**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**V A R A Ž D I N**

**Monika Jakopanec**

**Natalija Jurčić**

**Jakov Nakić**

**Hrvoje Šoštarić**

**Ivor Gradiški Zrinski**

**FOOD2GO**

**Projekt iz kolegija analiza i razvoj programa**

**tehnička Dokumentacija**

**Varaždin, studeni 2019.**

**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**V A R A Ž D I N**

**Tim: AIR1914**

**Jakopanec, Jurčić, Nakić, Šoštarić, Gradiški Zrinski**

**GitHub:** <https://github.com/AIR-FOI-HR/AIR1914>

**ZenHub:** <https://app.zenhub.com/workspaces/air1914-5db63915a2fd04000141cb7d/board?repos=217937076>

**Web stranica:** <https://github.com/AIR-FOI-HR/AIR1914/tree/master/Food2GoWebDio>

**FOOD2GO**

**Projekt iz kolegija analiza i razvoj programa**

**tehnička Dokumentacija**

Mentor:

Doc.dr.sc. Zlatko Stapić

**Varaždin, studeni 2019.**

**Sadržaj:**

[1. Uvod 4](#_Toc24885463)

[1.1. Korištene tehnologije i alati 4](#_Toc24885464)

[1.2. Funkcionalnosti 4](#_Toc24885465)

[2. Specifikacija korisničkih zahtjeva 6](#_Toc24885466)

[3. Arhitektura sustava i aplikacije 7](#_Toc24885467)

[3.1. Arhitektura sustava 7](#_Toc24885468)

[3.2. Arhitektura aplikacije 7](#_Toc24885469)

[3.3. Arhitektura modula 9](#_Toc24885470)

[4. Skica ekrana aplikacije 10](#_Toc24885471)

[5. ERA model 19](#_Toc24885472)

[6. Specifikacija funkcionalnosti 20](#_Toc24885473)

1. Uvod

FOOD2GO je mobilna aplikacija koja korisnicima omogućuje naručivanje hrane i pića. Cilj ove aplikacije je omogućiti korisnicima brzo naručivanje određene hrane i pića te plaćanje karticom.

Aplikacija će biti dostupna korisnicima koji koriste Android platformu te će administratoru biti dostupna i u web obliku.

* 1. Korištene tehnologije i alati
* Za ovaj projekt koristit ćemo:
* Java
* PHP
* SourceTree
* Android Studio
* GitHub
* Visual Paradigm
* MySql Workbench
* ZenHub
* PhpStorm
* Marvel
* ScrumDesk
* Dropbox
* PHPMyAdmin
  1. Funkcionalnosti
* Registracija i prijava korisnika - unos općih podataka preko forme za registraciju, slanje maila s aktivacijskim kodom, aktivacija korisnika i daljnja prijava istoimenog korisnika
* Narudžba hrane - Registrirani korisnik pregledava ponudu i naručuje željenu hranu kroz mobilnu aplikaciju
* Potvrda i praćenje narudžbe - Zaposlenik zaprima i potvrđuje klijentovu narudžbu te joj mijenja status
* Bodovi vjernosti (QR kod) - klijent s računa skenira generirani QR kod i pomoću toga skuplja bodove pomoću kojih ostvaruje posebne pogodnosti
* Povratna informacija (ocjena) od strane klijenta - nakon konzumacije hrane klijent ima mogućnost dati svoj komentar na naručenu hranu te je ocijeniti

Prethodno navedene funkcionalnosti će biti implementirane za mobilnu aplikaciju. U svrhu izrade ovog projekta izradit će se i Web aplikacija koju koristi administrator sa sljedećim funkcionalnostima:

* CRUD nad artiklima
* Pregled i blokiranje korisnika
* Pregled povratnih informacija
* Kreiranje i upravljanje s bodovima vjernosti

1. Specifikacija korisničkih zahtjeva

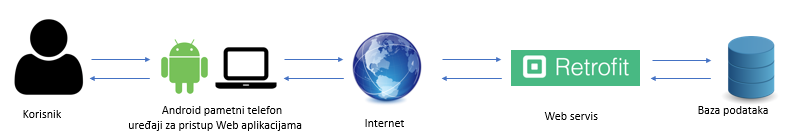
FOOD2GO aplikaciju mogu koristiti restorani brze hrane jer omogućuje korisniku da brzo naruči određenu hranu i piće. Ovu aplikaciju će moći koristiti dvije vrste korisnika, a to su: neregistrirani korisnik i registrirani korisnik. Također, za administratora, koji je određen od strane restorana, bit će napravljena Web aplikacija.

Neregistrirani korisnik ima pristup prikazu hrane i pića koje nudi restoran. Također, može pregledavati recenzije koje ostavljaju registrirani korisnici.

Registriranom korisniku pružaju se veće mogućnosti nego neregistriranom. Osim što će moći sve što i neregistrirani korisnik, moći će i naručiti hranu i piće. Omogućeno mu je i online plaćanje prilikom kojeg korisnik može skenirati QR kod kako bi ostvario bodove, a nakon što je sakupio određeni broj bodova može iskoristiti nagradu prilikom sljedeće narudžbe. Nakon što naruči i plati, moći će ostaviti recenziju. Nakon registracije ili tijekom korištenja korisnik može svoje podatke promijeniti u aplikaciji.

Administrator će dobiti svoju Web aplikaciju koju će samo on koristiti. Aplikacija će mu omogućiti unos hrane i pića, unos nagrada, ažuriranje artikala i nagrada, pregled i blokiranje korisnika te pregled recenzija.

1. Arhitektura sustava i aplikacije
   1. Arhitektura sustava

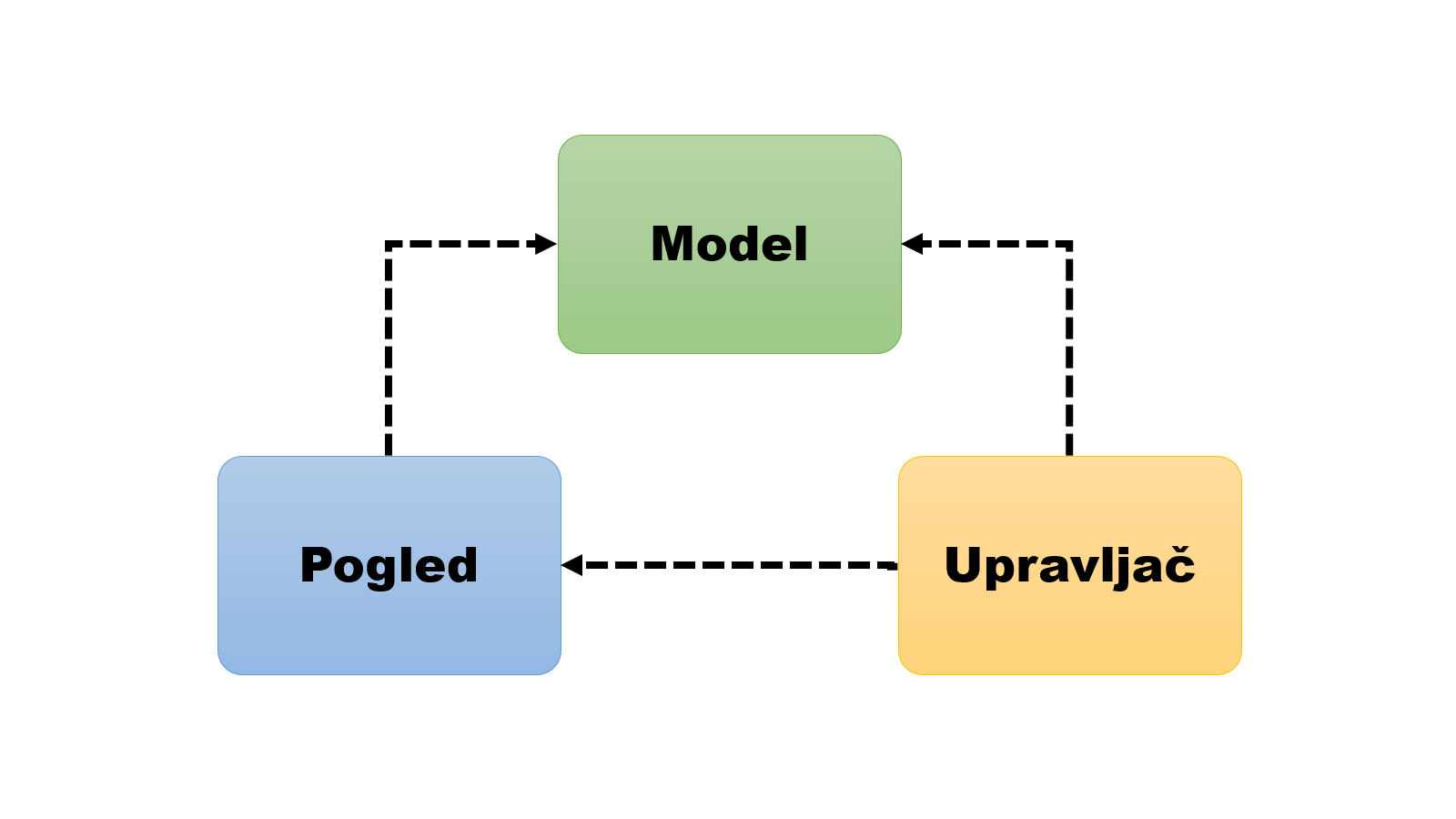


Slika 1. Arhitektura sustava i aplikacije

Korisnik se pomoću pametnog telefona ili stolnog računala prijavi i automatski se pošalje HTPP zahtjev Web servisu. Uloga Web servisa je dohvatiti podatke iz baze podataka i pohraniti ih u HTTP odgovor. Kada se pohrane, šalje te podatke aplikaciji koja ih dekodira i potom prikaže na ekranu korisnika.

* 1. Arhitektura aplikacije

Budući da se kod izrade mobilnih aplikacija logika korisničkog sučelja mijenja češće nego poslovna logika, potreban nam je način odvajanja funkcionalnosti korisničkog sučelja. Zbog toga koristimo Model-Pogled-Upravljač troslojnu arhitekturu kao rješenje spomenutog.



**Slika 2. Troslojna arhitektura aplikacije**

* **Model** — podatkovni sloj, odgovoran za upravljanje poslovnom logikom i rukovanje mrežom ili API-jem baze podataka.
* **Pogled** — sloj korisničkog sučelja, vizualizira podataka iz Modela.
* **Upravljač** — logički sloj, dobiva obavijest o ponašanju korisnika i ažurira Model po potrebi.

Dakle, ovo znači da i Upravljač i Pogled ovise o Modelu: Upravljač za ažuriranje podataka, Prikaz za dobivanje podataka. Ali ono što je najvažnije - Model je odvojen i može se testirati neovisno o korisničkom sučelju.

**Prednosti**

Model-Pogled-Upravljač arhitektura dosta dobro podržava odvajanje odgovornosti. Ova prednost ne samo da povećava provjerljivost koda, već i olakšava njegovo proširivanje, omogućavajući prilično jednostavnu implementaciju novih značajki.

Klase Modela nemaju nikakvu referencu na Android klase i stoga su jednostavne za jedinično testiranje. Upravljač ne proširuje ili ne implementira nijednu Android klasu i treba imati referencu na klasu sučelja Pogleda. Na taj je način moguće i jedinično testiranje Upravljača.

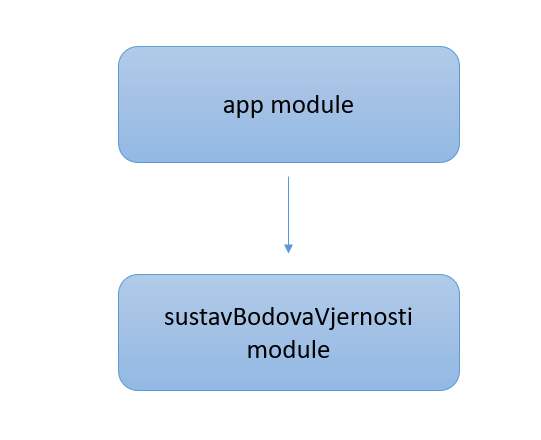
Ako Pogledi poštuju načelo jedinstvene odgovornosti, onda je njihova uloga samo ažurirati Upravljača za svaki korisnički događaj i samo prikazati podatke iz Modela, bez implementiranja bilokakve poslovne logike. U ovom bi slučaju UI testovi trebali biti dovoljni da pokriju funkcionalnosti Pogleda.

**Nedostaci**

Glavni nedostatak je što Pogled ovisi o Upravljaču i o Modelu. Zavisnost Pogleda o Modelu postaje nedostatak u složenim Pogledima. Da bi se logika u Pogledu svela na najmanju moguću mjeru, Model bi trebao biti u mogućnosti osigurati metode testiranja za svaki element koji se prikazuje.

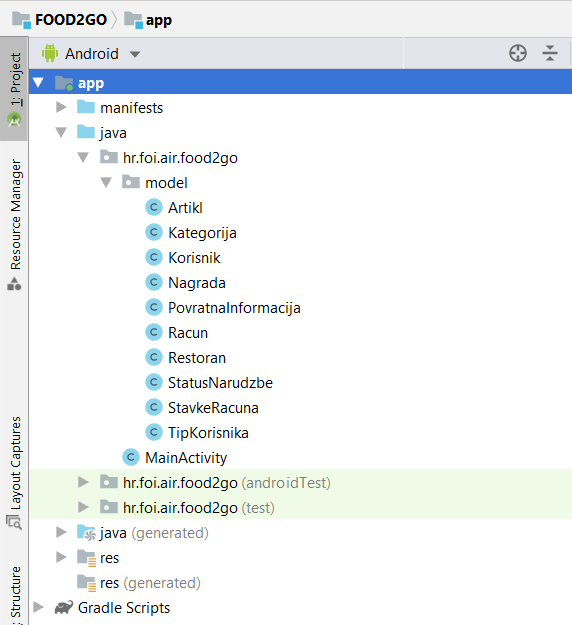
S obzirom da Pogled ovisi i o Upravljaču i Modelu, promjene u logici korisničkog sučelja mogu zahtijevati ažuriranja u nekoliko klasa, smanjujući fleksibilnost ove arhitekture.

* 1. Arhitektura modula

****

Slika 3. Ovisnost modula

Slika 3. prikazuje da će ova aplikacija koristiti dva modula. Od ta dva modula za prvu fazu implementiran je samo jedan modul, tj. app modul. Drugi modul se odnosi na sustav bodova vjernosti i on će biti implementiran u kasnijim fazama projekta.



Slika 4. App modul

1. Skica ekrana aplikacije



**Početni zaslon**

Prilikom pokretanja aplikacije otvara se početni zaslon. Na njemu se nalaze ime i logo aplikacije, te gumbovi za prijavu i registraciju. Prijaviti se može samo registrirani korisnik (korisnik koji već ima kreiran korisnički račun), a registrirati se može neregistrirani korisnik odnosno onaj koji još nema korisnički račun.



**Prijava**

Klikom na gumb „Prijavi se“ sa početnog zaslona, otvara se novi zaslon – Prijava. Ovdje registrirani korisnik upisuje svoje korisničko ime i lozinku, a potom se prijavljuje u aplikaciju klikom na gumb „Prijavi se“. Ako je korisnik nekim slučajem zaboravio svoju lozinku, može kliknuti na „Zaboravili ste lozinku?“, čime mu se otvara zaslon za opravak lozinke.



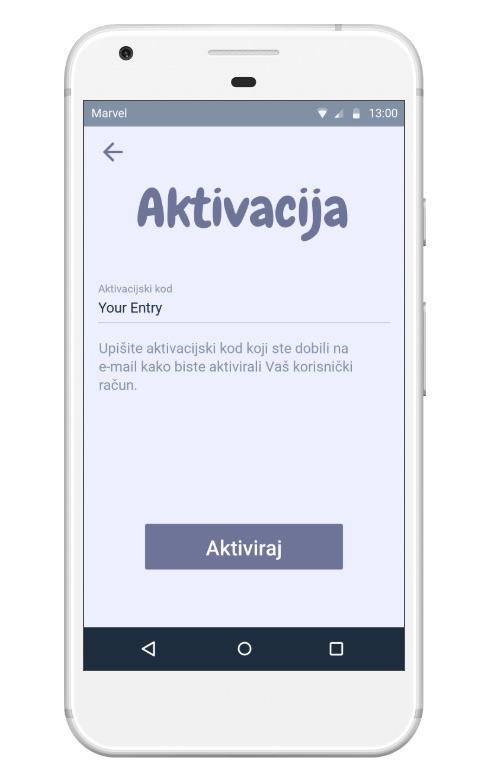
**Oporavak lozinke**

Klikom na „Zaboravili ste lozinku?“ sa zaslona prijave, otvara se zaslon Oporavak lozinke. Ovdje korisnik može oporaviti svoju lozinku na način da upiše svoj e-mail (onaj s kojim se registrirao) i klikne na gumb „Pošalji lozinku“. Ovim postupkom, korisnik će na e-mail primiti lozinku kojom se može prijaviti. Jednom kad se prijavi, lozinku može promijeniti u postavkama svog korisničkog računa.



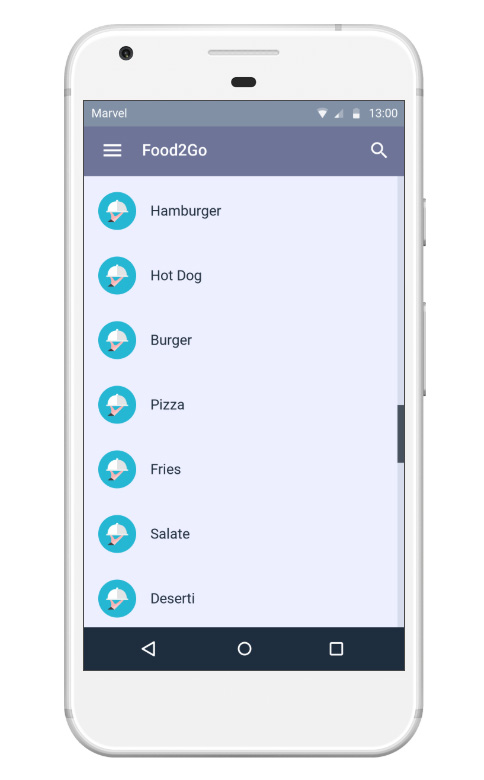
**Registracija**

Klikom na gumb „Registriraj se“ sa početnog zaslona, otvara se zaslon Registracija. Ovo je zaslon kojim se koristi neregistrirani korisnik, a u svrhu registracije istog. Ovdje korisnik treba upisati svoje osobne podatke, te e-mail, korisničko ime, i lozinku. Kada je sve popunjeno, može se kliknuti gumb „Registriraj se“, čime se otvara zaslon za aktivaciju korisničkog računa.



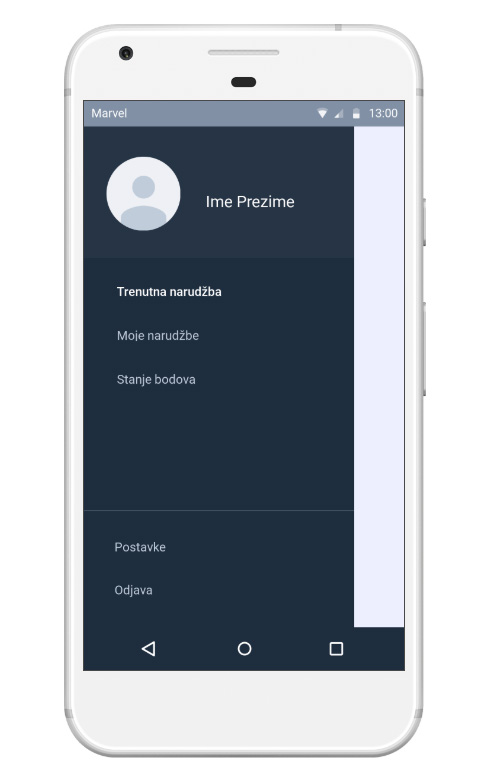
**Aktivacija**

Aktivacija je zaslon koji se otvara kada se neregistrirani korisnik registrira odnosno kada isti klikne na „Registriraj se“ sa zaslona registracije. Klikom na spomenuti gumb se otvara ovaj zaslon, te se na e-mail korisnika šalje aktivacijski kod koji korisnik na ovom zaslonu treba upisati kako bi aktivirao svoj korisnički račun.



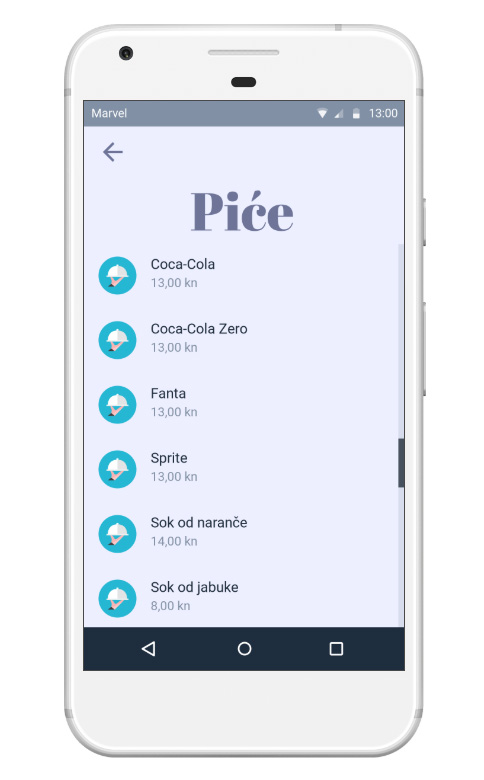
**Glavni zaslon**

Ovo je glavni zaslon koji se otvara kada se registrirani korisnik prijavi. Na vrhu, ikona sa tri vodoravne crte služi za prikazivanje i sakrivanje izbornika. Ispod aplikacijske trake prikazana je lista različitih kategorija. Klikom na jednu od kategorija, otvara se zaslon koji prikazuje sve artikle iz te kategorije.



**Izbornik**

Klikom na ikonu sa tri vodoravne crtice, prikazuje se izbornik. Na vrhu se nalazi slika, ime i prezime korisnika. Ispod se nalaze gumbovi koji otvaraju istoimene zaslone. Zadnji gumb, „Odjava“, odjavljuje korisnika te ga vraća na početni zaslon.



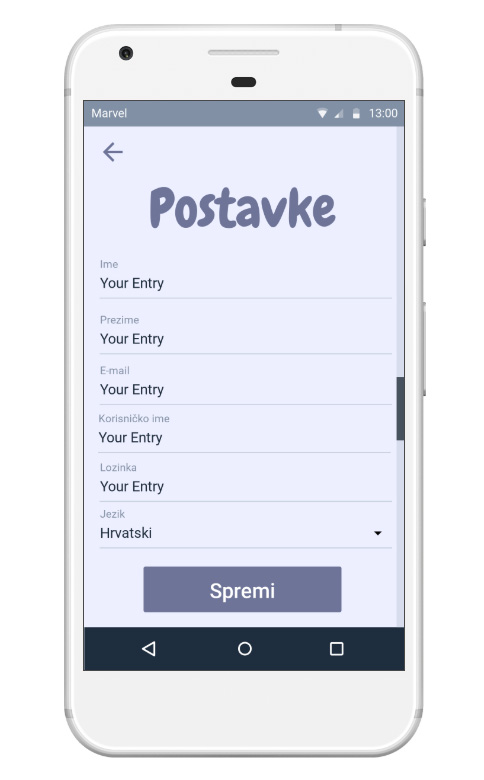
**Kategorija**

Klikom na jednu od kategorija sa glavnog zaslona, otvara se zaslon sa detaljima te kategorije, odnosno prikazuju se svi artikli iz te kategorije. Uz svaki artikl, prikazana je i njegova jedinična cijena, te ikona sa lijeve strane.



**Artikl**

Klikom na jedan od artikala sa zaslona kategorije, otvara se zaslon na kojem su prikazani detalji odabranog artikla. Prikazani su ime, slika, opis, te cijena artikla. Klikom na „Dodaj u košaricu“, artikl se dodaje u košaricu odnosno kreira se nova narudžba (ako već ne postoji) i u nju se dodaje odabrani artikl, ili se dodaje odabrani artikl na već postojeću narudžbu (ako ona već postoji).



**Postavke**

Klikom na gumb „Postavke“ sa izbornika, otvara se istoimeni zaslon. Ovdje korisnik može promijeniti svoje osobne podatke te jezik aplikacije. Nakon bilokakvih promjena, treba kliknuti na gumb „Spremi“ kako bi se promjene pohranile.



**Stanje bodova**

Klikom na gumb „Stanje bodova“ sa izbornika, otvara se istoimeni zaslon. Ovdje korisnik može vidjeti trenutan broj bodova. Bodovi se skupljaju svaki put kad korisnik izvrši narudžbu, a broj bodova koji se dobiva ovisi o veličini iste. Bodovi se mogu iskoristiti kod naručivanja za ostvarivanje popusta.



**Nagrade**

Klikom na gumb „Nagrade“ sa izbornika, otvara se istoimeni zaslon. Ovdje korisnik može vidjeti trenutno aktualne nagrade. Zeleno označena nagrada je ona na koju korisnik ima pravo s obzirom na svoj trenutni broj bodova. Ako nijedna nagrada nije označena zeleno bojom, to znači da korisnik nema dovoljno skupljenih bodova ni za najmanju nagradu.



**Trenutna narudžba**

Klikom na gumb „Trenutna narudžba“ sa izbornika, otvara se istoimeni zaslon. Ovdje korisnik može vidjeti sve artikle koje namjerava naručiti. Uz artikl, prikazana je i njegova ikona, jedinična cijena, te količina u kojoj korisnik želi naručiti istog. Na dnu, u lijevom kutu, prikazana je ukupna cijena narudžbe, a u desnom kutu dva gumba. Gumb „Iskoristi bodove“ služi za iskorištavanje popusta (ukoliko korisnik ima pravo na njega), a gumbom „Naruči“ se narudžba izvršava.



**Iskoristi bodove**

Klikom na gumb „Iskoristi bodove“ sa zaslona trenutne narudžbe, otvara se prozorčić u kojem se pita korisnika želi li uistinu iskoristiti popust na trenutnu narudžbu. Korisnik ima pravo samo na najbolju nagradu s obzirom na svoje trenutne bodove. Naravno, korisnik koji nema dovoljno bodova za bilokakvu nagradu niti ne može kliknuti na gumb „Iskoristi bodove“ odnosno za njega je gumb onemogućen.



**Naruči**

Klikom na gumb „Naruči“ sa zaslona trenutne narudžbe, otvara se prozorčić u kojem se potvrđuje korisniku da je narudžba zaprimljena te da će korisnik dobiti račun na svoj e-mail.



**Moje narudžbe**

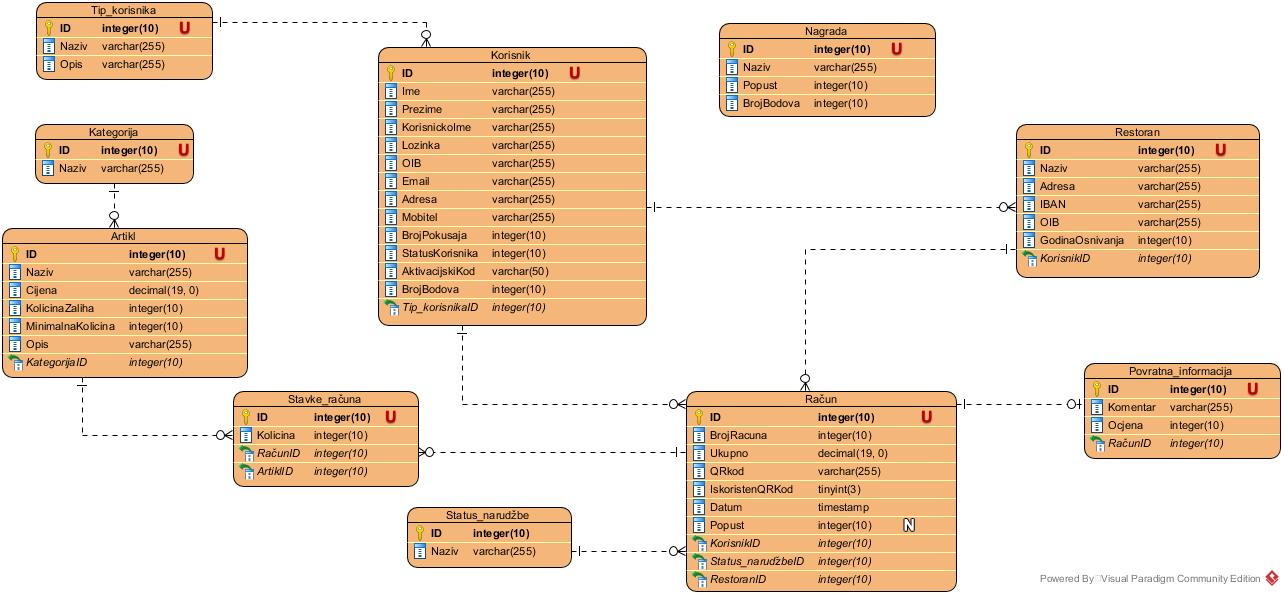
Klikom na gumb „Moje narudžbe“ sa izbornika, otvara se istoimeni zaslon. Ovdje korisnik može vidjeti sve svoje prethodne narudžbe (ako ih ima). Uz svaku narudžbu prikazano je nekoliko artikala iste, datum izvršavanja narudžbe, te ukupna cijena.



**Detalji narudžbe**

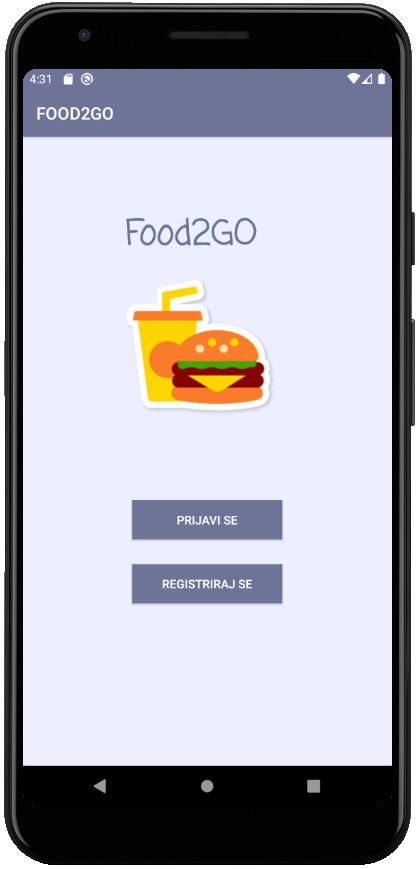
Klikom na jednu od narudžbi, otvara se prozorčić sa detaljima iste. Ovdje se mogu vidjeti svi artikli odabrane narudžbe, zajedno sa njihovom jediničnom cijenom i količinom u kojoj su bili naručeni. Na dnu je prikazana ukupna cijena narudžbe.

1. ERA model



Slika 5. ERA model

FOOD2GO mobilna aplikacija ima dva tipa korisnika, a to su registrirani i neregistrirani korisnik i njih definiramo pomoću tablice *Tip\_korisnika.* Navedena tablica povezana je s tablicom *Korisnik* u kojoj su sadržani osnovni podaci o korisniku (ime, prezime, korisnickoIme, lozinka…). Jedne od važnijih tablica su *Artikl* i *Račun*. Tablica *Artikl* sadrži podatke o hrani i piću koje su u ponudi restorana. Radi lakše organizacije baze dodali smo tablicu *Kategorija* koja je spojena s tablicom *Artikl*. Tablica *Račun* prikazuje što je sve korisnik naručio i koliki mu je račun. Također, u tablici *Račun* postoji atribut QRKod koji sadrži kod preko kojega korisnik dobiva nagradne bodove. Korisnik će u svakom trenutku biti u mogućnosti vidjeti koje su nagrade i upravo zbog toga nam je potrebna i tablica *Nagrada*. Navedene tablice, *Račun* i *Artikl,* su povezane vezom N:M i zato nam je potrebna još jedna tablica koja se zove *Stavke\_računa.* Kada korisnik plati račun, može ostaviti recenziju, a za to nam je bila potrebna tablica *Povratna\_informacija.* Kako bismo znali koji je restoran izdao račun, tablica *Racun* je povezana s tablicom *Restoran*. Korisnik će moći vidjeti i status svoje narudžbe, a da bismo to mogli realizirati dodali smo tablicu *Status\_narudžbe.*

1. Specifikacija funkcionalnosti



1. Web stranica

Web stranica aplikacije Food2Go dostupna je na [linku](https://airfood2go.000webhostapp.com/index.php). Stranica sadrži funkcionalnosti navedene u uvodnom dijelu aplikacije kao i u prijavi projekta. Od funkcionalnosti je potrebno implementirati funkcionalnost “Pregled povratnih informacija”.

Podaci za prijavu:

Korisničko ime: igradiski

Lozinka: roleta66