

Arhitekturni projekat

PlanIT Teams

Članovi tima:

Ana Stojanović 15905
Damjan Trifunović 15948
Milica Todorović 15937

Naziv tima:

Anonymous Mink

Sadržaj

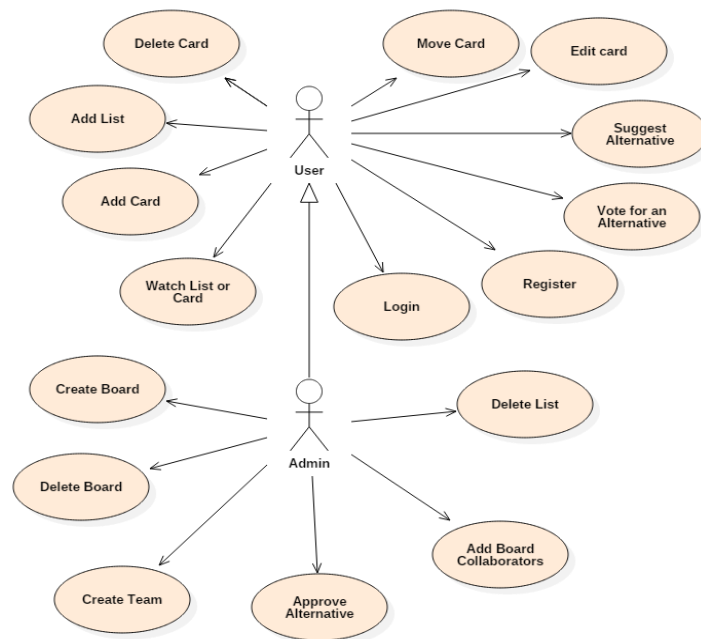
Domen problema.....	3
Funkcionalni zahtevi	3
Arhitekturni zahtevi	4
Nefunkcionalni zahtevi	4
Arhitekturni dizajn	4
<i>Strukturni pogled</i>	<i>4</i>
<i>Bihevioralni pogled</i>	<i>5</i>
<i>Alokacioni pogled</i>	<i>6</i>
Arhitekturni obrasci.....	6
<i>Layered</i>	<i>6</i>
<i>Publish-subscribe</i>	<i>7</i>
<i>Broker</i>	<i>7</i>
<i>MVVM (Model – View – View Model)</i>	<i>7</i>
Specifikacija biblioteka I programskih okvira.....	8

Domen problema

PlanIT Teams je desktop aplikacija namenjena timovima koji sarađuju na nekom projektu. Član tima kreira „kanban“ tablu i daje pristup ostalim članovima, tako da mogu svi da kreiraju kartice za određene zadatke, da ih premeštaju u različite liste na tabli i da predlažu drugačije prioritete zadataka (alternative datih listi).

Funkcionalni zahtevi

- Povezivanje klijenta i servera
- Kreiranje korisničkog naloga i login
- Kreiranje timova (grupisanje korisnika u timove)
- Kreiranje kanban board-a
- Dodavanje timova ili drugih korisnika kao saradnike na *svoj* board
- Dodavanje i izmena lista i kartica (u okviru datih listi) na datu board
- Označavanje (watch) odgovarajuće liste ili kartice koje korisnik želi da prati (dobija in- app notifikaciju kada dodje do neke izmene na datoj listi ili kartici)
- Predlaganje alternativne liste i glasanje od strane korisnika za date alternative. Klijenti koji prate datu listu bivaju obavešteni o aktivnom glasanju
- Omogućavanje kolaboracije saradnika na zajedničkom board-u
- Sistem notifikacije zainteresovanih saradnika o promenama na posmatranim listama
- Skladištenje podataka



Use case dijagram sistema

Arhitekturni zahtevi

- Sistem će biti dostupan 7 dana nedeljno, 24 sata dnevno
- Potrebno je da šema baze podataka bude skrivena
- Potrebno je omogućiti trenutnu notifikaciju o događajima

Nefunkcionalni zahtevi

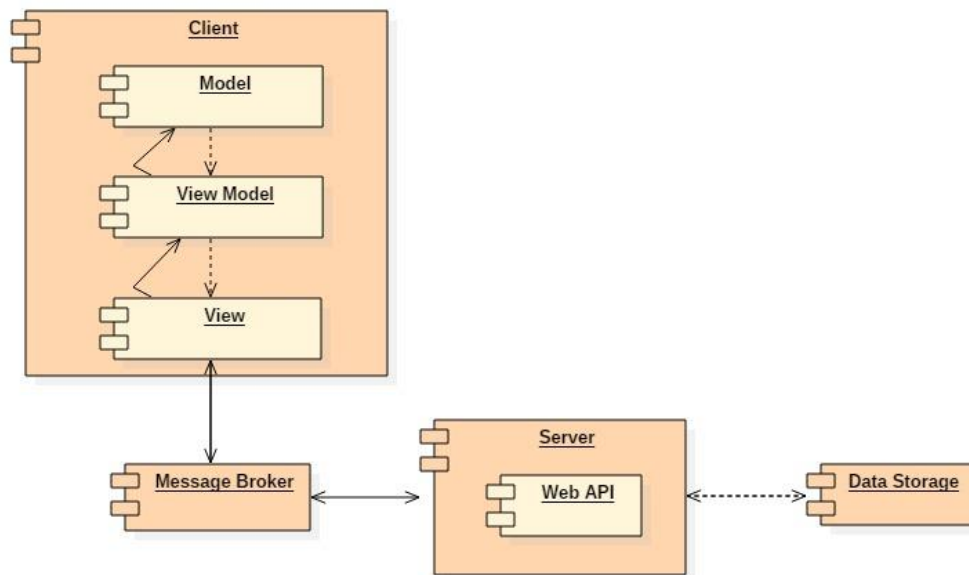
- Pouzdanost
- Performanse – potrebno je smanjiti vreme odziva i prilagoditi propusnu moć sistemu
- Dostupnost – potrebno je da je aplikacija bude dostupna 24/7
- Modifikabilnost – potrebno je omogućiti relativno laku promenu sistema
- Skalabilnost – potrebno je da aplikacija može da podrži povećanje broja korisnika
- Lakoća korišćenja – potrebno je da aplikacija bude intuitivna i jednostavna za korišćenje

Arhitekturni dizajn

Strukturni pogled

Navedeni dijagram ilustruje strukturu sistema navodeći komponente sistema kao i njihovu međusobnu povezanost.

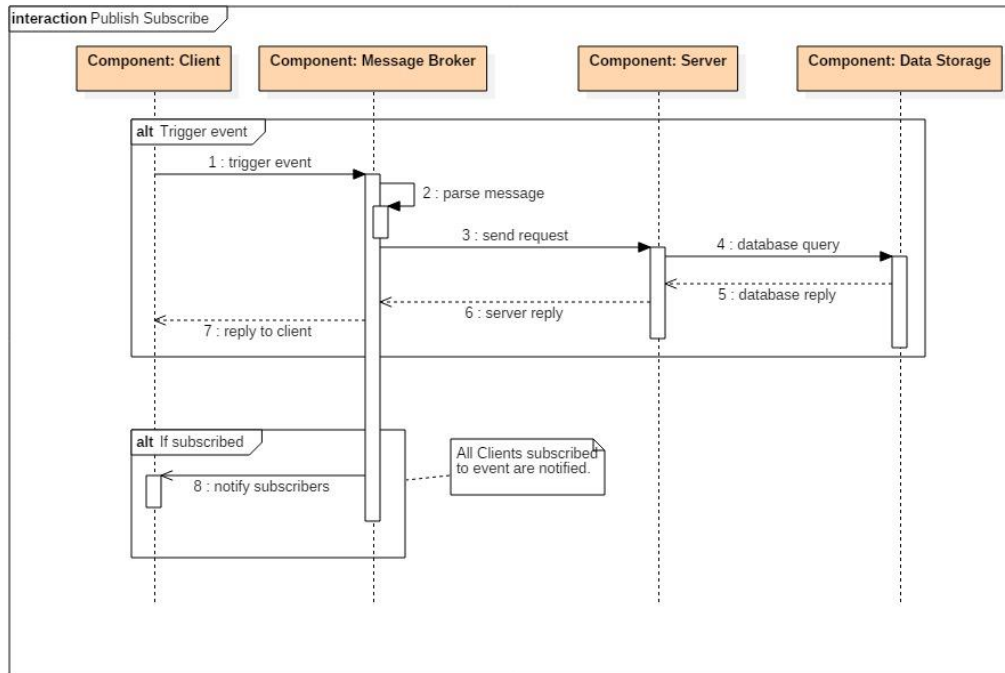
Struktura klijenta zasnovana je na *MVVM* projektnom obrascu. Komunikacija između klijentske i serverske aplikacije ostvarena je korišćenjem *Message Broker*-a. Server predstavlja *Web API*, preko serverske aplikacije se ostvaruje komunikacija sa odgovarajućim *Data Storage*-om.



Dijagram strukture sistema

Bihevioralni pogled

Navedeni dijagram ilustruje bihevioralni pogled jednog događaja u sistemu u opštem slučaju. Klijent okida događaj i ostvaruje se konekcija sa brokerom, koji raspakuje poruku, i kontaktira server. Server vrši obradu zahteva, i smešta/čita podatke iz baze po potrebi. Zatim server šalje odgovor brokeru koji ga dalje prosleđuje kljentu. Broker zatim obaveštava sve klijente koji su subscribe-ovani na dati događaj.

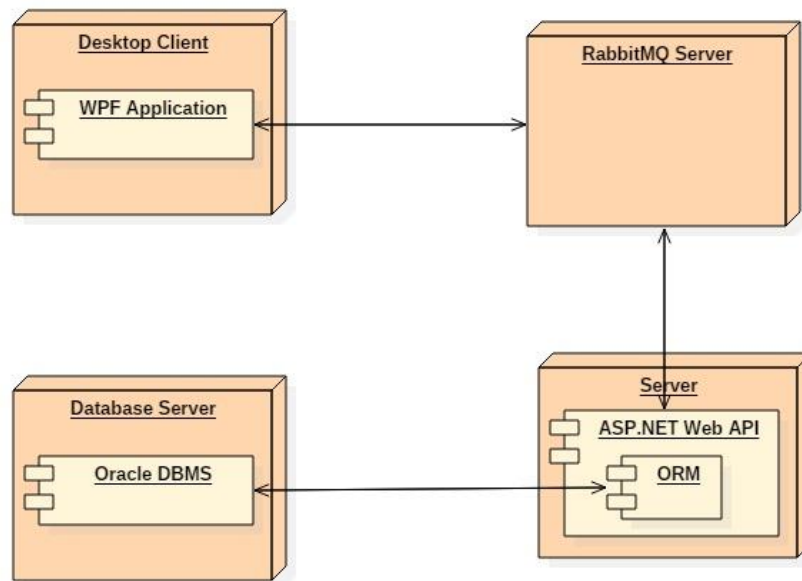


Dijagram sekvence sistema

Alokacioni pogled

Navedeni dijagram ilustruje alokacioni pogled na sistem odnosno raspored prethodno opisanih komponenti na čvorovima (računarima).

Klijentska aplikacija nalazi se na računaru sa Windows operativnim sistemom, kao *WPF aplikacija*. Komunikacija između klijentske i serverske aplikacije ostvaruje se korišćenjem *RabbitMQ Servera*. Serverska aplikacija realizuje se kao *ASP.NET Web API*, dok se kao database server koristi *Oracle DBMS*.



Dijagram raspoređivanja sistema

Arhitekturni obrasci

Layered

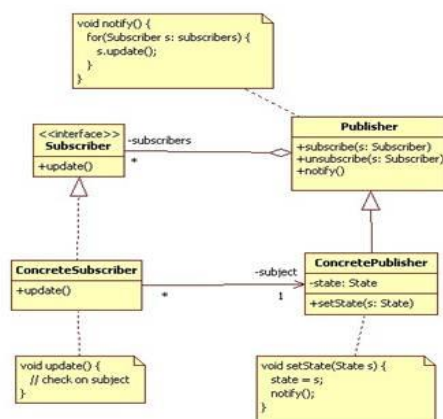
PlanIT Teams sistem će implementirati Layered (Slojeviti) arhitekturni obrazac.

Sistem će se sastojati iz četiri sloja: klijentska aplikacija, message broker, serverska aplikacija i baza podataka(data storage). Komunikacija između klijentske i serverske aplikacije ostvaruje se preko message broker-a. Serverska aplikacija biće API (realizovan pomoću *RESTful* principa) i omogućiće komunikaciju sa odgovarajućim DBMS-om koristeći dati ORM alat za mapiranje domenskih entiteta.

Publish-subscribe

PlanIT Teams sistem će implementirati Publish-subscribe arhitekturni obrazac (ovaj obrazac je sadržan u okviru *Message Broker* komponente).

Svaki board ima određen broj kolaboratora. Kada dodje do bilo koje promene u board-u, svim kolaboratorima je vidljivo da je došlo do promene. Kolaboratori mogu da se subscribe-uju na određene liste ili kartice u okviru board-a. U tom slučaju, subscribe-ovanim učesnicima stižu in-app notifikacije kojim se posebno naglašava da je došlo do promene na listama koje prate.



Dijagram publish-subscribe

Broker

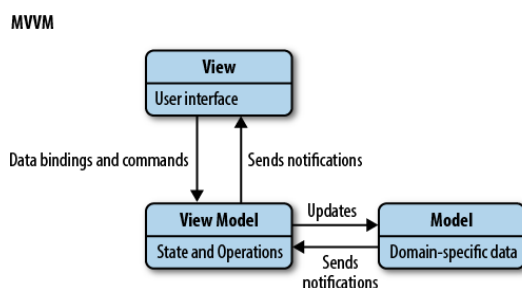
PlanIT Teams sistem će implementirati broker arhitekturni obrazac.

Broker komponenta predstavlja posrednika između klijentske i serverske aplikacije. Klijentska aplikacija šalje zahtev brokeru, koji dalje kontaktira odgovarajući server upućujući mu zahtev, zatim server vrši obradu zahteva i vraća rezultat brokeru koji dati rezultat presleđuje nazad klijentu. Broker obrazac biće realizovan korišćenjem *message broker* komponente.

MVVM (Model – View – View Model)

PlanIT Teams sistem će implementirati MVVM obrazac na strani klijentske aplikacije.

MVVM obrazac je modifikacija (varijanta) standardnog MVC obrazca. PlanIT Teams developeri su istraživanjem došli do zaključka da je MVVM najpogodniji obrazac za razvoj WPF aplikacije (klijentske aplikacije datog sistema).



Dijagram MVVM

Specifikacija biblioteka I programskih okvira

Windows Presentation Foundation (WPF) – Podsystem .NET frejmworka za izradu Windows klijentskih aplikacija

RabbitMQ .NET Client Library – Biblioteka koja omogućava komunikaciju između RabbitMQ server i .NET klijentske aplikacije

RabbitMQ Server – Message broker, implementira redove poruka i njemu se obraćaju korisnici redova

ASP.NET Web API aplikacija – Serverska aplikacija

Entity Framework – Objektno-relacioni mapper (ORM alat)

Oracle DBMS – Baza podataka