

Dokumentacija

Projekat iz predmeta OOAD

Grupa: 10

Tim: TimSMTH

Demonstrator: Jasmina Bajramović

Članovi:

- Mašović Haris
- Salihić Samra
- Šabović Dženana

Sadržaj

Opis teme	3
Funkcionalnosti	3
Dizajn projekta (dijagrami).....	4
Use Case dijagrami.....	4
Dijagram klasa (MVVM)	7
Dijagrami aktivnosti	8
Dijagrami sekvenci i komunikacije	12
Dijagrami komponenti, paketa i raspoređivanja	15
Izvještaj o radu	16
Tabela očekivanih i implementiranih funkcionalnosti	19

Opis teme

Vicinor¹ je aplikacija koja omogućava korisniku (user-u) da na jednostavan način pronađe mjesto/objekat koje servira/prodaje hranu kao npr. fast food/restorani itd. pri čemu korisnik bira objekat koji mu se sviđa iz liste ponuđenih objekata.

Često korisnik dolazi u situaciju gdje ne može odlučiti gdje jesti, te pomoći ove aplikacije ima izbor restorana/fast food-ova (generalno food places) u određenom radijusu u odnosu na trenutnu lokaciju. Korisnik može izabrati određeni restoran i/ili dodati u listu svojih omiljenih restorana ponuđene restorane od strane aplikacije ili restorane u odnosu na trenutnu lokaciju definisane određenim radijusom kojeg je korisnik zadao.

Nakon toga aplikacija daje upustva kako doći na određenu lokaciju koju je korisnik izabrao.

Funkcionalnosti

Korisnik:

- Mogućnost izbora/promjene radijusa pretrage restorana u odnosu na lokaciju korisnika
- Mogućnost pregleda restorana u vidu liste
- Mogućnost izbora restorana
- Mogućnost dobijanja uputa do odabranog restorana
- Mogućnost dobijanja broja za poziv restorana/objekta, prilikom izbora restorana
- Mogućnost izbora iz skupa restorana preporučenih od strane aplikacije

Registrirani korisnik:

- Mogućnost uređivanje korisničkog profila
- Mogućnost pravljenja liste želja (lista omiljenih restorana)

Neregistrirani korisnik:

- Mogućnost registracije

Administrator:

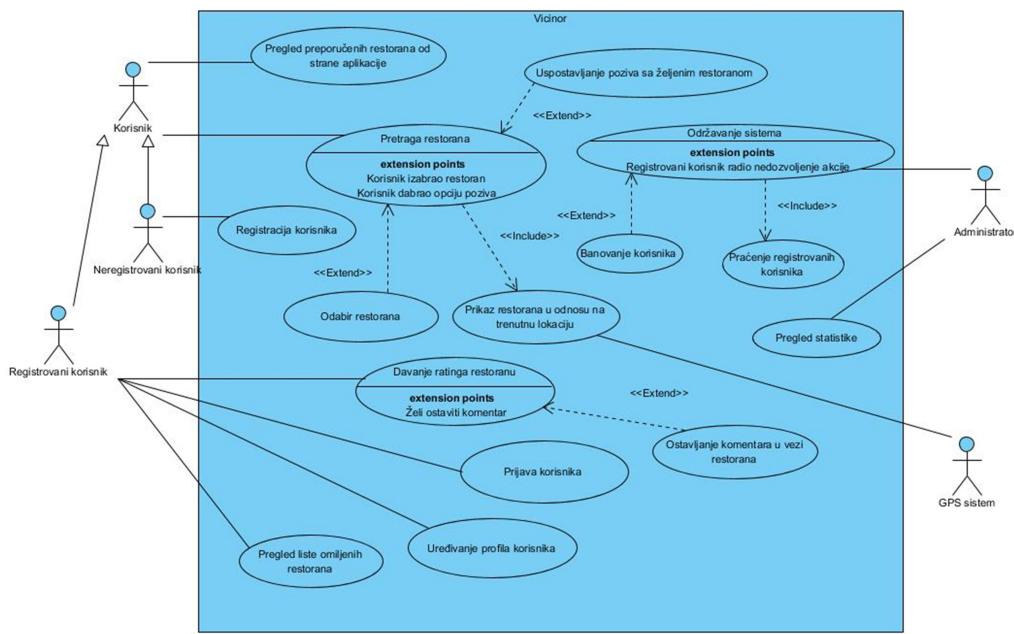
- Prikaz statistike
- Praćenje registrovanih korisnika sistema te banovanje korisnika po potrebi
- Uređivanje profila

¹ vicino - ital. znači nearby/close - blizu, a r kao restorani).

Dizajn projekta (dijagrami)

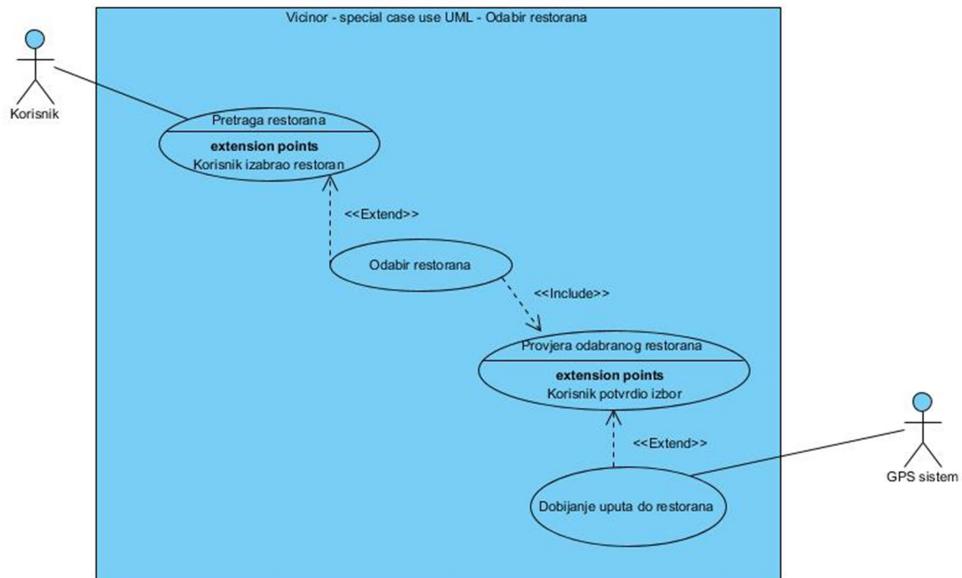
Use Case dijagrami

Na slici je prikazan glavni Use case dijagram:

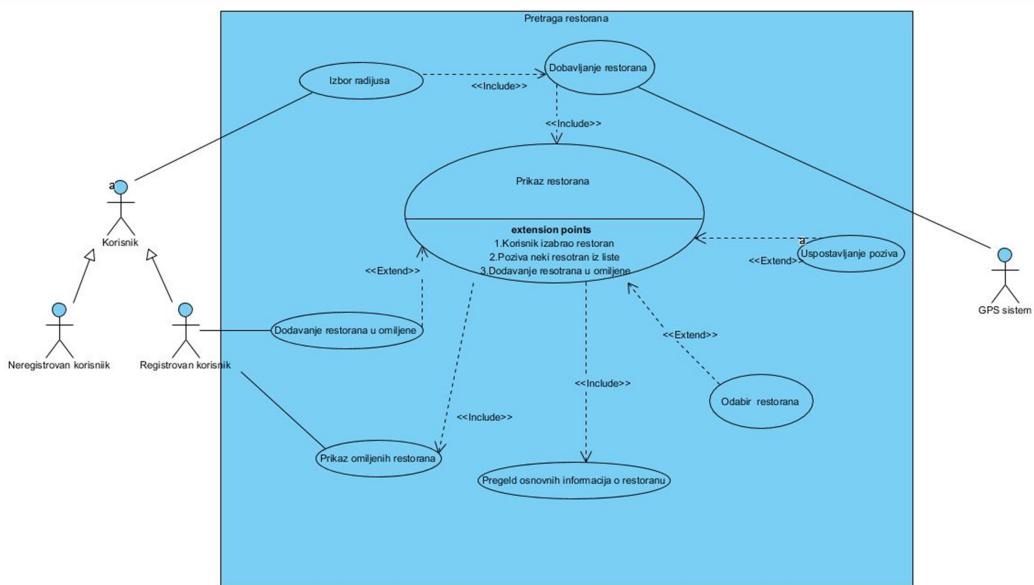


Slika 1. Vicinor Use Case dijagram

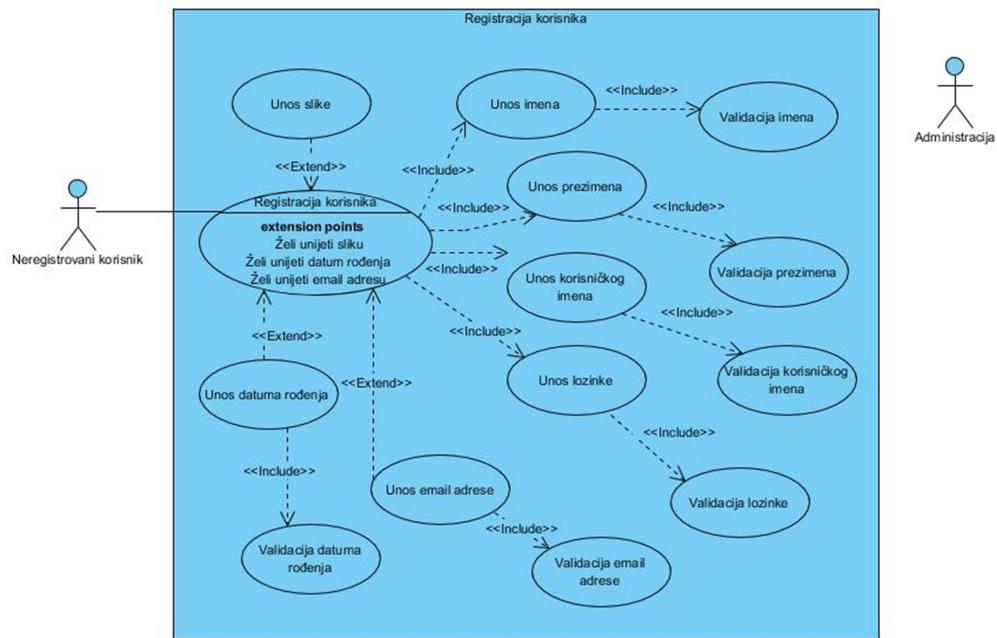
Na sljedećim slikama su prikazani use case dijagrami za odabir restorana, pretragu restorana i registraciju korisnika:



Slika 2. Odabir restorana

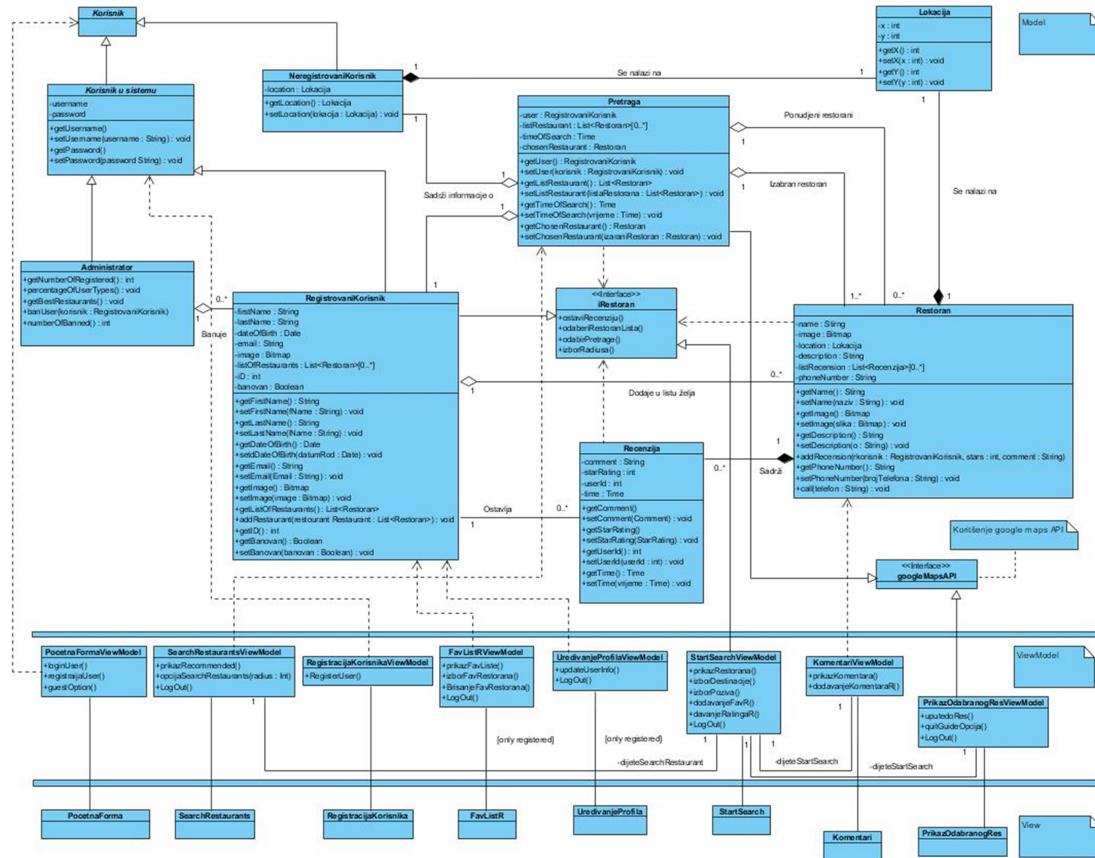


Slika 3. Pretraga restorana



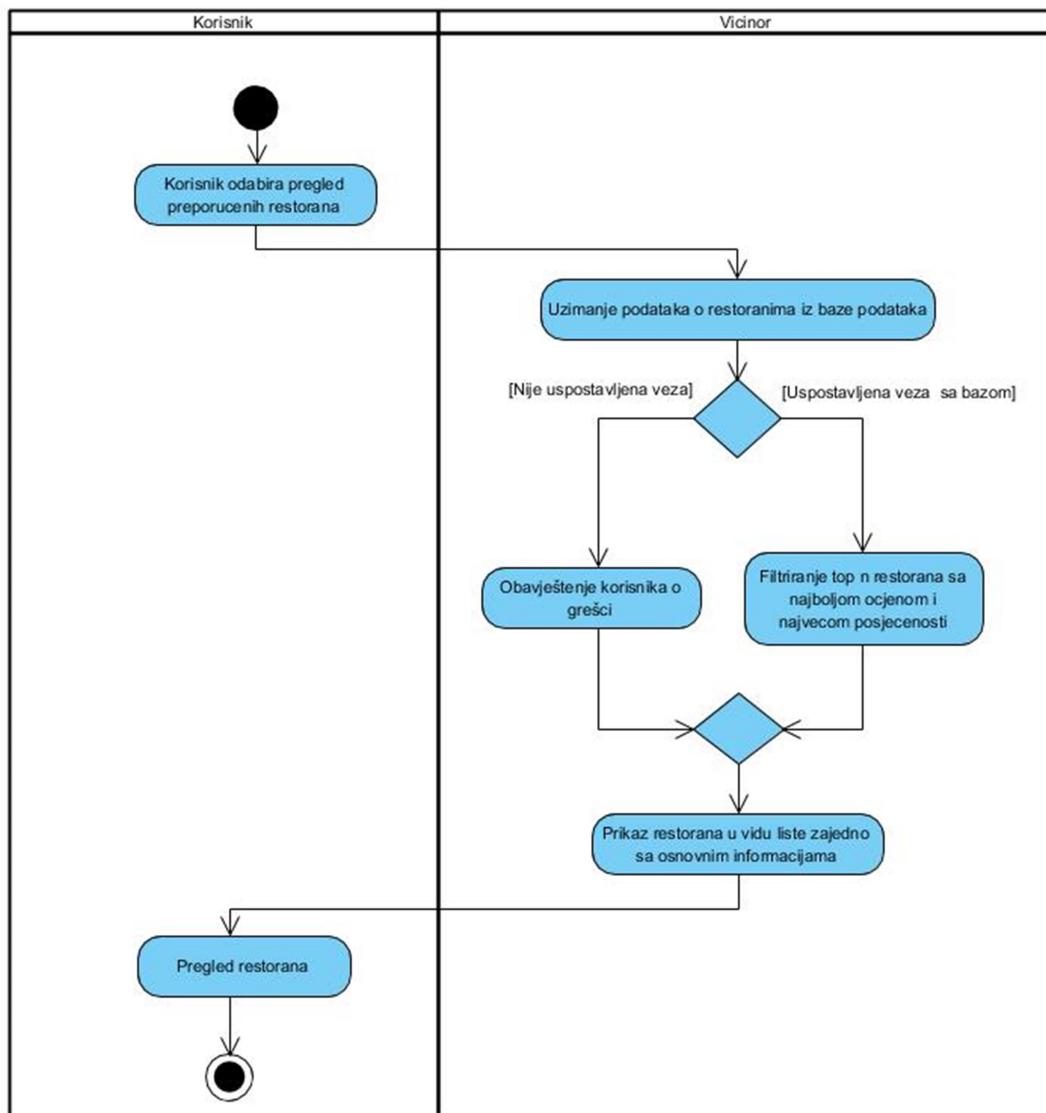
Slika 4. Registracija korisnika

Dijagram klasa (MVVM)

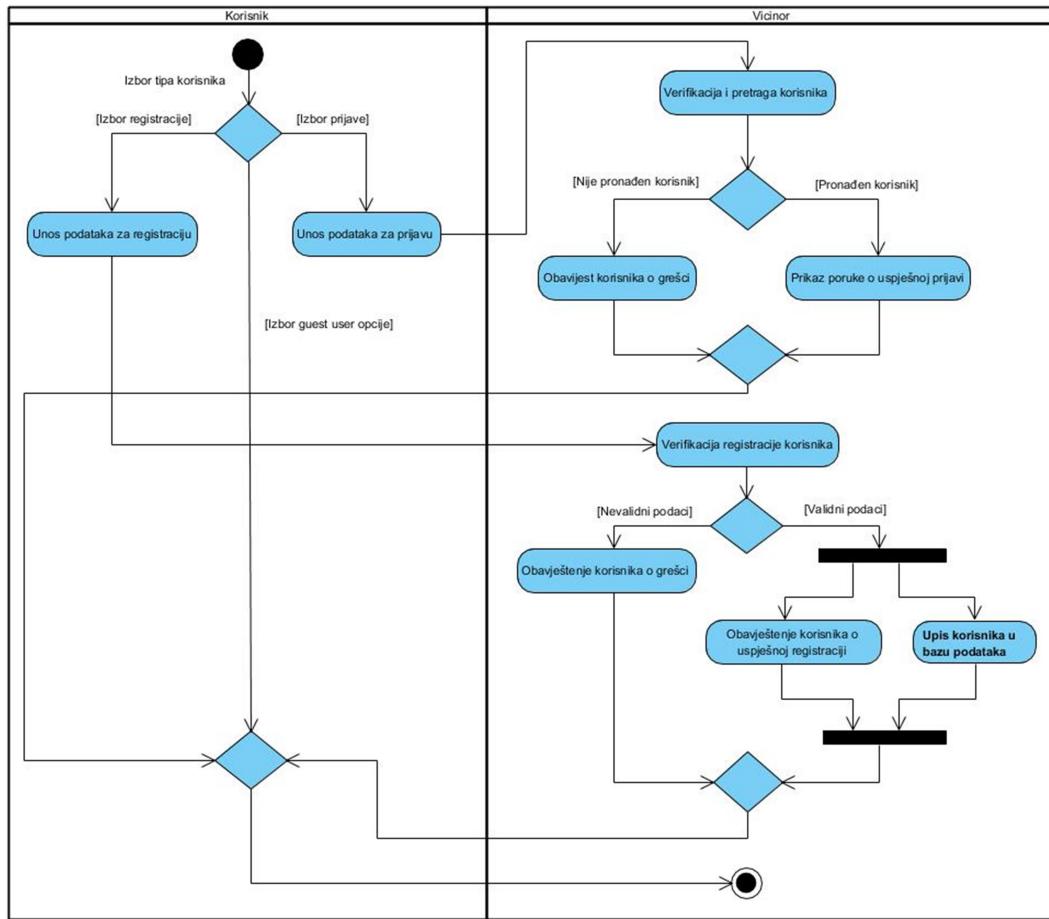


Slika 5. Dijagram klasa

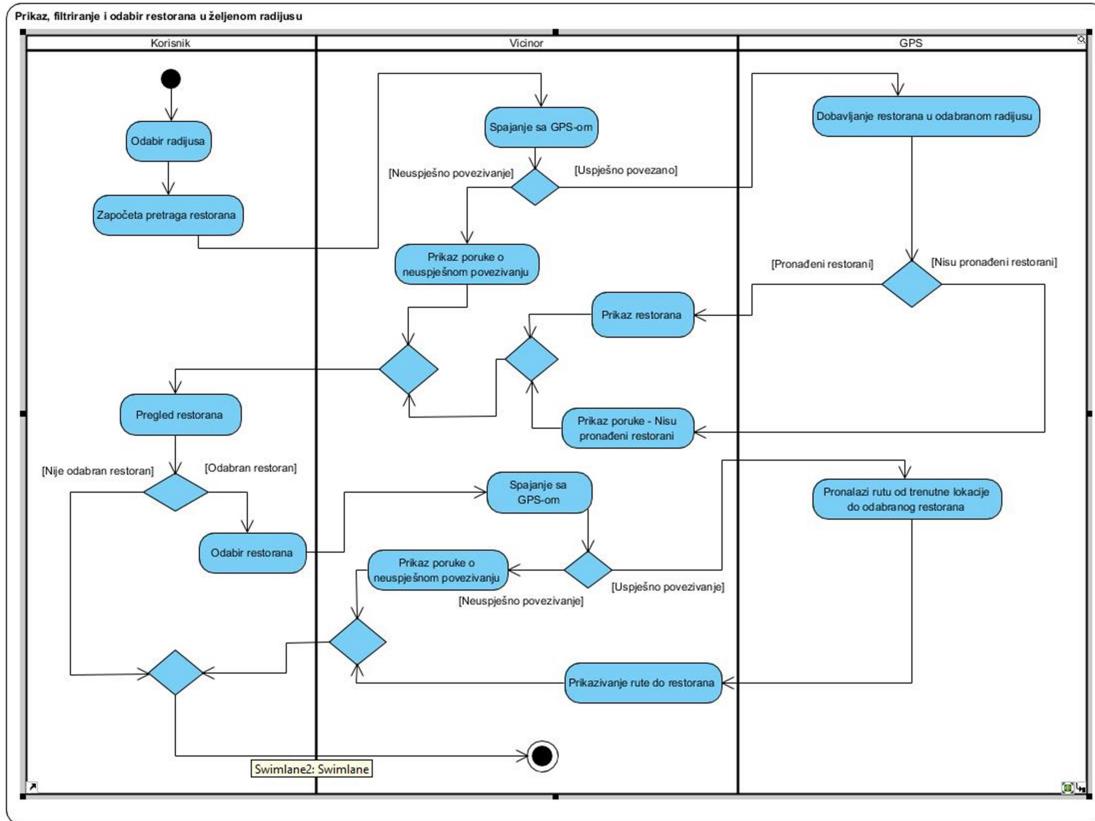
Dijagrami aktivnosti



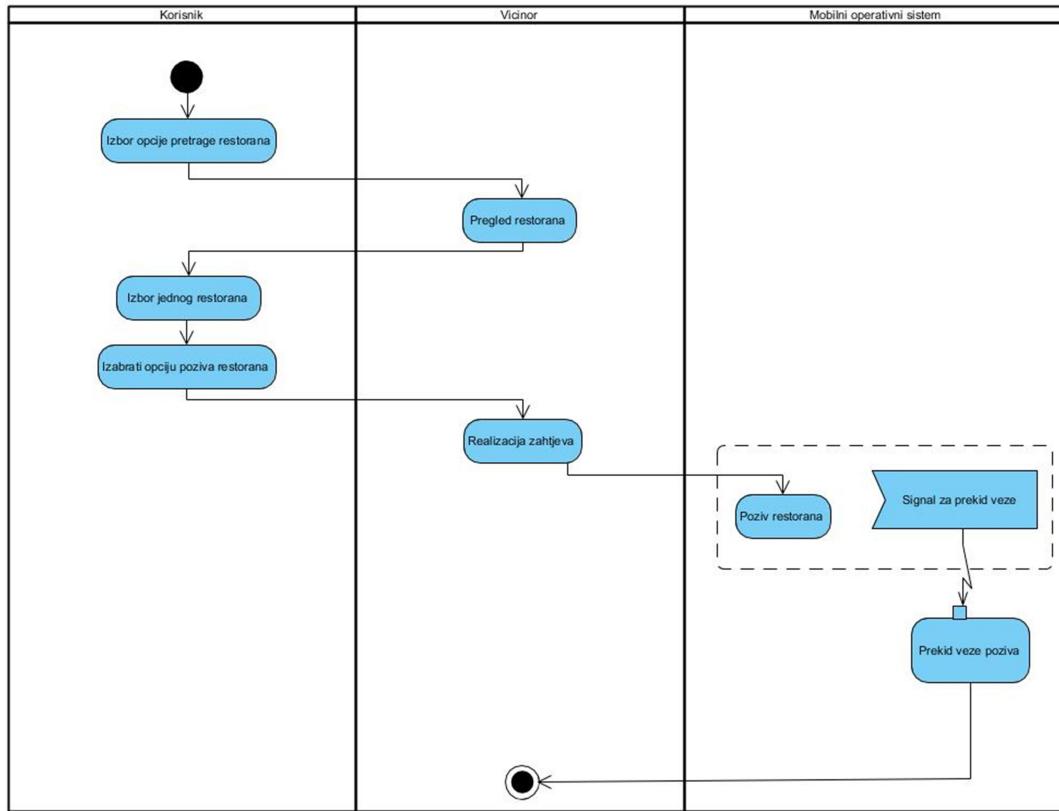
Slika 6. Generisanje preporučenih restorana



Slika 7. Prijava i registracija

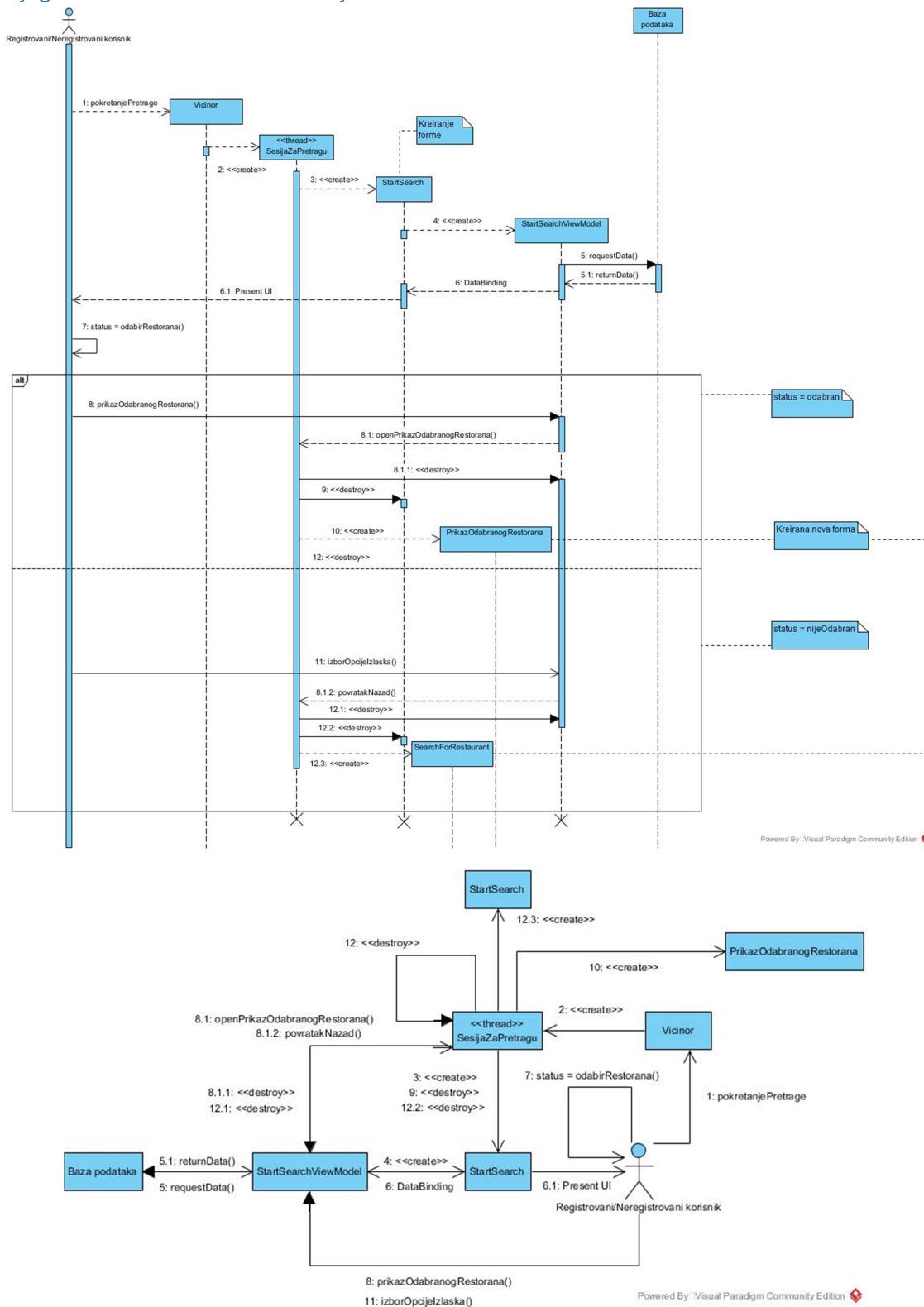


Slika 8. Prikaz, filtriranje i odabir restorana

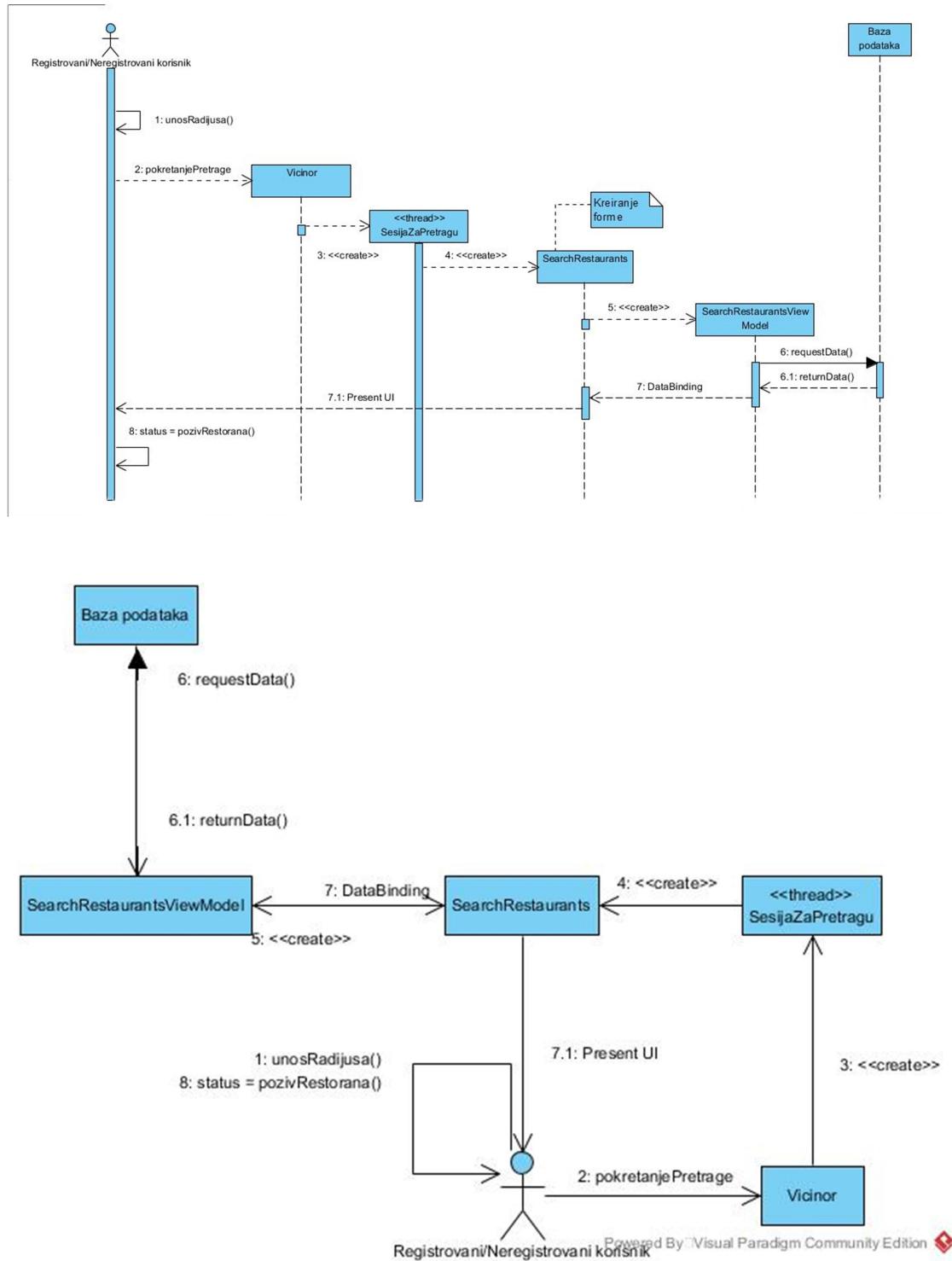


Slika 9. Odabir pazara

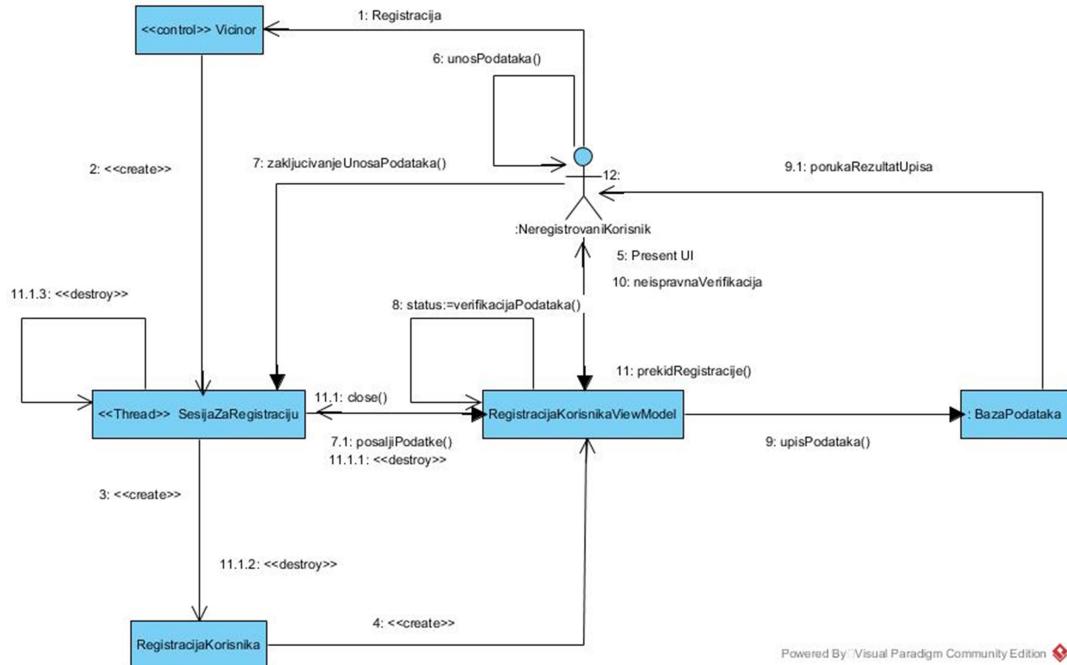
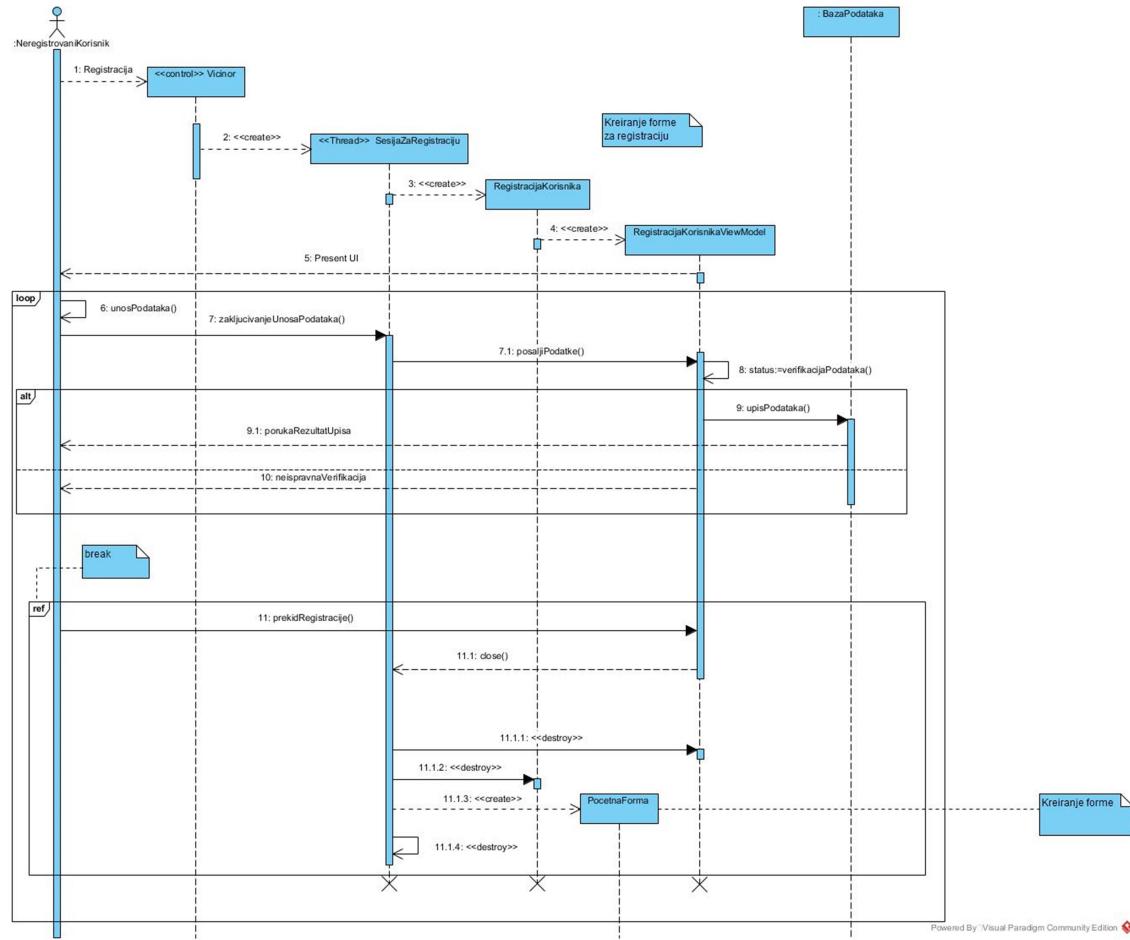
Dijagrami sekvenci i komunikacije



Slika 10. i 11. Odabir restorana

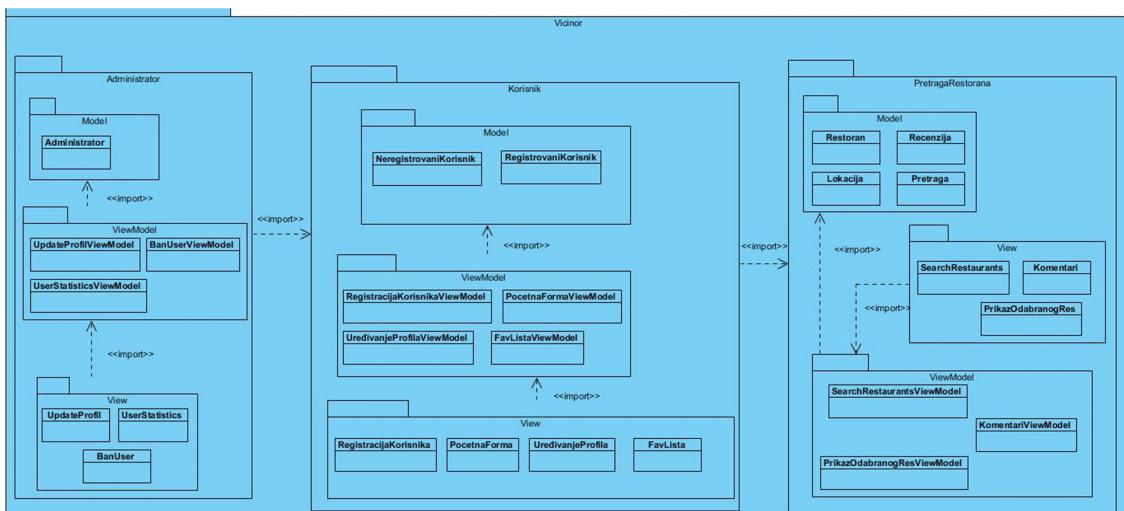
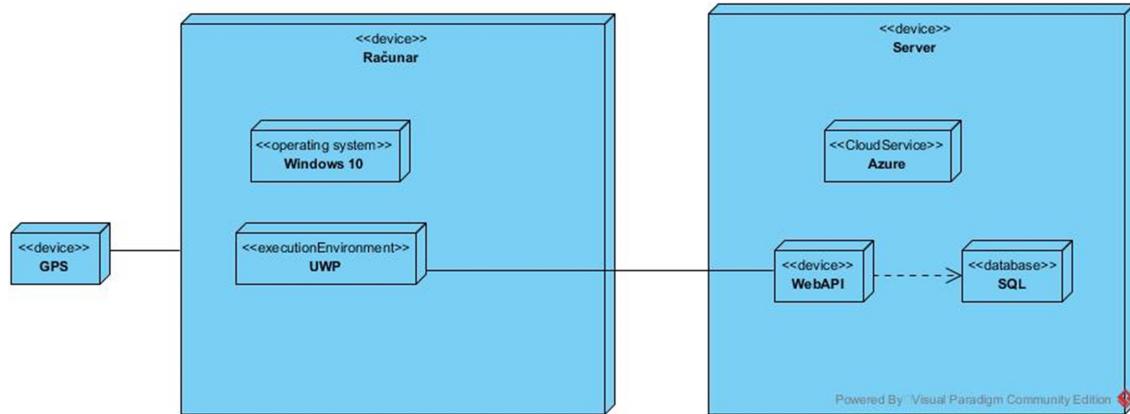
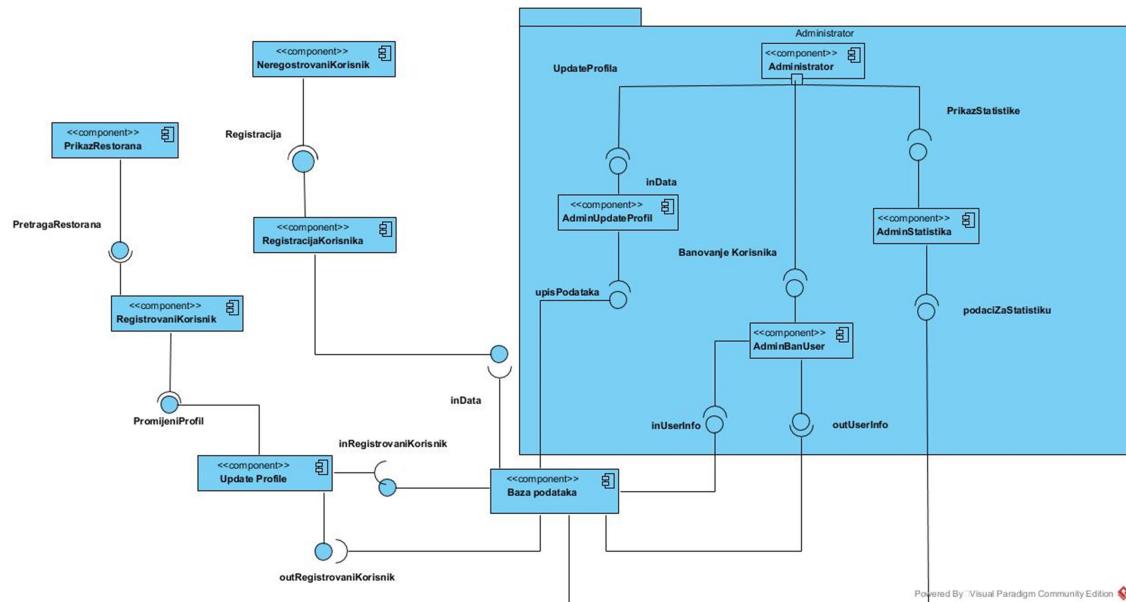


Slika 12. i 13. Pretraga restorana



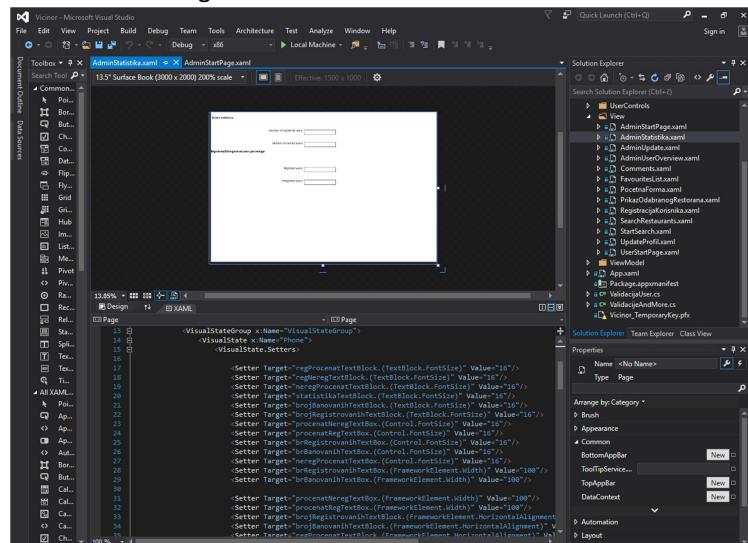
Slika 14. i 15. Registracija korisnika

Dijagrami komponenti, paketa i raspoređivanja

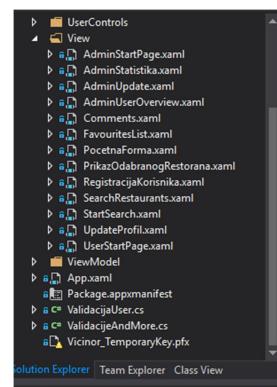


Izvještaj o radu

- Odrađen dizajn u UWP-u i implementacija adaptivnog korisničkog Interfejsa (View) za forme administracije AdminStartPage, AdminStatistika, AdminUpdate i AdminUserOverview. Svaki član tima implementirao po jednu formu te po jednu korisničku kontrolu. Korisničke kontrole koje su implementirane: loginUserControl, procenaUserCOntrol i banUserControl. Naknadno dodane i korisničke kontrole RatingControl, welcomeLogoUserControl i welcomeAdminLogoUserControl.

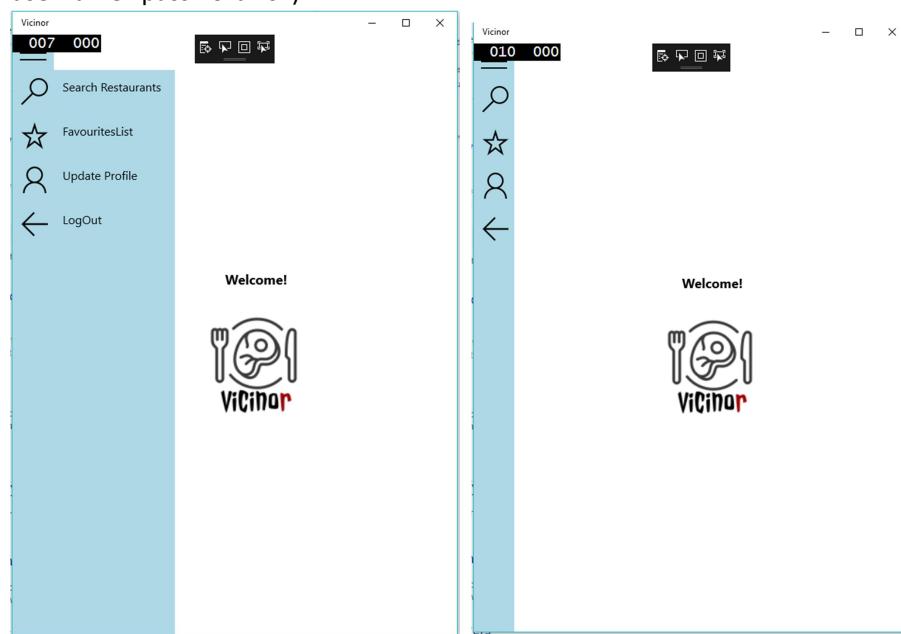


Slika 16. Adaptivni interfejs

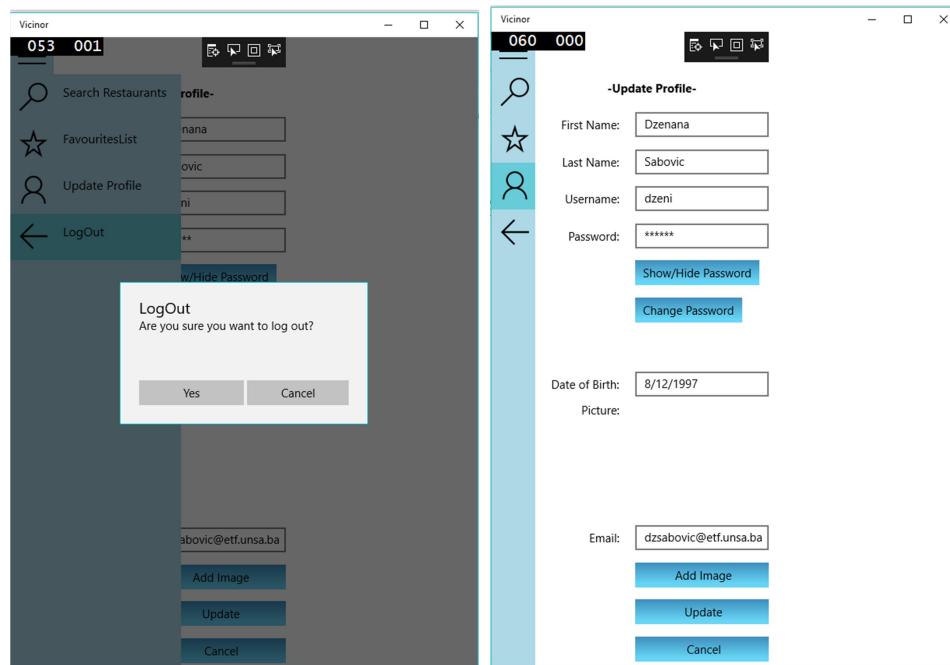


Slika 17. View klase

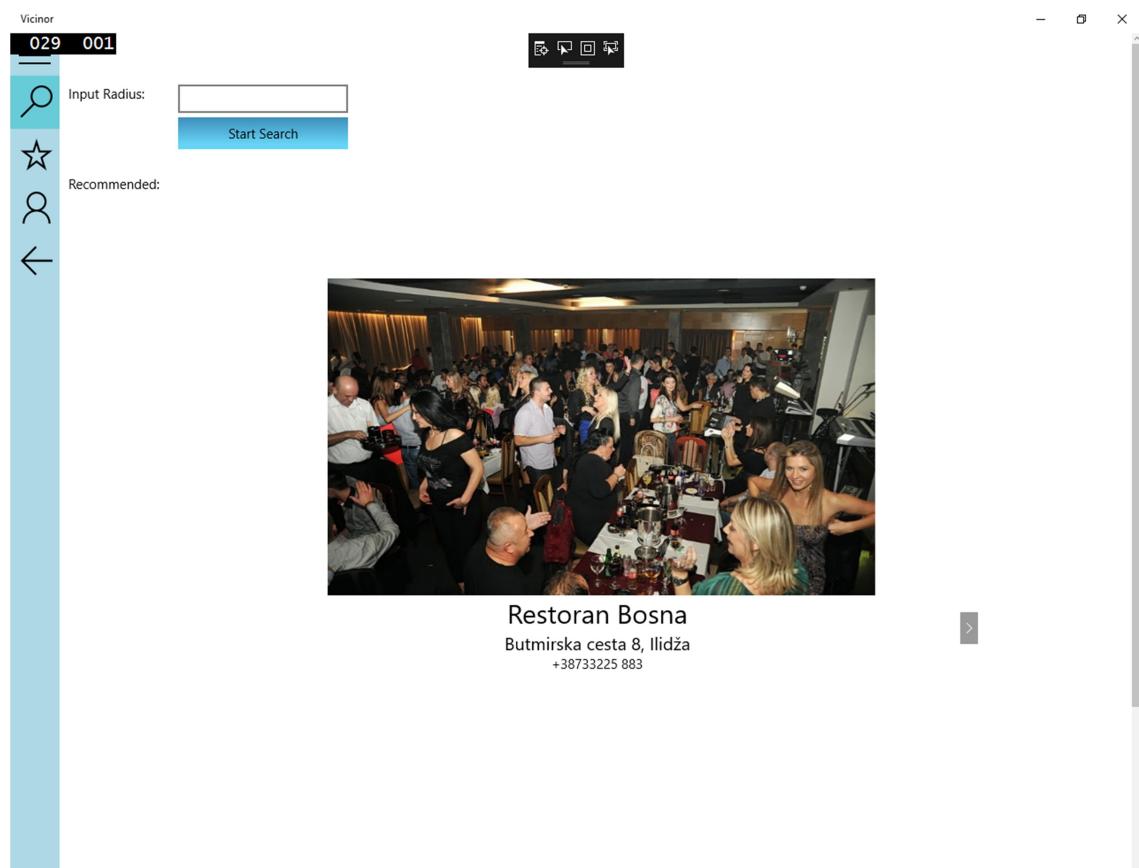
- Implementiran model – funkcionalnosti u UWP-u za forme administracije. Korišten MVVM arhitekturalni patern. Svaki član tima implementirao po jednu klasu ViewModel za odgovarajući View. Implementirane klase modela. Odrađena validacija podataka (ispravnan username i password i sl.).



Slika 18. i 19. Početna forma za korisnika (analogno urađeno i za administratora)

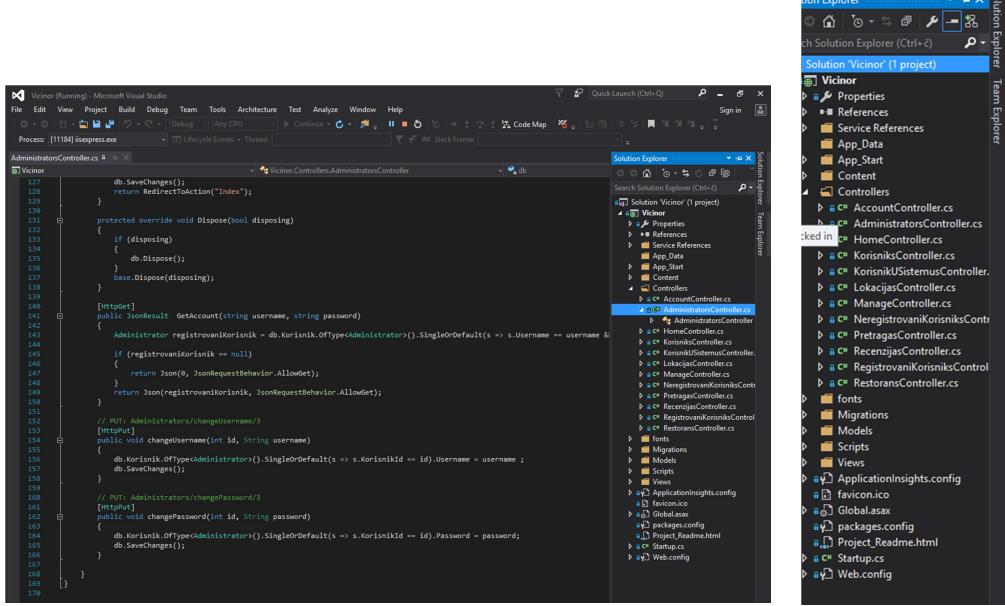


Slika 20. i 21. LogOut korisnika i UpdateProfile forma (Phone view)



Slika 22. Forma za pretragu restorana (Desktop View)

3. Implementirana persistencija podataka – rad sa bazom podataka na cloudu. Povezani backend i frontend dijelovi projekta. Implementirano spašavanje podataka za administraciju. Svaki član tima implementirao upis podataka ili čitanje podataka iz baze.
4. Kreiran ASP.NET projekat pri čemu je arhitektura aplikacije MVC. Implementiran model. Rađeno zajedno.



The screenshot shows the Microsoft Visual Studio interface. The left pane displays the code editor for `Vicinor\Controllers\AdministratorsController.cs`. The right pane shows the Solution Explorer with the project structure for 'Vicinor' (1 project). The structure includes:

- Solution Explorer:**
 - Properties
 - References
 - Service References
 - App_Data
 - App_Start
 - Content
 - Controllers
 - AccountController.cs
 - AdministratorsController.cs
 - HomeController.cs
 - KorisniciController.cs
 - KorisnikUstesumusController.cs
 - LokacijeController.cs
 - ManageController.cs
 - NeregistrovaniKorisniciController.cs
 - NevezetaVezbaController.cs
 - RecenzijeController.cs
 - RestoransController.cs
 - RestoranController.cs
 - Migrations
 - Models
 - Scripts
 - Views
 - ApplicationInsights.config
 - favicon.ico
 - Global.asax
 - packages.config
 - Project_Readme.html
 - Startup.cs
 - Web.config
- Team Explorer:** Shows 'checked in' files.

Slika Kontroleri

5. Kreirani odgovarajući kontroleri i Views te izvršeno povezivanje sa modelom. Radili svi članovi zajedno.
6. Rađeno sa Azure bazom podataka.
7. Implementiran rad sa vanjskim uređajima (GPS za uzimanje trenutne lokacije korišten na više mesta – za uzimanje restorana u određenom radijusu u odnosu na trenutnu lokaciju te za prikaz rute do odabranog restorana). Korištene mape za prikaz rute. Svaki član tima je implementirao rad sa vanjskim uređajem ili jednu specifičnu funkcionalnost.
8. Implementiran ASP.NET WEB API servis i REST API za određene funkcionalnosti koje koristi MVC aplikacija. Radili svi članovi skupa.
9. Odrađen poziv API servisa iz ASP.NET MVC aplikacije. Radili svi članovi skupa.
10. Odrađena dokumentacija.
11. Urađen refactoring koda na osnovu kataloga refactoringa. Analizirana mogućnost primjene 6 refactoringa te svaki član tima implementirao po jedan refactoring. Analizirana mogućnost refactoringa pomoću dizajn paterna i implementirana tri dizajn paterna (Svaki član tima implementirao po jedan). Napisana dokumentacija za refactoring.
12. Urađen online deployment @ <https://vicinor.azurewebsites.net/>

Tabela očekivanih i implementiranih funkcionalnosti

Akter	Funkcionalnost	Implementirano
Administrator	Izmjena profila (lozinka i password)	Da
Administrator	Pregled statistike	Da
Administrator	Banovanje korisnika	Da
Neregistrovani korisnik	Registracija korisnika	Da
Registrovani korisnik	Dodavanje restorana u listu omiljenih	Da
Registrovani korisnik	Uređivanje profila	Da
Korisnik	Pretraga restorana u željenom radijusu	Da
Korisnik	Pregled osnovnih informacija o restoranu	Da
Korisnik	Pregled recenzija restorana	Da
Korisnik	Dobijanje uputa do restorana	Da