**2Order**

Projekat iz predmeta Arhitektura i projektovanje softvera

Studenti:

Nikola Vidanović, 16507

Uroš Ilić, 16612

Sadržaj

[Kontekst i cilj projekta 2](#_Toc59993877)

[Arhitekturni zahtevi 2](#_Toc59993878)

[Arhitekturno značajni slučajevi korišćenja (glavni funkcionalni zahtevi) 2](#_Toc59993879)

[Nefunkcionalni zahtevi (atributi kvaliteta) 4](#_Toc59993880)

[Tehnička i poslovna ograničenja 4](#_Toc59993881)

[Arhitekturni dizajn 4](#_Toc59993882)

[Generalna arhitektura 5](#_Toc59993883)

[Strukturni pogledi 5](#_Toc59993884)

[Bihevioralni pogledi 7](#_Toc59993885)

[Implementaciona pitanja 8](#_Toc59993886)

[Analiza arhitekture 8](#_Toc59993887)

[Potencijalni rizici u implementaciji i strategije prevazilaženja 8](#_Toc59993888)

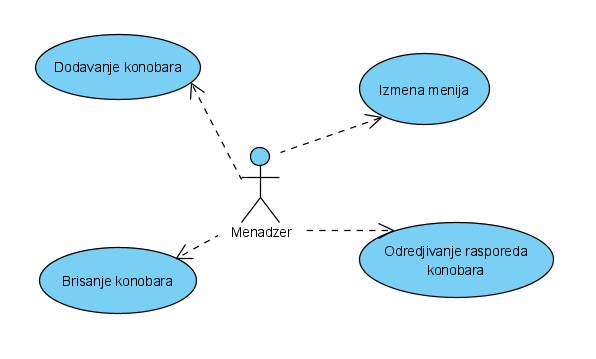
# Kontekst i cilj projekta

Cilj ovog projekta jeste da olakša proces poručivanja u ugostiteljskom objektu. To bi značilo da konobari ne treba da prilaze stolovima kako bi primili porudžbinu. Ideja je da svako od gostiju za istim stolom ima pristup virtualnom meniju iz koga može da izabere neku od ponuđenih stavki. Virtualni meni biće zajednički za sve goste koji su za istim stolom, a nakon odabira formira se zajednička porudžbina koja će biti prosleđena konobaru, koji je odgovoran za taj sto. Aplikacija takođe pruža i uslugu dostave na kućnu adresu, pri čemu korisnici aplikacije mogu zajedno da kreiraju porudžbinu sa svojih uređaja. Nakon odabira se formira zajednički porudžbina i prosleđuje ugostiteljskom objektu. Korisnici u oba slučaja mogu da izaberu način plaćanja. Korisnici će imati mogućnost i da zajedno napišu recenziju o ugostiteljskom objektu. Takođe jedan od glavnih ciljeva jeste da u svakom trenutku pruži konobarima evidenciju o stanju stolova u ugostiteljskom objektu.

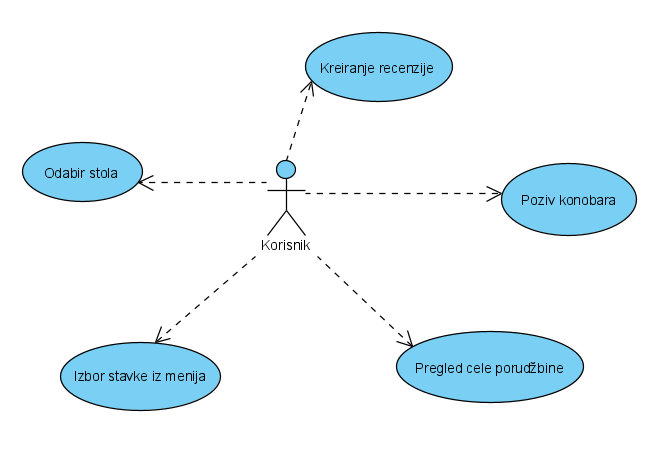
# Arhitekturni zahtevi

## Arhitekturno značajni slučajevi korišćenja (glavni funkcionalni zahtevi)

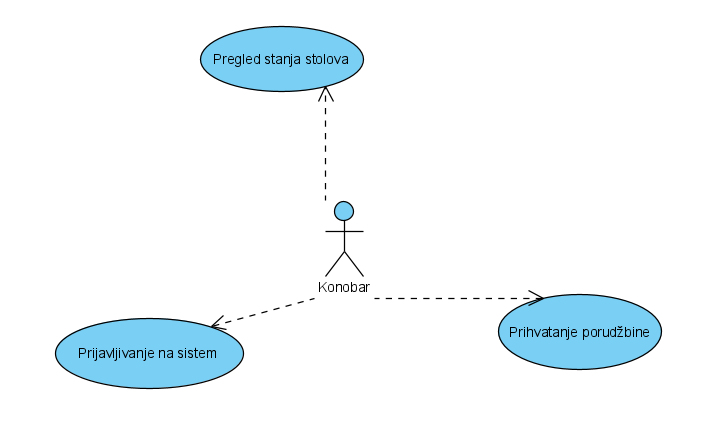
* Prijavljivanje konobara na sistem – Konobar ima mogućnost da se uz pomoć svojih podataka prijavi na sistem.
* Pregled stanja stolova – Konobar može u svakom trenutku da proveri porudžbine stolova za koje je on zadužen.
* Prihvatanje porudžbine – Konobar obaveštava korisnika da je porudžbina prihvaćena, nakon što mu je korisnik prosledi.
* Odabir stola – Prilikom ulaska u ugostiteljski objekat, kako bi dobio mogućnost da kreira porudžbinu, korisnik mora da izabere svoj sto. Ukoliko korisnik izabere sto koji je već zauzet, mora da unese šifru koja se dobija prilikom zauzimanja stola.
* Izbor stavke iz menija – Svaki korisnik koji je izabrao svoj sto može da izabere stavku iz menija koja predstavlja deo celokupne porudžbine.
* Pregled cele porudžbine – Svaki korisnik koji je “prijavljen” na istom stolu ima mogućnost da vidi celokupnu porudžbinu koju su sastavili svi korisnici za tim stolom.
* Poziv konobara za plaćanje – Svi korisnici imaju mogućnost, a dovoljno je da jedan od njih izabere način plaćanja i ukoliko je potrebno pozove konobara.
* Kreiranje recenzije – Svi korisnici imaju mogućnost da napišu komentar o usluzi u ugostiteljskom objektu. Kreiranje recenzije biće kolaborativno, što znači da svaki od korisnika može da napiše svoj utisak o ugostiteljskom objektu.
* Izmena menija – Menadžer ugostiteljskog objekta ima mogućnost da u svakom trenutku promeni ponudu tako što će dodati novu stavku u meniju, izbrisati ili promeniti postojeću.
* Dodavanje konobara – Menadžer ugostiteljskog objekta treba da angažuje konobara za rad u ugostiteljskom objektu i odobri mu pristup aplikaciji.
* Brisanje konobara – Menadžer ugostiteljskog objekta treba da zabrani pristup konobaru ukoliko je konobar otpušten.
* Određivanje rasporeda konobara po stolovima – Menadžer ugostiteljskog objekta treba da svakom konobaru dodeli sto za koji je konobar zadužen, kako bi posao konobara bio ravnomerno raspoređen.



Slika 1 – Slučajevi korišćenja Menadžera



Slika 2 – Slučajevi korišćenja Korisnika



Slika 3 – Slučajevi korišćenja Konobara

## Nefunkcionalni zahtevi (atributi kvaliteta)

* Pouzdanost – Svaka porudžbina treba da bude prosleđena menadžeru i odgovarajućem konobaru i procesirana.
* Dostupnost – Sistem mora da bude 99% vremena dostupan.
* Sigurnost – Odnosi se na bezbednost prilikom online plaćanja.
* Portabilnost – Sistem treba da radi na svim novijim verzijama web-čitača. Aplikacija treba da bude responsive, jer će korisnici uglavnom koristiti mobilne telefone.
* Jednostavnost – Korišćenje aplikacije treba da bude intuitivno, što podrazumeva da će korisnici moći da koriste aplikaciju bez dodatnog uputstva.

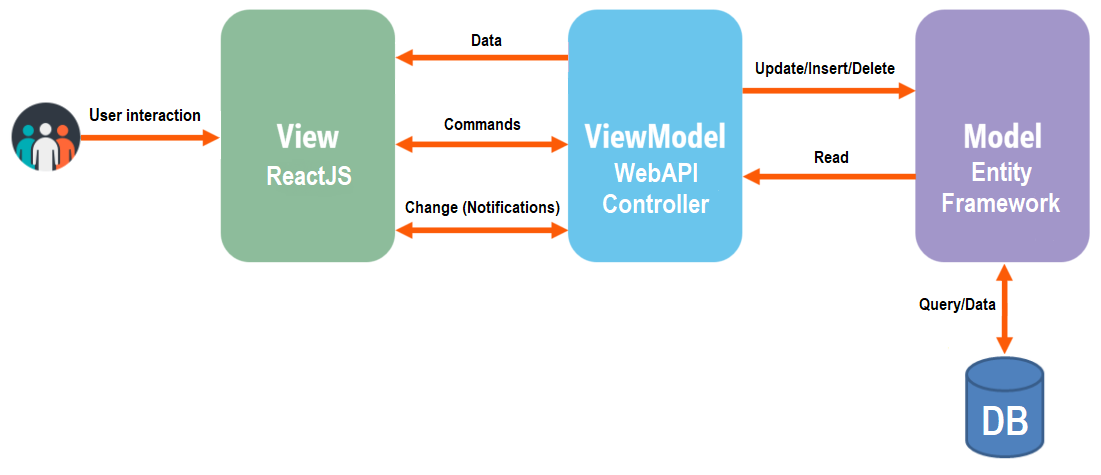
## Tehnička i poslovna ograničenja

Sistem neće zahtevati nabavljanje novog hardvera i oslanjaće se na besplatna softverska rešenja, tako da neće zahtevati kupovinu dodatnog softvera.

# Arhitekturni dizajn

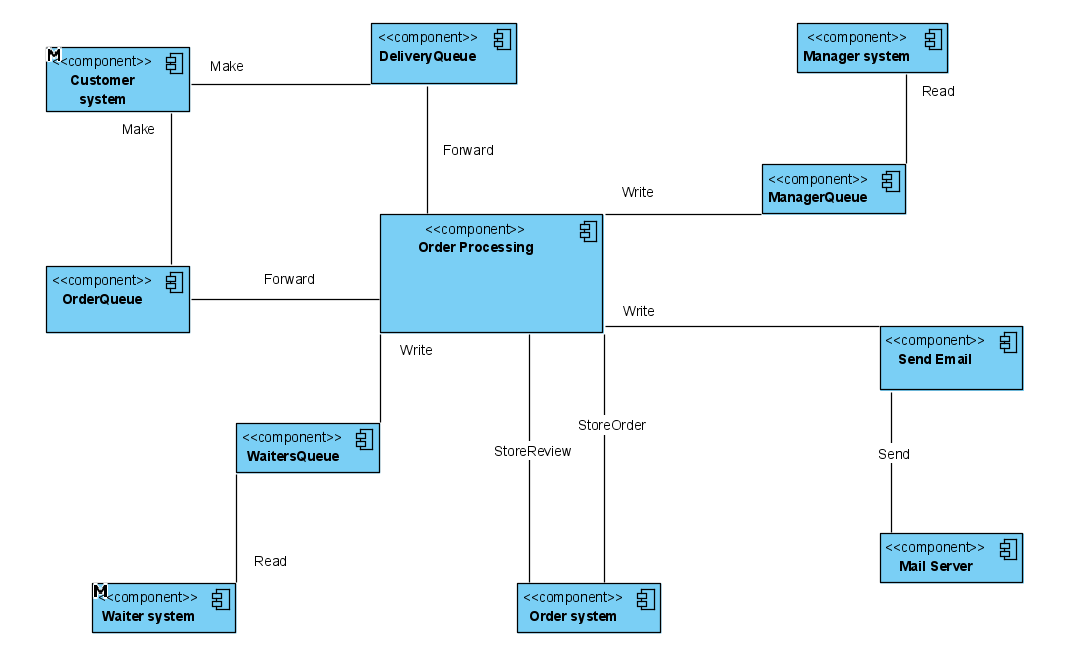
U izradi ove aplikacije koristićemo sledeće arhitekturne obrasce: Layered, MVVM, Repository, Publish-Subscriber.

MVVM: Arhitektura aplikacije biće uređena po MVVM obrascu, pri čemu će se na serverskoj strani nalaziti RESTFul API koji će pružati usluge klijentskoj strani. U ovom obrascu View komponenta predstavlja deo aplikacije sa kojom klijent interaguje i biće implementirana pomoću ReactJS framework-a. Odgovarajućim komandama View će preuzimati podatke iz ViewModel-a, odnosno iz kontrolerskih klasa(slanjem get, put, post, delete HTTP zahteva). Komponenta ViewModel, koju predstavljaju WebApi kontroleri, preuzima podatke iz modela i formatira ih kako bi se iskoristili za prikazi u View-u. Model sadrži domenske klase i upravlja njihovom perzistencijom u bazi podataka, slanjem upita ka bazi podataka.



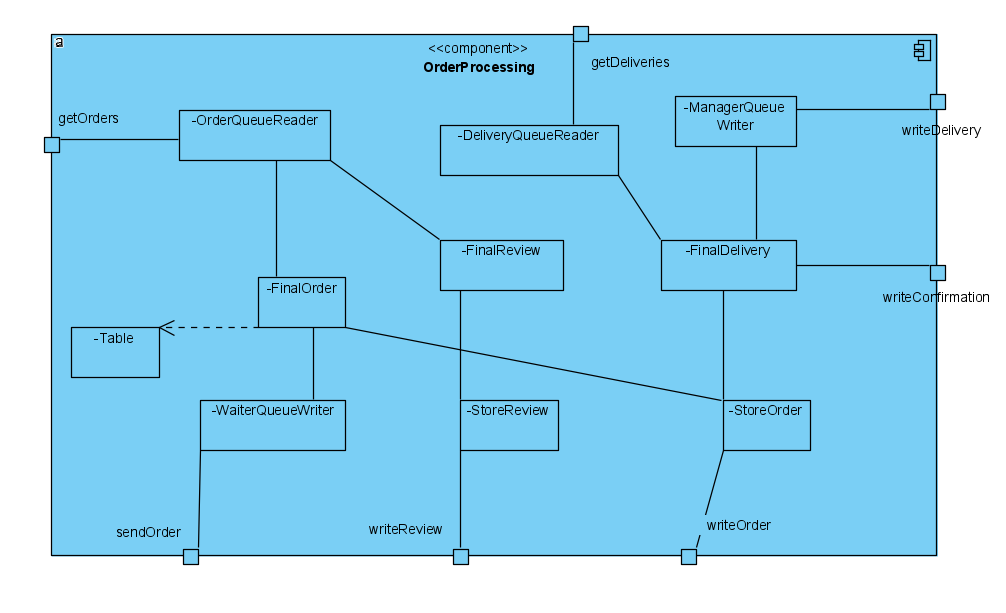
Slika 4 – MVVM arhitektura

## Generalna arhitektura

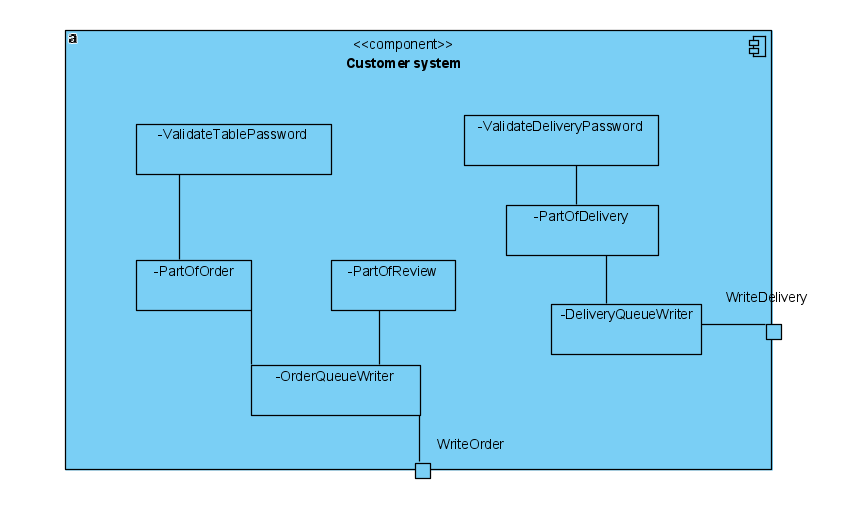


Slika 5 – Dijagram komponenti: Glavna arhitektura sistema

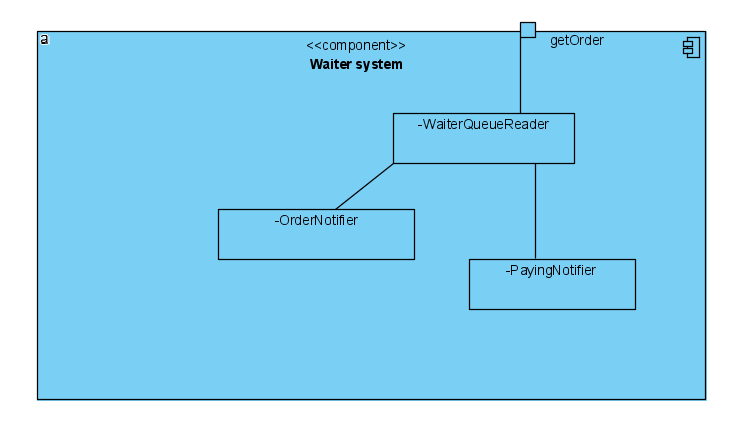
## Strukturni pogledi



Slika 6 – Dijagram komponenti: Komponenta Order Processing



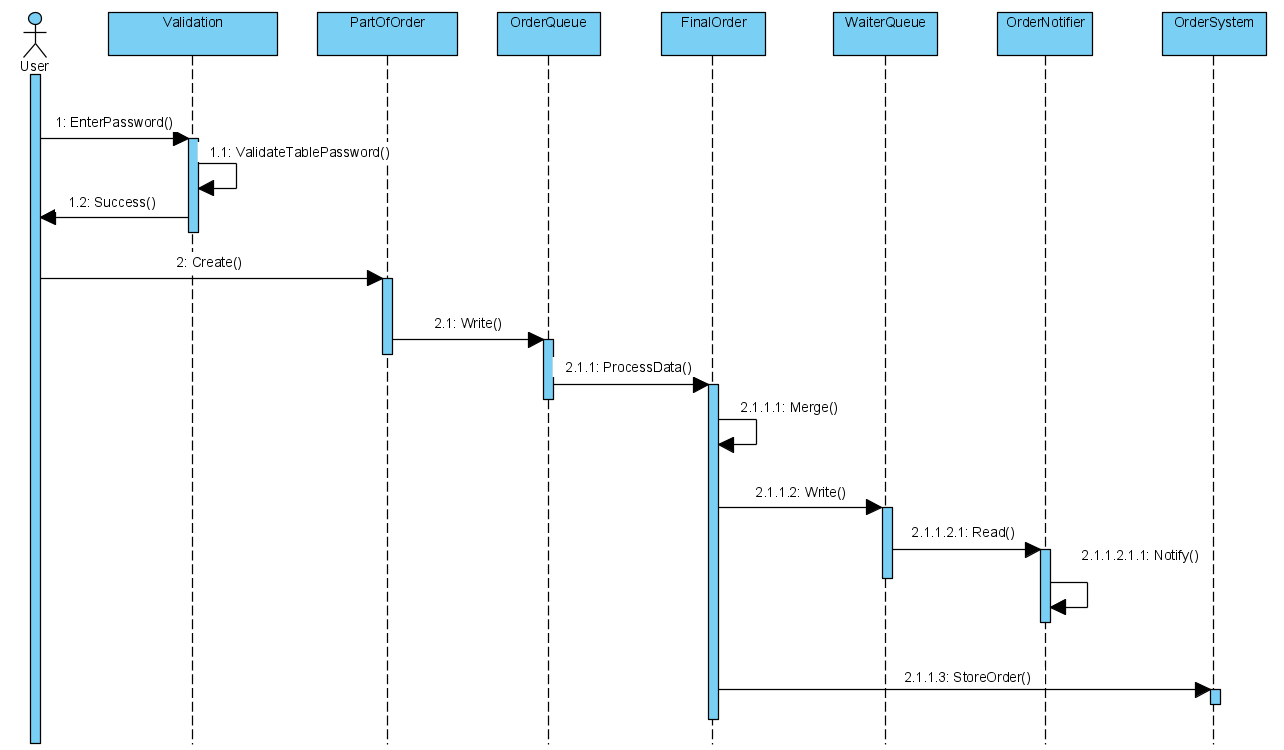
Slika 7 – Dijagram komponenti: Komponenta Customer system



Slika 8 – Dijagram komponenti: Komponenta Waiter system

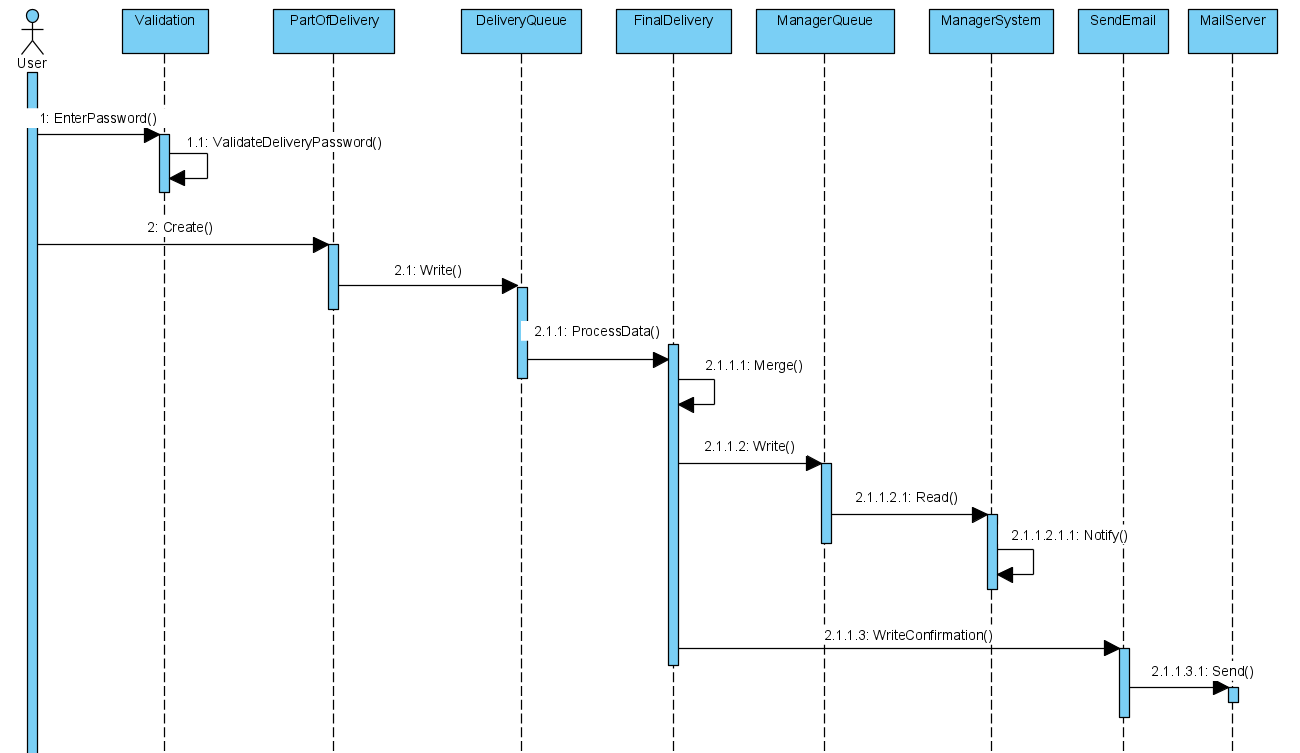
## Bihevioralni pogledi

Kreiranje porudžbine:



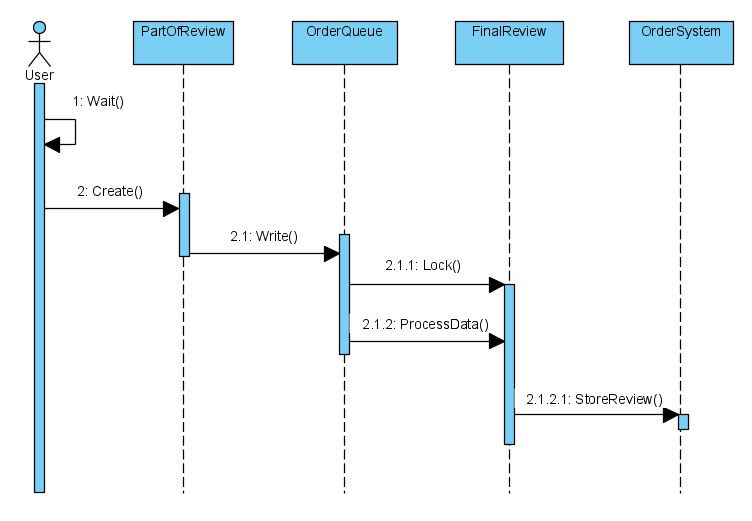
Slika 9 – Sekvencijalni dijagram: Kreiranje porudžbine

Kreiranje dostave:



Slika 10 – Sekvencijalni dijagram: Kreiranje dostave

Kreiranje recenzije:



Slika 11 – Sekvencijalni dijagram: Kreiranje recenzije

## Implementaciona pitanja

U implementaciji ove aplikacije koristićemo sledeće biblioteke, framework-e i komponente:

.NET Framework, React, RabbitMQ, Entity Framework Core, MySQL baza podataka.

# Analiza arhitekture

## Potencijalni rizici u implementaciji i strategije prevazilaženja

Aplikacija nema potencijalnih rizika.