MACROP - Manageable and Collaborative Resources Of Project

Millan Milićević 15234
Stefan Milovanović 15246

Sadržaj

Opis projekta

MACROP aplikacija omogućuje lakše upravljanje projektom, tako što nudi mogućnosti planiranja i kolaboracije neophodnih resursa jednog projekta.

Arhitekturni zahtevi

Funkcionalnosti sistema

Sistem će se sastojati od sledećih funkcionalni:

1. Registorvanje korisnika sistema, gde je moguće imati različite uloge, npr. vođa projekta, vođa tima, dizajner, web developer…
2. Kreiranje projekta i dodavanje članova projektu, kao i organizacija članova po timovima
3. Dodavanje i dodeljivanje zaduženja na nivou pojedinačnog člana, kao i na nivou tima ili celokupnog projekta
4. Mogućnost upravljanja zaduženjima (izmena, brisanje, dodavanje specifičnosti)
5. Razmenjivanje poruka(chat messages)
6. Postojanje worksheet-ova na nivou projekta i na nivou timova

Nefunkcionalni zahtevi sistema

Sistem takođe poseduje i nefunckionalne zahteve i to su:

1. **Pristupačnost** – web aplikacija ne zavisi od platforme na kojoj se izvršava
2. **Dostupnost** – pošto se aplikacija nalazi na serverima koji garantuju **dosupnost** 24/7, te će naša palikacija biti dostupna uvek svim korisnicima
3. **Sigurnost** – svaki korisnik ima svoj nalog i niko drugi sem njega nema pristup tim podacima

4. **Preformanse** – pošto se ceo prikaz izvršava na klinetskoj strani, a komunikacija sa serverom je minimalna, obrada zahteva je veoma brza
5. **Modifikabilnost** – zbog slabo spregnutih servisa, lako je dodati servis ili izmeniti postojeći
6. **Interoperabilnost** – zbo postojanja REST API-ja moguće je lakše integrisati sa nekom drugom komponentom

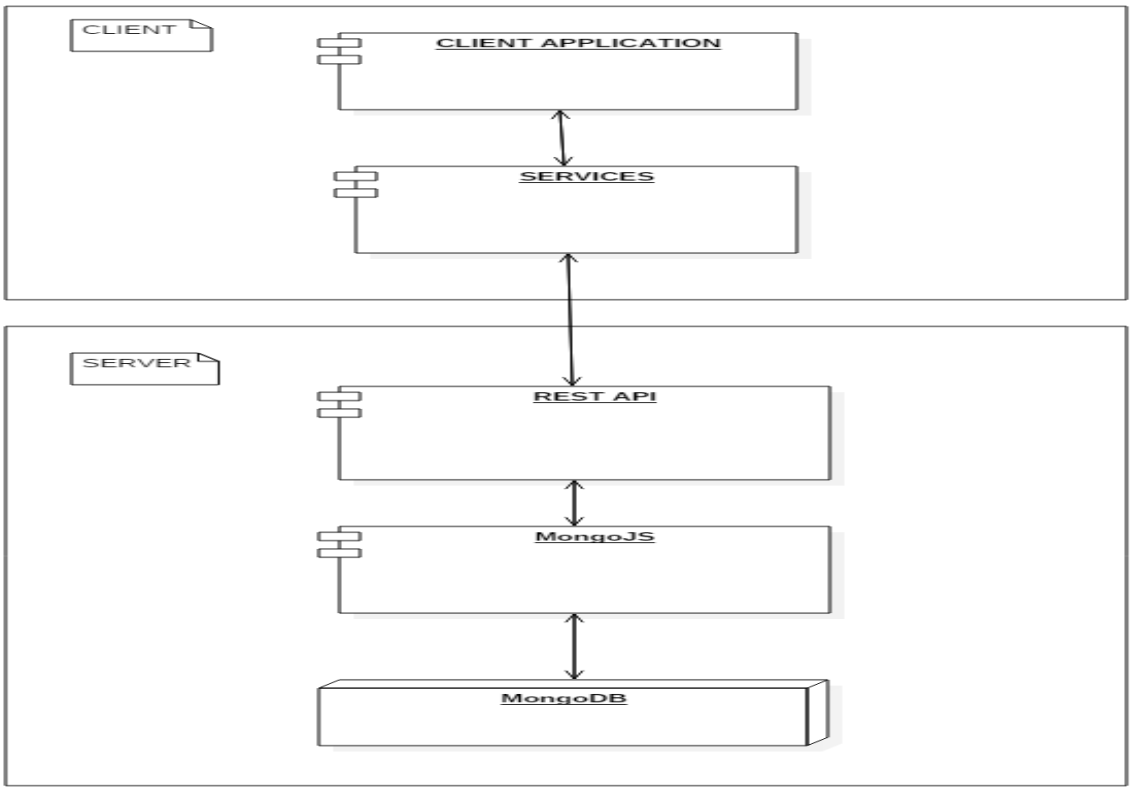
Arhitekturni obrasci

Aplikacija će koristiti sledeće obrasce:

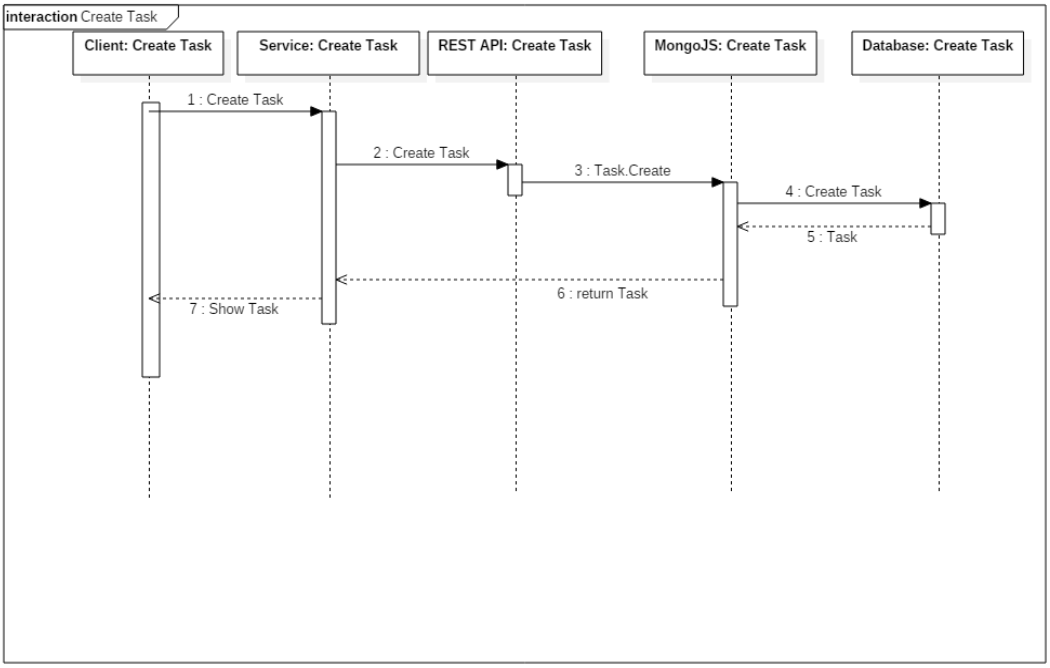
- **Layerd** – naša aplikacija će se sastojati od prezistencionog sloja gde se nalazi baza podataka, servera gde se nalazi biznis model podataka i sa kojim komunicira klijent preko API-ja, a celokupna prezentaciona logika će se nalaziti na klijentu sa svim servisima
- **Broker** – aplikacija će pomoću bibiloteka i framework-a asinhorno komunicirati sa serverom i pretplaćivaće se na određene događaje, na koje će server pri svakoj promeni emitovati tu peomenu svim stranama koje su se pretplatile
- **MVC** – pored toga aplikacija u sebi sadrži i MVC obrazac tako što se na klijentu nalaze kontorleri, modeli i pogledi za prikaz podataka. Modele popunjava komunikacijom sa serverom, dok manipulacijom nad prikazom mu dozvoljavaju kontorleri. Dok gledano sa serverske strane model podataka je biznis model, kontroler je REST API, dok je cela klijentska aplikacija pogled

Arhitekturni pogled

Strukturni pogled

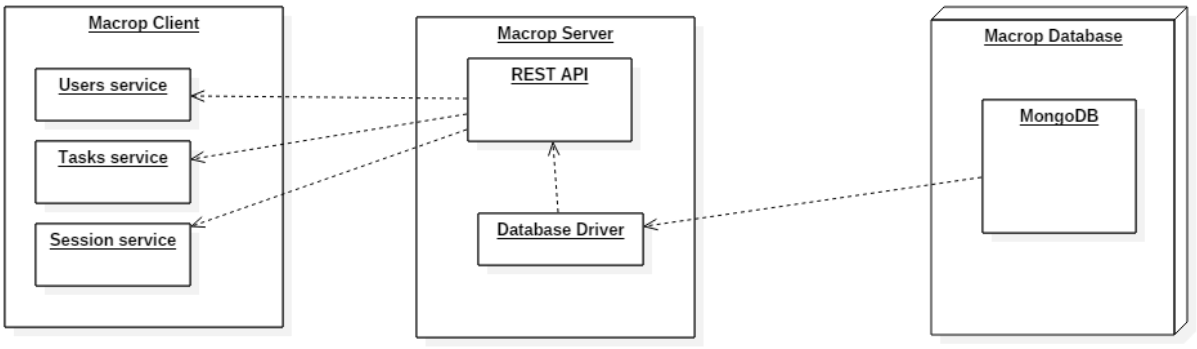


Bihevioralni pogled



Millan Milićević 15234
Stefan Milovanović 15246

Alokacioni pogled



Framework-ovi i biblioteke

Angular 4

Angular je JS framework koji je sam po sebi nudi component-based i mvc arhitekturu. Omogucava razvoj Single Page apliakcija, sto prakticno znaci da se komunikacija sa serverom svodi na minimum. Posto se isntalira preko npm (Node Package Manager), shodno tome dostupan je veliki broj biblioteka na koriscenje. Angular 2 je uveo typescript, tipiziarani javascript, odnosno ecmascript 6, tako da je razvoj web aplikacija veoma blizak C# programiranju. Velika popularnost ecmascript-a kao i typescript-a omogucava funkcionalno programiranje, jer je podrska za rad sa kolekcijama veoma siroka, kao i za asinhronu komunikaciju poput promisa, observabla itd.

NodeJS

Iskorišćen je NodeJS zbog veoma lakog razvoja web aplikacija, jer poseduje mnoštvo biblioteka i freamwork-ova koji mu omogućavaju jednostavnije kreiranje samog servera, asinhronu akomunikaciju sa klijentom kao i sa bazom.

Express

Express framework je okvir za NodeJS koji omogućava lakše pokretanje servera i već generiše gotove komponente na serveru i praktično pravi MVC aplikaciju podeljenu u module, a pored toga uvozi sve neophodne osnovne biblioteka kao što su http, upravljanje file sistemom, komunikacija sa klijentom…

SocketIO

SocketIO je bibliotak za NodeJS koja služi za veoma jednostavnu asinhornu komunikaciju sa klijentom otvaranjem socketa koji su implementirani u njoj, i pogodna je za stvari kao što je chatovanje.

MongoJS

MongoJS je driver za MongoDB bazu podataka, i on radi lako pretraživanje i preslikavanje u biznis domen. Takođe ima ugrađene osnove CRUD operacije i pogodan je za jednostavne upite ka bazi.

MongoDB

MongoDB je document baza podataka i iskoristili smo tu prednost da pribavljamo i smestamo sve podatke u JSON formatu radi lakšeg korišćenja i kasnijeg prevođenja u objekte. Takođe postoje dobro dokumentovane biblioteka koje mogu da se koriste u sprezi sa NodeJS-om kao što je gore navedeno.

Odlučili smo se za ove tehnologije zbog toga što sve pružaju rad u Javascript jeziku te nije potrebno prelaziti sa jednog na drugi jezik, i takođe postoji dobra sprega izmedju svih komponenti ovih sistema te je ravoj lakši i pogodniji za developere.