



RešiMe

Sajt za prijavljivanje komunalnih dužnosti
Arhitekturni Projekat

Verzija: 1.0

Autor: Filip Menković, Filip Jovanović

Datum: 25.10.2025

Sadržaj

1.	Cilj dokumenta	4
2.	Opseg dokumenta	4
3.	Reference	4
4.	Predstavljanje arhitekture	4
5.	Ciljevi i ograničenja arhitekture	4
6.	Pogled na slučajeve korišćenja	5
6.1.	Dijagrami slučajeve korišćenja	5
6.2.	Kratak opis slučajeve korišćenja	7
6.2.1.	Kreiranje objave	7
6.2.2.	Brisanje objave	7
6.2.3.	Pretraga objava	8
6.2.4.	Pregled objava	8
6.2.4.1.	Minimalni pregled objave	8
6.2.4.2.	Detaljni pregled objave	8
6.2.5.	Filtriranje objava	8
6.2.6.	Pregled mape	8
6.2.7.	Postavljanje komentara	8
6.2.8.	Kreiranje naloga	8
6.2.9.	Rešavanje problema	9
6.2.10.	Dodavanje role	9
7.	Pregled na logičku arhitekturu sistema	9
7.1.	Pregled arhitekture – organizacija paketa i podsistema u slojeve	9
7.1.1.	Korisnički interfejs	10
7.1.2.	Aplikaciona logika	10
7.1.3.	Pristup podacima	10
7.1.4.	.NET	10
7.1.5.	SQL Server Express	10
7.1.6.	Angular	10
7.1.6.1.	HTML	11
7.1.6.2.	SCSS	11
7.1.6.3.	TypeScript	11
8.	Pogled na procese	11
8.1.	Web čitač	11
8.2.	Web server	12
8.3.	.NET	12
8.4.	SQL Server Express	12
9.	Pogled na raspoređivanje sistema	12
9.1.	Klijent	12
9.2.	Web server	12
9.3.	DBMS server	13
10.	Pogled na implementaciju sistema	13
10.1.	Model domena	13

10.2.	Šema baze podataka	14
11.	Performanse	14
12.	Kvalitet	14

1. Cilj dokumenta

Cilj ovog dokumenta je detaljni opis arhitekture RešiMe portala – veb-aplikacije razvijene od strane tima FFD koja građanima omogućava da prijavljuju, prate i komentarišu komunalne i druge svakodnevne probleme, a nadležnim organima da ih evidiraju, klasifikuju i označavaju kao rešene. Arhitektura je zasnovana na modernim veb-tehnologijama: Angular (klijent), .NET (serverska logika) i SQL Server Express (baza podataka). Dokument pruža uvid u sve arhitektonske poglede: slučajeve korišćenja, logičku i procesnu arhitekturu, raspoređivanje komponenata, model domena, šemu baze, performanse i mere kvaliteta, čime se stvara osnova za dalji razvoj, održavanje i skaliranje sistema.

2. Opseg dokumenta

Dokument obuhvata kompletan arhitektonski opis RešiMe portala koji će biti razvijen od strane tima FFD korišćenjem tehnologija Angular, .NET i SQL Server Express. RešiMe predstavlja interaktivnu veb-platformu namenjenu građanima i nadležnim organima za prijavu, praćenje i rešavanje komunalnih i drugih svakodневnih problema. Namena sistema je dvojaka: građanima omogućuje jednostavno i brzo prijavljivanje problema (putem teksta, slika i geolokacije), dok organima vlasti pruža pregledne, filtrirane i geolocirane izveštaje radi efikasnije intervencije. Dokument pokriva sve arhitektonske slojeve, od korisničkog interfejsa, preko aplikacione logike i pristupa podacima, do procesne i rasporedne arhitekture, performansi i mera kvaliteta.

3. Reference

Spisak korišćene literature:

RešiMe – Predlog projekta, FFD-RešiMe-01, V1.0, 2025, FFD tim.

RešiMe – Vizija sistema, V1.0, 2025, FFD tim.

4. Predstavljanje arhitekture

Arhitektura sistema u dokumentu je prikazana kao serija pogleda na sistem: pogled na slučajeve korišćenja, pogled na logičku arhitekturu sistema, pogled na procese, pogled na razmeštaj komponenti sistema i pogled na implementaciju. Ovi pogledi su predstavljeni odgovarajućim UML dijagramima.

5. Ciljevi i ograničenja arhitekture

Ključni zahtevi i sistemska ograničenja koja imaju značajan uticaj na izbor arhitekture i projektovanje sistema su:

1. RešiMe portal će biti implementiran kao Web aplikacija zasnovana na .NET platformi i SQL Server Express bazi podataka [4].
2. Klijentski deo RešiMe portala će biti optimizovan za sledeće Web čitače: Google Chrome 90 i noviji, Mozilla Firefox 88 i noviji, Microsoft Edge 90 i noviji, kao i Safari 14 i noviji [4].
3. Svi zahtevi u pogledu performansi dati u [5] moraju biti uzeti u obzir pri izboru arhitekture i razvoju sistema.

4. Klasifikaciju objava vrši se prema unapred definisanim kategorijama problema (saobraćaj, komunalije, javna rassveta, itd.).

6. Pogled na slučajeve korišćenja

U ovom odeljku je dat pogled na slučajeve korišćenja definisane u specifikaciji zahteva [5].

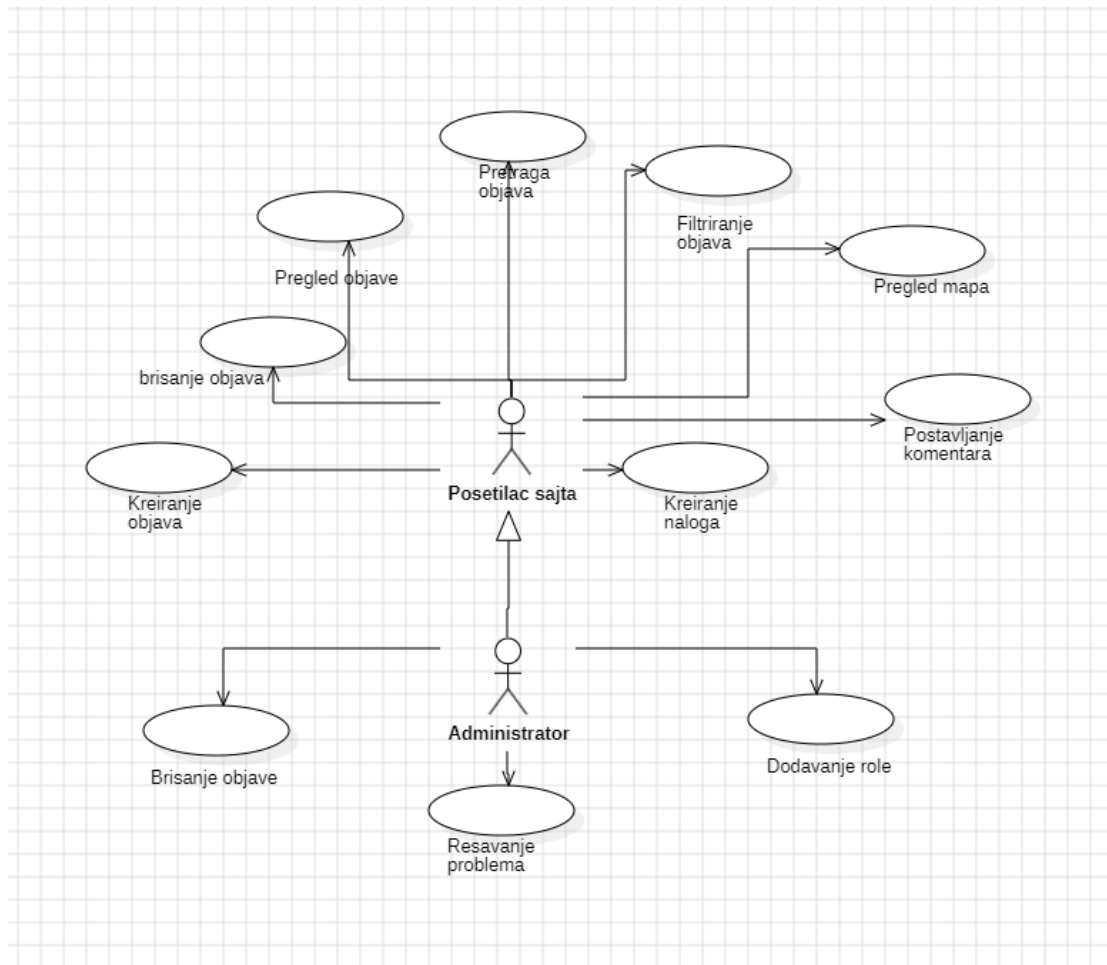
Slučajevi korišćenja RešiMe portala su:

- Kreiranje objave
- Brisanje objave
- Pretraga objava
- Pregled objava
 - Minimalni pregled objave
 - Korisnik
 - Naslov
 - Tekst
 - Datum
 - Otvaranje detaljnog pregleda
 - Detaljni pregled objave
 - Obriši objavu
 - Slika
 - Komentari
- Filtriranje objava
 - Ozbiljnost
 - Regije
 - Oznake
- Pregled mape
 - Filtriranje
 - Pregled pinova
- Postavljanje komentara
- Kreiranje naloga
- Rešavanje problema
- Dodavanje role

Ove slučajeve korišćenja mogu instancirati korisnik ili administrator.

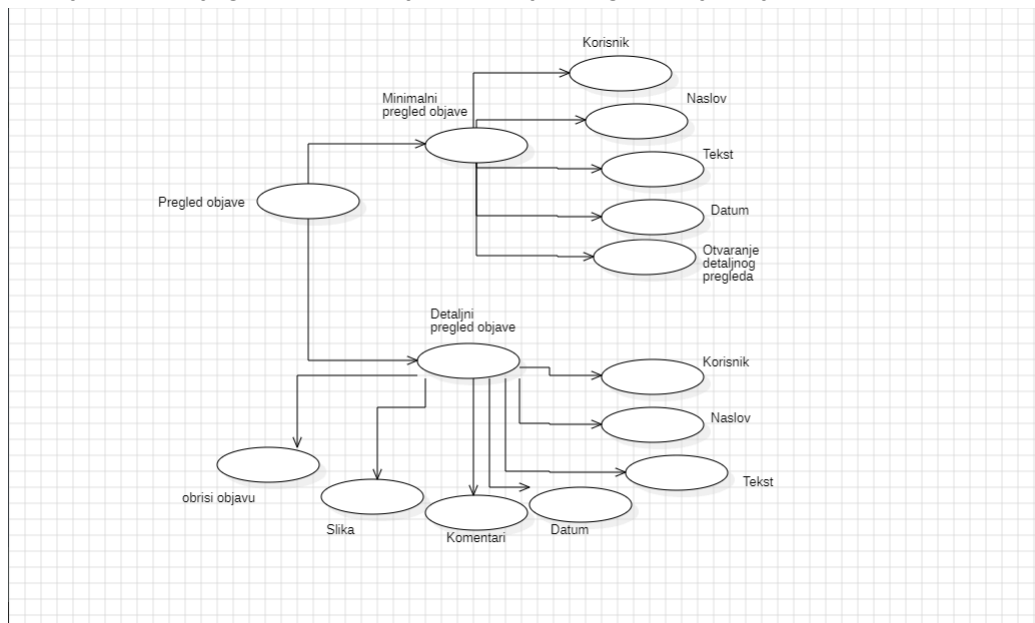
6.1. Dijagrami slučajeva korišćenje

Osnovni UML dijagram koji prikazuje korisnike i slučajeve korišćenja RešiMe portala prikazan je na slici:

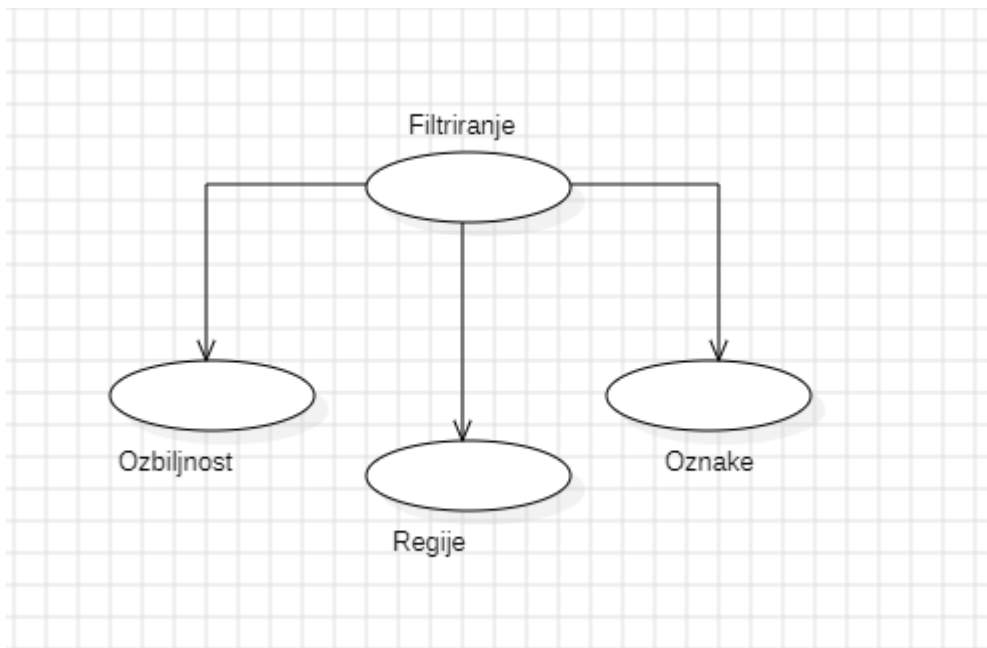


Slučajevi korišćenja pregled informacija — Pregled objave, Filtriranje objava i Pregled mape — obuhvataju složenije radnje koje se mogu dalje razložiti na pojedinačne slučajeve korišćenja.

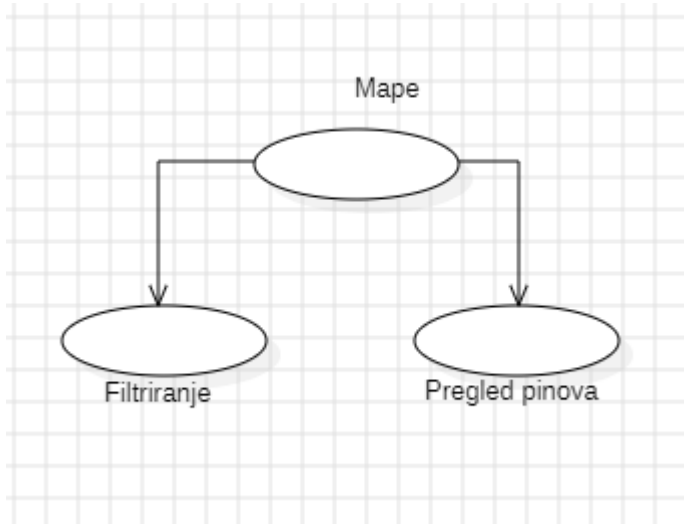
Detaljni UML dijagram za slučaj korišćenja pregled objave je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja filtriranje objava je prikazan na sledećoj slici:



Detaljni UML dijagram za slučaj korišćenja prikazivanje mapa je prikazan na sledećoj slici:



6.2. Kratak opis slučajeva korišćenja

6.2.1. Kreiranje objave

Kratak opis: Prijavljeni korisnik popunjava formu za prijavu problema, unosi naslov, opis, sliku, lokaciju, kategoriju i nivo ozbiljnosti, čime kreira novu objavu dostupnu ostalim korisnicima.

Akteri koji iniciraju slučaj korišćenja: Korisnik (građanin).

6.2.2. Brisanje objave

Kratak opis: Autor objave ili administrator otvaranjem detaljnog pregleda briše prijavu, nakon čega ona prestaje da bude vidljiva ostalim korisnicima.

Akteri koji iniciraju slučaj korišćenja: Korisnik (vlasnik objave), Administrator.

6.2.3. Pretraga objava

Kratak opis: Prijavljeni korisnik unosi ključnu reč u polje za pretragu i dobija spisak objava čiji naslov, opis ili tagovi sadrže traženi izraz.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.4. Pregled objava

6.2.4.1. Minimalni pregled objave

Kratak opis: Prijavljenom korisniku se na glavnoj stranici prikazuje spisak postojećih objava u sažetom obliku: korisničko ime autora, naslov, prvi redovi teksta i datum objavljivanja; klikom na bilo koju objavu otvara se detaljan pregled.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.4.2. Detaljni pregled objave

Kratak opis: Korisniku koji otvori pojedinačnu objavu prikazuju se svi dostupni podaci: korisničko ime autora, naslov, kompletan tekst, datum objave, priložena slika, postojeći komentari i dugme za dodavanje novih i dugme „Obriši“ (ukoliko je korisnik vlasnik objave ili administrator).

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.5. Filtriranje objava

Kratak opis: Korisnik može da suzi spisak objava biranjem filtera po nivou ozbiljnosti, geografskoj regiji ili unapred definisanim oznakama (tagovima); sistem u realnom vremenu prikazuje samo one objave koje ispunjavaju sve izabrane kriterijume.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.6. Pregled mape

Kratak opis: Korisnik otvara stranicu sa interaktivnom mapom na kojoj su geolocirani pinovi svih objava; može primeniti filtere (ozbiljnost, regija, oznake) da bi prikazao samo odgovarajuće pinove, a klikom na pin otvara detaljan pregled izabrane prijave.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.7. Postavljanje komentara

Kratak opis: U okviru detaljnog pregleda objave prijavljeni korisnik upisuje tekst komentara i potvrđuje ga klikom na dugme „Comment“ ili pritiskom tastera Enter; komentar se odmah prikazuje ispod objave i vidljiv je svim korisnicima.

Akteri koji iniciraju slučaj korišćenja: Korisnik.

6.2.8. Kreiranje naloga

Kratak opis: Novi posetilac unosi korisničko ime, e-mail adresu i lozinku u registracionu formu i potvrđuje je klikom na dugme „Submit“; sistem kreira profil i omogućava mu dalje korišćenje portala.

Akteri koji iniciraju slučaj korišćenja: Posetilac sajta.

6.2.9. Rešavanje problema

Kratak opis: Administrator u detaljnom pregledu objave menja njen status (npr. „u toku“, „rešeno“), čime se ažurira stanje prijave i obaveštavaju svi korisnici o napretku.

Akteri koji iniciraju slučaj korišćenja: Administrator.

6.2.10. Dodavanje role

Kratak opis: Administrator iz spiska registrovanih korisnika bira pojedinca i dodeljuje mu novu ulogu (npr. moderator), čime mu proširuje skup funkcija dostupnih na portalu.

Akteri koji iniciraju slučaj korišćenja: Administrator.

7. Pregled na logičku arhitekturu sistema

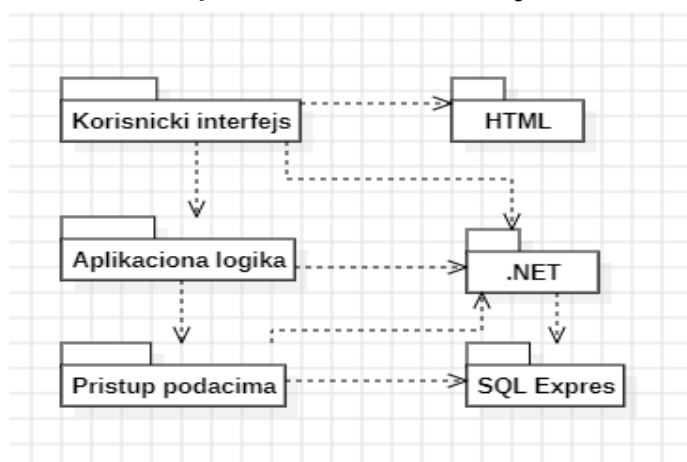
U ovom odeljku dat je pregled logičke arhitekture RešiMe portala. Pogled obuhvata najznačajnije klase, njihovu organizaciju u pakete i podsisteme, kao i raspored u slojeve. Dinamički aspekti prikazani su kroz opise realizacije ključnih slučajeva korišćenja, a veze između arhitekturno značajnih komponenti ilustrovane su odgovarajućim UML dijagramima klasa.

Logički pogled na RešiMe portal obuhvata tri glavna paketa:

- Korisnički interfejs – razvijen pomoću Angular frameworka, HTML, SCSS i TypeScript-a. Sadrži komponente za prikaz objava, mapu, filtere, forme za prijavu i registraciju.
- Aplikaciona logika – implementirana u .NET okruženju, sadrži kontrolere, servise i DTO objekte koji obrađuju poslovnu logiku i komunikaciju između klijenta i baze podataka.
- Pristup podacima – ostvaren kroz SQL Server Express bazu podataka i .NET repozitorijume, omogućava čuvanje, ažuriranje i preuzimanje podataka o korisnicima, objavama, komentarima, regijama i oznakama.

Ovi paketi organizovani su u troslojnu arhitekturu: sloj korisničkog interfejsa, sloj aplikacione logike i sloj pristupa podacima, čime se postiže jasna razdvojenost odgovornosti i laka održivost sistema.

7.1. Pregled arhitekture - organizacija paketa i podsistema u slojeve



7.1.1. Korisnički interfejs

– sloj

Ovaj sloj realizuje korisnički interfejs RešiMe portala. Razvijen je pomoću Angular frameworka, HTML, SCSS i TypeScript tehnologija. Sadrži sve komponente, servise i šablone potrebne za prikaz objava, mapu, filtere, forme za kreiranje naloga i prijavu problema, kao i za komunikaciju sa serverskim delom aplikacije.

Sloj korisničkog interfejsa zavisi od sloja aplikacione logike (.NET API), koji obrađuje poslovnu logiku i omogućava pristup podacima.

7.1.2. Aplikaciona logika

– sloj

Sloj aplikacione logike je srednji sloj u troslojnoj arhitekturi RešiMe portala. Implementiran je pomoću .NET platforme i sadrži kontrolere, servise i DTO objekte koji obrađuju poslovnu logiku, vrše validaciju podataka i upravljaju tokom slučajeva korišćenja. Ovaj sloj omogućava komunikaciju između korisničkog interfejsa (Angular) i sloja za pristup podacima (SQL Server Express).

Sloj aplikacione logike zavisi od sloja za pristup podacima i .NET okruženja.

7.1.3. Pristup podacima

– sloj

Sloj za pristup podacima nalazi se na dnu troslojne arhitekture RešiMe portala.

Implementiran je pomoću SQL Server Express baze podataka i .NET repozitorijuma koji vrše sve operacije pribavljanja, dodavanja, ažuriranja i brisanja podataka. Ovaj sloj obezbeđuje perzistenciju podataka o korisnicima, objavama, komentarima, regijama i oznakama.

Sloj za pristup podacima ne zavisi od drugih slojeva, ali je zavisn od SQL Server Express i .NET okruženja.

7.1.4. .NET

tehnologija

.NET pruža runtime i biblioteke za izgradnju RESTful Web API-ja koji obrađuje poslovnu logiku, autorizaciju, validaciju i upravljanje podacima između klijenta i baze.

7.1.5. SQL Server Express

DBMS

SQL Server Express je relacioni sistem za upravljanje bazama podataka koji čuva sve podatke o korisnicima, objavama, komentarima, regijama i oznakama te omogućava brz i siguran pristup podacima putem .NET Entity Framework-a.

7.1.6. Angular

tehnologija

Angular je front-end okvir koji upravlja prikazom podataka, rutiranjem, validacijom formi i komunikacijom s .NET API-jem, realizujući sav korisnički interfejs RešiMe portala.

7.1.6.1. HTML

tehnologija

HTML definiše semantičku strukturu Angular komponenti i omogućava prikaz teksta, slika, mapa i formi za unos podataka u klijentskom pregledaču.

7.1.6.2. SCSS

tehnologija

SCSS proširuje CSS sintaksu varijablama, ugnježđivanjem i miks-inima te određuje izgled, raspored i responsivnost svih elemenata korisničkog interfejsa RešiMe portala.

7.1.6.3. TypeScript

tehnologija

TypeScript pruža strogu tipizaciju i moderne konstrukcije jezika za razvoj Angular komponenti, servisa i modela, čime se povećava održivost i čitljivost klijentskog koda.

8. Pogled na procese

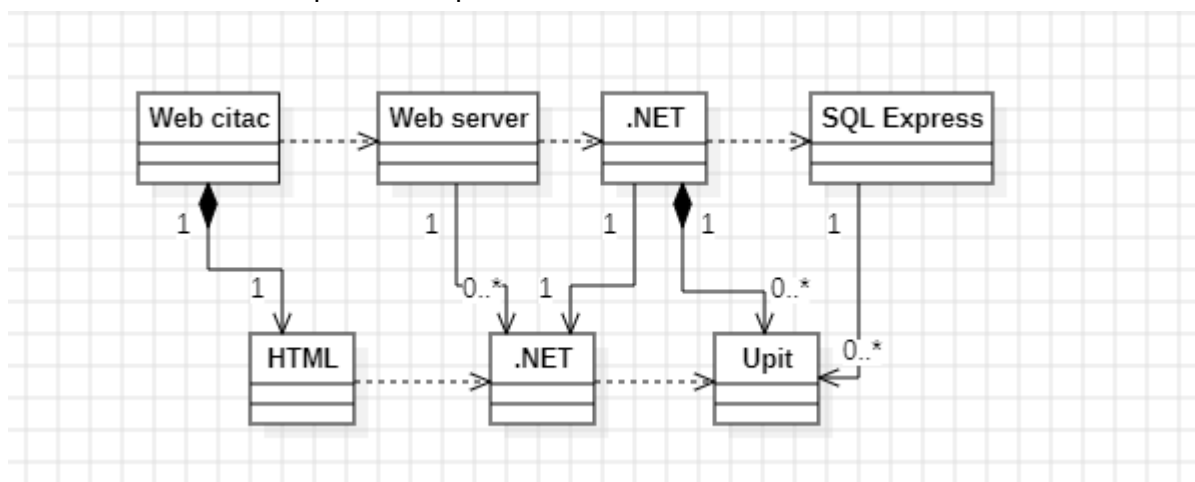
U ovom odeljku je sadržan pogled na procesnu arhitekturu sistema. Ovaj opis treba da sadrži specifikaciju različitih zadataka (procesu i niti) uključenih u rad sistema. Takođe je potrebno dati dijagrame koji pokazuju njihovu interakciju i konfiguraciju. Dodela objekata i klasa na određene zadatke takođe spada u opis procesne arhitekture.

Web aplikacije zasnovane na .NET-u imaju relativno jednostavan procesni model koji je u potpunosti pod kontrolom Web servera. Sa stanovišta projektanta .NET Web aplikacije nije potrebno voditi računa o načinu rada Web servera i načinu izvršavanja skripti.

Ilustracije radi u nastavku je dat opis procesa uključenih u izvršenje RešiMe portala kao Web aplikacije.

- Procesi

Na sledećem UML dijagramu klasa prikazani su procesi koji učestvuju u izvršenju RešiMe portala. Dijagram je opšteg tipa i može se primeniti na bilo koju Web aplikaciju zasnovanu na .NET-u i SQL Server Express bazi podataka.



8.1. Web čitač

Web čitač je proces koji izvršava funkcionalnost aplikacije za prikaz HTML stranica dobijenih od nekog Web servera. U najopštijem slučaju Web čitač u jednom trenutku može da

prikazuje samo jednu HTML stranicu.

Web čitač zavisi od Web servera koji generiše i vraća odgovarajuću HTML stranicu na zahtev.

8.2. Web server

Web server je proces koji izvršava funkcionalnost opsluživanja zahteva prispelih sa više Web čitača. Ukoliko je zahtevana stranica .NET Core, Web server inicira izvršenje .NET procesa koji obrađuje odgovarajući skript i generiše sadržaj koji se vraća čitaču. Web server može paralelno da inicira veći broj .NET procesa.

8.3. .NET

.NET proces obavlja posao obrade zadatog .NET Core i generiše odgovarajući tekstualni sadržaj koji Web server šalje Web čitaču. Za izvršenje .NET Core ovaj proces može da zahteva usluge SQL Server Express servera-a. Komunikacija između .NET procesa i SQL Server Express servera se obavlja preko prosleđivanja upita i vraćanja rezultat.

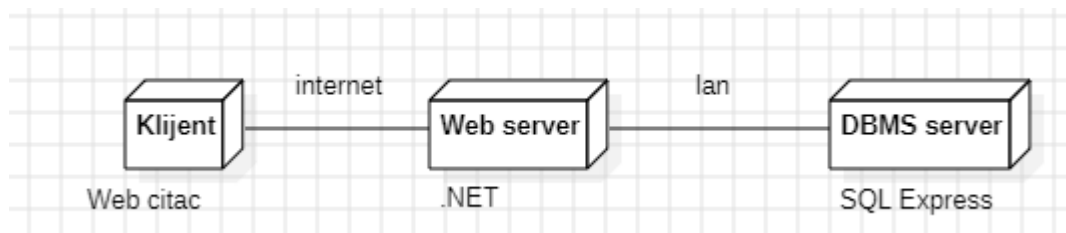
8.4. SQL Server Express

DBMS server je računar na kome se izvršava SQL Server Express proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Zbog sigurnosti podataka koji se na ovom računaru čuvaju pristup bazi je ograničen samo na računare iz lokalne mreže (LAN).

9. Pogled na rasporedjivanje sistema

Pogled na raspoređivanje sistema prikazuje različite fizičke čvorove za najopštiju konfiguraciju sistema. Fizičkim čvorovima koji predstavljaju procesore vrši se dodeljivanje identifikovanih procesa.

Na sledećoj slici dat je UML dijagram raspoređivanja RešiMe portala.



9.1. Klijent

Pristup RešiMe portal se obavlja preko klijentskih računara na kojima se izvršava Web čitač. Za povezivanje između klijenta i Web servera koristi se Internet infrastruktura tako da nema ograničenja u pogledu lokacije klijenta.

9.2. Web server

Računar na kome se izvršava Web server opslužuje više klijenata koji pristupaju preko Interneta. Pored osnovnog procesa koji realizuje funkcionalnost Web servera, na ovom računaru mogu da se izvršavaju i procesi .NET koji vrše obradu zadatih .NET Core. U najopštijoj konfiguraciji DBMS se izvršava na posebnoj mašini koja je sa Web serverom u lokalnoj mreži (LAN).

9.3. DBMS

DBMS server je računar na kome se izvršava SQL Server Express proces koji realizuje funkcionalnost sistema za upravljanje bazama podataka. Zbog sigurnosti podataka koji se na ovom računaru čuvaju pristup bazi je ograničen samo na računare iz lokalne mreže (LAN).

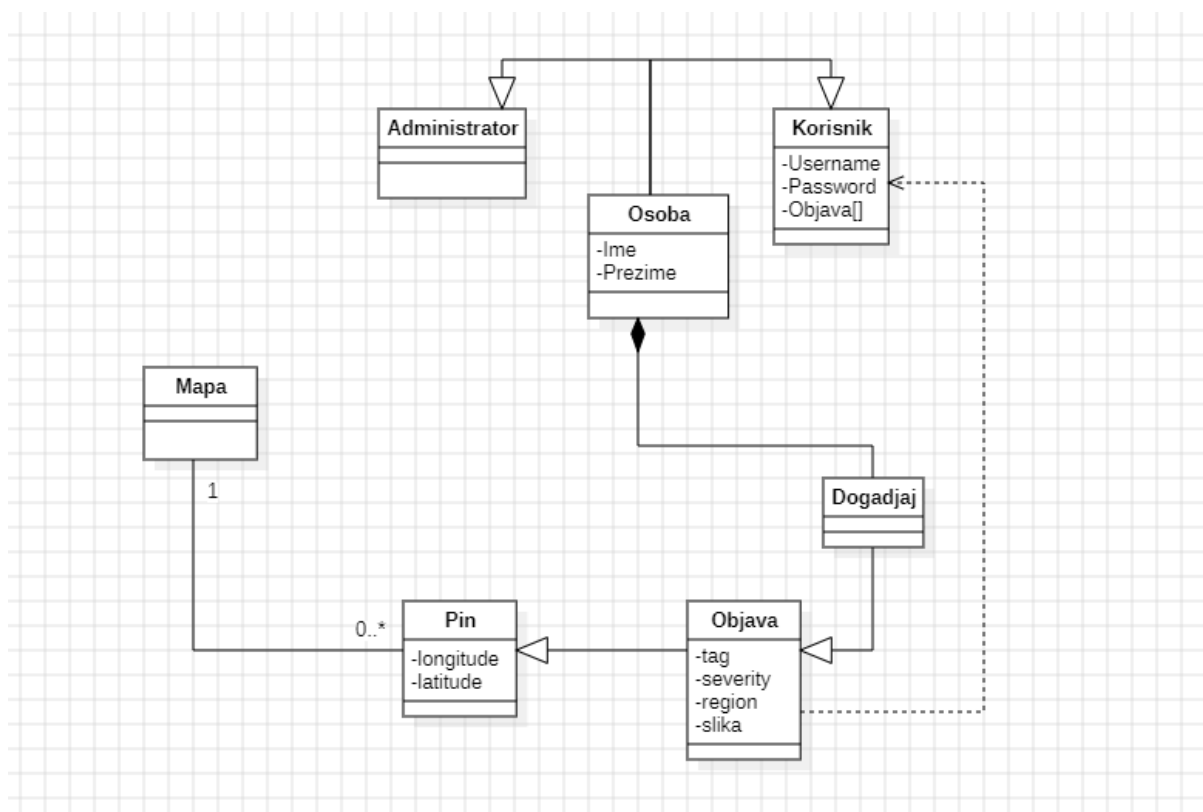
10. Pogled na implementaciju sistema

Pogled na implementaciju prikazuje različite aspekte bitne za implementaciju sistema. U slučaju RešiMe portala ovaj odeljak sadrži model domena, šemu baze podataka i prikaz komponenti sistema razvrstanih u ranije identifikovane pakete.

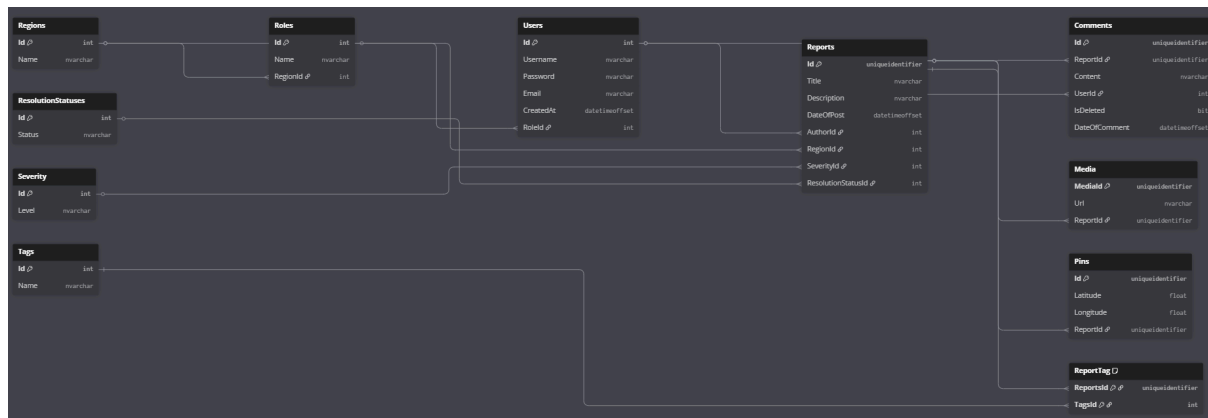
10.1. Model domena

Model domena za koji se RešiMe portal projektuje je ilustrovan UML dijagramom klasa. U njemu su prikazane domenske klase, neki od njihovih atributa, kao i veze koje se mogu identifikovati između njih.

Model domena predstavlja osnovu za projektovanje baze podataka, ali i identifikaciju nekih od komponenti koje će biti implementirane.



10.2. Šema baze podataka



11. Performanse

Izabrana arhitektura softvera podržava zahteve u pogledu broja korisnika koji mogu simultano pristupati sistemu i vremena odziva za pristup bazi podataka specificirane u zahtevima u pogledu performansi [5]:

1. Sistem će da podrži do 1000 simultanih pristupa korisnika portalu.
2. Vreme potrebno za pristupanje bazi podataka u cilju izvršenja nekog upita ne sme da bude veće od 5 sekundi.

Zahtevane performanse su zadovoljene izborom tehnologija na kojima će sistem biti razvijen i definisane hardverske platforme [5].

12. Kvalitet

Izabrana arhitektura softvera podržava zahteve u pogledu dostupnosti i srednjeg vremena između otkaza specificirane u zahtevima u pogledu pouzdanosti [5]:

1. RešiMe portal će biti dostupan 24 časa dnevno, 7 dana u nedelji. Vreme kada portal nije dostupan ne sme da pređe 10%.
2. Srednje vreme između dva sukcesivna otkaza ne sme da padne ispod 120 sati.