    result: {
      data: {
        me: null,
      },
    },
  },
];
```

```
        },
        ],
    };

const earthquakesMock = {
    request: { query: earthquakes, variables: { archived: true } },
    result: {
        data: {
            earthquakes: [],
        },
    },
};

describe("Archived", () => {
    it("renders a loading", () => {
        render(
            <MockedProvider mocks={mocks} addTypename={false}>
                <ArchivedEarthquakesPage />
            </MockedProvider>
        );
    });

    const heading = screen.getByText("Loading...");  
    expect(heading).toBeInTheDocument();
});  
});
```

Testni Slučaj 3

```
const mocks = [
    {
        request: { query: me },
        result: {
            data: {
                me: null,
            },
        },
    },
],
```

```
];

describe("Home", () => {
  it("renders a heading", () => {
    render(
      <MockedProvider mocks={mocks} addTypename={false}>
        <Navigation />
      </MockedProvider>
    );
  });

  const heading = screen.getByRole("heading", {
    name: /TECTONIC HR/i,
  });

  expect(heading).toBeInTheDocument();
});
});
```

Testni Slučajevi 4, 5, 6

```
describe("Intensity generation", () => {
  it("Expect to check simple", () => {
    const data = ["1", "5", "5", "5", "3"];
    expect(calcIntensity(data)).toBe(5);
  });

  it("Expect to more stuff simple", () => {
    const data = ["1", "<=3", ">=8", "7", "6", "<=4", "5", "5",
      "5", "3"];
    expect(calcIntensity(data)).toBe(8);
  });

  it("Expect to or simple", () => {
    const data = ["1", "2|3"];
    expect(calcIntensity(data)).toBe(3);
  });
});
```

5.2.2 Ispitivanje sustava

Testni Slučaj 1: Prijava s ispravnim podacima

- **Ulaz:**
 1. email: "foo@bar.com"
 2. password: "foobar123"
- **Očekivani izlaz:**
 1. prijava u sustav
 2. preusmjerjenje na početnu stranicu
- **Rezultat:**
 1. prijava u sustav
 2. preusmjerjenje na početnu stranicu

```
@Test()
public void loginTestGoodCredentials() {
    System.setProperty("webdriver.chrome.driver", "C:\\\\Program
        Files\\\\chromedriver\\\\chromedriver.exe");
    WebDriver driver = new ChromeDriver();
    driver.manage().timeouts().implicitlyWait(10,
        TimeUnit.SECONDS);
    // posjeti web stranicu
    driver.get("https://tectonichr.tk/");

    // otvori izbornik s gumbom za prijavu
    WebElement element =
        driver.findElement(By.id("menu-button-2"));
    element.click();

    // otvori stranicu za prijavu
    element = driver.findElement(By.id("menu-list-2"));
    element.click();

    // unesi email
    element = driver.findElement(By.name("email"));
    element.sendKeys("foo@bar.com");

    // unesi zaporku
```

```
element = driver.findElement(By.name("password"));
element.sendKeys("foobar123");

// pritiskom na gumb pokusaj se prijaviti u sustav
driver.findElement(By.cssSelector("button[type='submit']")).click();

try {
    // pricekaj dok se provjeri prijava
    TimeUnit.MILLISECONDS.sleep(1000);
} catch (Exception e) {
    System.out.println("Exception while waiting for login: " +
        e.getMessage());
}

// ako je doslo do preusmjerenja na pocetnu stranicu prijava
// je uspjesna
String redirURL = driver.getCurrentUrl();
boolean compRes = redirURL.equals("https://tectonichr.tk/");
assertEquals(compRes, true);

driver.quit();
}
```

Testni Slučaj 2: Prijava s neispravnim podacima

- **Ulaz:**

1. email: "neispravni@email.com"
2. password: "neispravnaZaporka"

- **Očekivani izlaz:**

1. neuspješna prijava u sustav

- **Rezultat:**

1. neuspješna prijava u sustav

```
@Test()
public void loginTestBadCredentials() {
    System.setProperty("webdriver.chrome.driver", "C:\\\\Program
        Files\\\\chromedriver\\\\chromedriver.exe");
    WebDriver driver = new ChromeDriver();
```

```
driver.manage().timeouts().implicitlyWait(10,
    TimeUnit.SECONDS);
// posjeti web stranicu
driver.get("https://tectonichr.tk/");

// otvori izbornik s gumbom za prijavu
WebElement element =
    driver.findElement(By.id("menu-button-2"));
element.click();

// otvori stranicu za prijavu
element = driver.findElement(By.id("menu-list-2"));
element.click();

// unesi email
element = driver.findElement(By.name("email"));
element.sendKeys("krivi@email.com");

// unesi zaporku
element = driver.findElement(By.name("password"));
element.sendKeys("neispravnaSifra");

// pritiskom na gumb pokusaj se prijaviti u sustav
driver.findElement(By.cssSelector("button[type='submit']")).click();

try {
    // pricekaj dok se provjeri prijava
    TimeUnit.MILLISECONDS.sleep(1000);
} catch (Exception e) {
    System.out.println("Exception while waiting for login: " +
        e.getMessage());
}

// ako nije doslo do preusmjerenja uspjesno je prepoznata
// prijava s neispravnim
// podacima
String redirURL = driver.getCurrentUrl();
boolean compRes =
```

```
    redirURL.equals("https://tectonichr.tk/auth/login");
    assertEquals(compRes, true);

    driver.quit();
}
```

Testni Slučaj 3: Unos novog seismologa sa ispravnom e-mail adresom

- **Ulaz:**

1. email: "slučajno izgenerirana e-mail adresa"
2. password: "123456"

- **Očekivani izlaz:**

1. novi seismolog
2. preusmjerenje na stranicu sa seismolozima

- **Rezultat:**

1. novi seismolog
2. preusmjerenje na stranicu sa seismolozima

```
@Test()
public void addSeismologistValidEmail() {
    System.setProperty("webdriver.chrome.driver", "C:\\\\Program
        Files (x86)\\\\Chrome Driver\\\\chromedriver.exe");
    WebDriver driver = new ChromeDriver();
    driver.manage().timeouts().implicitlyWait(10,
        TimeUnit.SECONDS);

    // posjeti pocetnu stranicu
    driver.get("https://tectonichr.tk/");

    // otvori izbornik
    WebElement element =
        driver.findElement(By.id("menu-button-2"));
    element.click();

    // pritisni gumb i otvori stranicu za prijavu
    element = driver.findElement(By.id("menu-list-2"));
    element.click();
```

```
// unesi email adresu
element = driver.findElement(By.name("email"));
element.sendKeys("foo@bar.com");

// unesi zaporku
element = driver.findElement(By.name("password"));
element.sendKeys("foobar123");

// posalji podatke i provjeri
driver.findElement(By.cssSelector("button[type='submit']")).click();

try {
    // pricekaj promjenu stranice
    TimeUnit.MILLISECONDS.sleep(1000);
} catch (Exception e) {
    System.out.println("Exception while waiting for login: " +
        e.getMessage());
}

// otvori izbornik
element = driver.findElement(By.id("menu-button-2"));
element.click();

// otvori stranicu seizmologa
element = driver.findElement(By.id("menu-list-2-menuitem-4"));
element.click();

// pritisni gumb za dodavanje novog seizmologa
element = driver.findElement(By.className("css-415j94"));
element.click();

// generiraj random email
Random randomGenerator = new Random();
int randomInt = randomGenerator.nextInt(1000);
String email="username"+ randomInt +"@gmail.com";

// unesi email
element = driver.findElement(By.name("email"));
element.sendKeys(email);
```

```
// unesi zaporku
element = driver.findElement(By.name("password"));
element.sendKeys("123456");

// posalji podatke i provjeri
driver.findElement(By.cssSelector("button[type='submit']")).click();

try {
    // pricekaj promjenu stranice
    TimeUnit.MILLISECONDS.sleep(1000);
} catch (Exception e) {
    System.out.println("Exception while waiting for login: " +
        e.getMessage());
}

// dohvati promjenjenu stranicu
String redirURL = driver.getCurrentUrl();

// ako je doslo do preusmjerenja, uspjesno je dodan novi
// seizmolog
boolean compRes =
    redirURL.equals("https://tectonichr.tk/admin/users");
assertEquals(compRes, true);

driver.quit();
}
```

Testni Slučaj 4: Unos novog seismologa sa postojećom e-mail adresom

- **Ulaz:**
 1. email: "existing@email.com"
 2. password: "123456"
- **Očekivani izlaz:**
 1. neuspješno dodavanje novog seismologa
 2. poruka o već postojećoj e-mail adresi
- **Rezultat:**
 1. neuspješno dodavanje novog seismologa
 2. poruka o već postojećoj e-mail adresi

```
@Test()
public void addSeismologistExistingEmail() {
    System.setProperty("webdriver.chrome.driver", "C:\\\\Program
        Files (x86)\\\\Chrome Driver\\\\chromedriver.exe");
    WebDriver driver = new ChromeDriver();
    driver.manage().timeouts().implicitlyWait(10,
        TimeUnit.SECONDS);

    // posjeti pocetnu stranicu
    driver.get("https://tectonichr.tk/");

    // otvori izbornik
    WebElement element =
        driver.findElement(By.id("menu-button-2"));
    element.click();

    // pritisni gumb i otvori stranicu za prijavu
    element = driver.findElement(By.id("menu-list-2"));
    element.click();

    // unesi email
    element = driver.findElement(By.name("email"));
    element.sendKeys("foo@bar.com");

    // unesi zaporku
    element = driver.findElement(By.name("password"));
    element.sendKeys("foobar123");

    // posalji podatke i provjeri
    driver.findElement(By.cssSelector("button[type='submit']")).click();

    try {
        // pricekaj promjenu stranice
        TimeUnit.MILLISECONDS.sleep(1000);
    } catch (Exception e) {
        System.out.println("Exception while waiting for login: " +
            e.getMessage());
    }
}
```

```
}

// otvori izbornik
element = driver.findElement(By.id("menu-button-2"));
element.click();

// otvori stranicu seismologa
element = driver.findElement(By.id("menu-list-2-menuitem-4"));
element.click();

// pritisni gumb za dodavanje novog seismologa
element = driver.findElement(By.className("css-415j94"));
element.click();

// unesi postojeći email
element = driver.findElement(By.name("email"));
element.sendKeys("existing@email.com");

// unesi zaporku
element = driver.findElement(By.name("password"));
element.sendKeys("123456");

// posalji podatke i provjeri
driver.findElement(By.cssSelector("button[type='submit']")).click();

try {
    // pričekaj promjenu stranice
    TimeUnit.MILLISECONDS.sleep(1000);
} catch (Exception e) {
    System.out.println("Exception while waiting for login: " +
        e.getMessage());
}

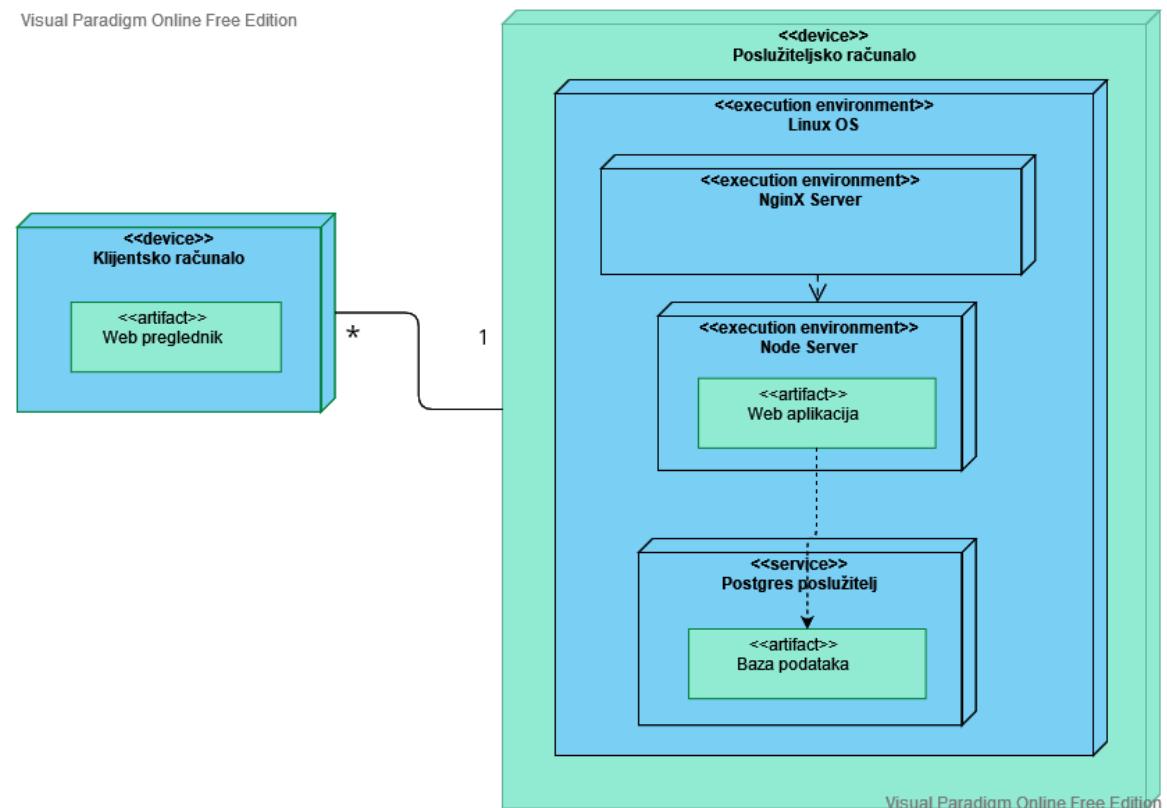
String redirURL = driver.getCurrentUrl();

// ako stranica nije promjenjena, email vec postoji
boolean compRes =
    redirURL.equals("https://tectonichr.tk/admin/users/new");
assertEquals(compRes, true);
```

```
    driver.quit();  
}
```

5.3 Dijagram razmještaja

Dijagram razmještaja opisuje topologiju sustava i odnos sklopoških i programskih dijelova. Specifikacijski dijagram razmještaja stavlja naglasak na implementaciju artefakata bez upućivanja na specifične slučajeve artefakata ili čvorova. Slika 5.1. prikazuje specifikacijski dijagram razmještaja cijele aplikacije. Na poslužiteljskom računalu, koje je izvedeno na Linux operacijskom sustavu, se nalaze Node server i Postgres baza podataka. Nginx služi kao *reverse proxy* (služi za proxy-anje HTTP zahtjeva) se nalazi ispred node servera koji salje static fileove za frontend (.js/.css/.html) te JSON podatke preko GraphQL Apija. Sustav se temelji na arhitekturi "klijent-poslužitelj", a komunikacija između njih odvija se pomoću HTTP protokola (koristenjem GraphQL jezika).



Slika 5.1: Dijagram razmještaja

5.4 Upute za puštanje u pogon

5.4.1 Lokalno pokretanje

Da bi se cijela aplikacija pokrenula trebate imati pristup nekoj postgres bazi podataka (može biti lokalno ili udaljeno/docker). Da biste instalirali pakete preporučuje se yarn ali može i s npm-om, pokrenite yarn ili npm install u folderima frontend i backend. Nakon toga ako želite pokrenuti backend, u backend folderu trebate kopirati .env.example u .env i popuniti ga s podatcima za pristup vašoj bazi podataka. Nakon toga trebate pokrenuti yarn db:rebuild sto će pokrenuti sve migracije te ubaciti osnovne podatke u bazu. Nakon toga prelazimo na frontend, njega možete pokrenuti i bez backenda (spajajući se na produkcijski backend). Nakon što ste pokrenuli yarn trebate kopirati .env.example u .env i dodati token od vašeg MapBox računa. Za spajanje na udaljeni backend možete zamijeniti localhost u NEXT_PUBLIC_APOLLO_SERVER s udaljenim linkom. To je sve što vam treba od postavljanja. Za pokretanje backenda lokalno koristite yarn dev, dok za frontend koristite yarn dev i ako želite automatski generirati graphql queryje yarn gen:watch, no njih možete i manualno pokretati s yarn gen svaki puta kada napravite promjenu

5.4.2 Puštanje u pogon na udaljenom serveru

Mi smo za ovo koristili linux VPS server tvrtke Vultr na kojem je bio instaliran Ubuntu. Za samo pokretanje koristili smo node alat pm2. Na serveru smo klonirali repository, pokrenuli yarn u oba foldera te nakon toga pokrenuli yarn frontend:build unutar backenda, to izgenerira produkcijsku verziju frontenda. Nakon toga smo sa pm2 startOrRestart ecosystem.config.js u rootu projekta pokrenuli servery cijele aplikacije

Dodatno smo setupirali i nginx da bi dobili reverse proxy za domenu, te bazu podataka na istom serveru (ali to ovisi od servera do servera, pa nećemo ici u detalje)

6. Zaključak i budući rad

Naš tim radio je na razvoju web aplikacije TectonicHR koja služi olakšanom prikupljanju i prikazu podataka o potresima. Korisnike koji pristupaju aplikaciji podijelili smo u tri kategorije: administrator, građani (neregistrirani korisnici) i se-izmolozi (korisnici koje registrira administrator). Potrese smo podijelili na arhivirane i aktualne. Aktualni potresi su oni za koje se još mogu ispuniti upitnici. Kada administrator arhivira neki potres, on prelazi u arhivirane potrese te se za njega više ne može ispuniti upitnik. Korisnici mogu ispuniti upitnik i za neki potres koji još nije ni u arhiviranim ni u aktualnim potresima. Na taj način oni prijavljuju novi potres. Na temelju odgovora iz upitnika računa se intenzitet potresa te se potresi prikazuju na karti bojom koja ovisi o njihovom intenzitetu.

Prije početka razvoja aplikacije, tim se morao sastati i razjasniti temu zadatka nakon čega je slijedila okvirna podjela poslova (tko bi htio raditi koji dio). Prvi dio rada tima više se fokusirao na dokumentaciju projekta. Upravo ta detaljna analiza zahtjeva aplikacije kasnije je olakšala izradu same aplikacije. Ova faza rada uključivala je dokumentiranje funkcionalnih i nefunkcionalnih zahtjeva, razradu obrazaca uporabe te crtanje dijagrama (dijagram obrazaca uporabe, sekvensijski dijagram, dijagram razreda, model baze podataka). Za izradu dijagrama i dokumentiranje projekta bilo je potrebno znanje s predavanja predmeta te se svaki dio dokumentacije pisao nakon što bi bio ispričan na predavanju. U prvom dijelu rada na projektu također smo izradili prototip u Figmi koji nam je dao ideju izgleda konačne verzije aplikacije i olakšao u izradi frontenda.

Druga faza projekta više se orijentirala na pisanje koda. Za izradu frontenda se koristio React što je većini članova bilo nepoznato te je zahtijevalo više samostalnog i ubrzanog učenja. Obrasci i dijagrami izrađeni u prvoj fazi su uvelike pomogli u izradi frontenda i backenda jer su pokazali kako bi aplikacija trebala komunicirati s korisnikom te odrediti gdje smije pristupiti, a gdje ne jer nemaju svi korisnici ista prava i mogućnosti u aplikaciji. U ovoj fazi smo dokumentirali dijagram stanja, dijagram aktivnosti, dijagram razmještaja i dijagram komponenti.

Dogовори међу članovima tima su se odvijali preko WhatsAppa te na čestim sastancima uživo čime smo postigli informiranost svih članova grupe o napretku

projekta. Sudjelovanje na ovakvom projektu bilo je vrijedno iskustvo svim članovima tima jer smo kroz intenzivnih nekoliko tjedana rada iskusili što to znači raditi u timu na nekom projektu. Također, osjetili smo važnost dobre vremenske organiziranosti, koordiniranosti i komunikacije između članova tima. Zadovoljni smo postignutim iako vidimo prostor za poboljšanje izrađene aplikacije što je posljedica neiskustva članova tima te premalenog vremenskog okvira za izradu aplikacije.

Popis literature

1. Programsko inženjerstvo, FER ZEMRIS, <http://www.fer.hr/predmet/proinzz.html>
2. I. Sommerville, "Software engineering", 8th ed, Addison Wesley, 2007.
3. T.C.Lethbridge, R.Langaniere, "Object-Oriented Software Engineering", 2nd ed. McGraw-Hill, 2005.
4. I. Marsic, Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, <http://www.ece.rutgers.edu/~marsic/books/SE>
5. The Unified Modeling Language, <https://www.uml-diagrams.org/>
6. Astah Community, <http://astah.net/editions/uml-new>
7. EMSC, <https://www.emsc-csem.org/#2>
8. TypeGraphql dokumentacija, <https://typegraphql.com/>
9. GraphQL dokumentacija, <https://graphql.org/>
10. Apollo client dokumentacija, <https://www.apollographql.com/docs/react/>
11. ChakraUI dokumentacija, <https://chakra-ui.com/>
12. TypeORM dokumentacija, <https://typeorm.io/>
13. Tutorial: Intro to React, <https://reactjs.org/tutorial/tutorial.html>
14. React Tutorial, w3schools, <https://www.w3schools.com/react/default.asp>
15. Design, Figma, <https://www.figma.com/design/>

Indeks slika i dijagrama

2.1 Slika 2.2: početna stranica EMSC-a	8
3.1 Dijagram obrazaca uporabe, funkcionalnosti koje imaju svi korisnici	24
3.2 Dijagram obrazaca uporabe, funkcionalnosti koje imaju registrirani korisnici	25
3.3 sekvencijski dijagram za UC8	27
3.4 sekvencijski dijagram za UC20	28
4.1 Model arhitekture	31
4.2 Dijagram baze podataka	34
4.3 Dijagram razreda za modeliranje baze podataka	35
4.4 Dijagram razreda bez baze podataka	36
4.5 Dijagram razreda cijele aplikacije	37
4.6 Dijagram stanja	40
4.7 Dijagram aktivnosti	42
4.8 Dijagram komponenti	43
5.1 Dijagram razmještaja	58

Dodatak: Prikaz aktivnosti grupe

Dnevnik sastajanja

1. sastanak

- Datum: 9.10.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, D.Vorkapić
- Teme sastanka:
 - upoznavanje članova tima

2. sastanak

- Datum: 19.10.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, M.Jurić, D.Vorkapić, F.Zekan
- Teme sastanka:
 - rasprava o nedoumicama u zadanom zadatku
 - razjašnjavanje zahtjeva
 - dogovor o korištenim tehnologijama
 - upoznavanje s korištenim tehnologijama
 - razrada specifikacije programske potpore

3. sastanak

- Datum: 26.10.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, M.Jurić, D.Vorkapić
- Teme sastanka:
 - pregled dosad izradjene dokumentacije
 - prijedlog potrebnih izmjena
 - dogovor o izgledu stranice
 - sastavljanje prototipa u alatu Figma
 - podjela zadataka

4. sastanak

- Datum: 9.11.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, M.Jurić, D.Vorkapić,

F.Zekan

- Teme sastanka:

- pregled dosad izrađene dokumentacije
 - diskusija o implementaciji aplikacije

5. sastanak

- Datum: 17.12.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, M.Jurić, D.Vorkapić, F.Zekan
- Teme sastanka:
 - izrada dijagrama stanja
 - izrada dijagrama aktivnosti
 - diskusija i podjela poslova u dalnjem radu

6. sastanak

- Datum: 8.01.2021.
- Prisustvovali: M.Ćurković, D.Čemeljić, A.Engler, K.Iličić, M.Jurić, D.Vorkapić, F.Zekan
- Teme sastanka:
 - pregled dosad izrađene dokumentacije
 - ispravljanje nekonzistentnosti u dokumentaciji
 - diskusija o preostalim zadacima

Tablica aktivnosti

	Maja Juric	David Čemeljić	Mihovil Ćurković	Antonija Engler	Klara Iličić	Dalijo Vorkapić	Fran Zekan
Upravljanje projektom	9						
Opis projektnog zadatka	3				4		
Funkcionalni zahtjevi		2	3	1			
Opis pojedinih obrazaca	3			3	3		
Dijagram obrazaca	3				3		
Sekvencijski dijagrami	4				4		
Opis ostalih zahtjeva		2					
Arhitektura i dizajn sustava		6				3	6
Baza podataka						3	3
Dijagram razreda		2					2
Dijagram stanja	1				1		
Dijagram aktivnosti	1		1		1		
Dijagram komponenti		1		1			1
Korištene tehnologije i alati	1				1		
Ispitivanje programskog rješenja			4			4	
Dijagram razmještaja		1		1			1
Upute za puštanje u pogon		2					2
Dnevnik sastajanja	2						
Zaključak i budući rad	2						

Nastavljeno na idućoj stranici

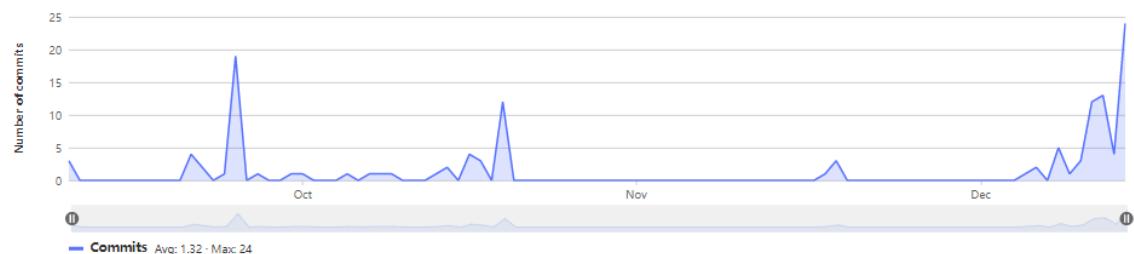
Nastavljeno od prethodne stranice

		Maja Jurić	David Čemeljić	Mihovil Ćurković	Antonija Engler	Klara Iličić	Dalijo Vorkapić	Fran Zekan
Popis literature	0.5						0.5	
Inicijalizacija aplikacije								4
Autentikacija			3					
Upitnici				3				
Kreiranje potresa							3	
Puštanje u pogon								2

Dijagrami pregleda promjena

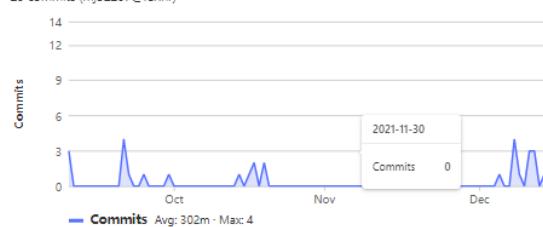
Commits to main

Excluding merge commits. Limited to 6,000 commits.



majajuric

29 commits (mj52207@fer.hr)



Fran Zekan

28 commits (zekan.fran369@gmail.com)



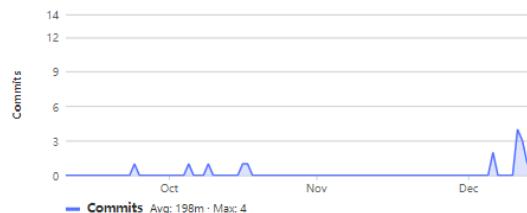
davidcemeljic

25 commits (david.cemeljic@fer.hr)



Klarall

19 commits (klara.ilic@fer.hr)



antonijaengler

12 commits (antonija.engler@fer.hr)



ecurmih

5 commits (mihovil.curkovic@ericsson.com)

