Sveučilište Jurja Dobrile u Puli

Fakultet Informatike u Puli

**Danijel Andrlik**

**Milorad Ivanović**

**Kristina Lukić**

**Leona Štrucelj**

**Aplikacija za naručivanje i dostavu hrane iz studentskog restorana (Menza TO GO)**

Projektna dokumentacija

Pula, siječanj, 2019. godine

SADRŽAJ

[1. SAŽETAK 3](#_Toc535878379)

[2. UVOD 4](#_Toc535878380)

[3. MOTIVACIJA 5](#_Toc535878381)

[3.1. Buduća poboljšanja 7](#_Toc535878382)

[4. RAZRADA FUNKCIONALNOSTI 8](#_Toc535878383)

[4.1. Use-Case dijagram 8](#_Toc535878384)

[4.2. Sequence dijagram 10](#_Toc535878385)

[4.3. Prototip 12](#_Toc535878386)

[4.4. Klasni dijagram 15](#_Toc535878387)

[5. IMPLEMENTACIJA 17](#_Toc535878388)

[6. KORISNIČKE UPUTE 20](#_Toc535878389)

[POPIS SLIKA I TABLICA 27](#_Toc535878390)

# SAŽETAK

Broj novootvorenih sastavnica i studenata na istima iz dana u dan raste. Povećanjem broja studenata na Sveučilištu javila se i potreba za povećanjem količine hrane u studentskom restoranu. Ponekada je potrebno provesti i do sat vremena u redu kako bi mogao pojesti topli obrok na žlicu. Studentski život se svodi uglavnom na brzu i jeftinu hranu s obzirom da većina studenata odlazi iz mjesta prebivališta i napušta „maminu kuhinju“, a financijske mogućnosti za priuštiti si kvalitetan i zdrav obrok svodi se na minimum. Obzirom da je broj istih studenata prevelik, tj. premašuje granice raspoloživog smještaja u studentskom domu, većina studenata primorena je pronaći smještaj izvan centra grada u kojem studira, a najčešće je taj smještaj udaljen kilometrima od fakulteta i studentskoga restorana. Student bi trebao biti na prvom mjestu pa je tako u interesu Sveučilišta, točnije cijelog Studentskog centra, olakšati i automatizirati studentske potrebe. Upravo s tim ciljem javila se ideja za aplikacijom i uvođenjem mogućnosti narudžbe i dostave hrane iz studentskog restorana.

Dokumentacija sadrži nekoliko dijelova pa ćemo tako na samom početku iste opisati aplikaciju i njenu svrhu, ciljano tržište, glavne prednosti njezinim uvođenjem te koje bismo mogućnosti još željeli dodati daljnjim razvijanjem. Nadalje, usporediti ćemo aplikaciju s konkurentskim rješenjem kroz navođenje prednosti iste uz pomoć SWOT analize. Slijedi implementacija aplikacije u kojoj ćemo se osvrnuti na model i ostale dijelove aplikacije. Na kraju dolazimo do uvođenja korisnika kroz aplikaciju pomoću korisničkih uputa.

# UVOD

Studentski restoran dio je ugostiteljstva iako se mnogo puta ne promatra iz iste perspektive s obzirom da su njegovi primarni korisnici isključivo studenti. Sama grana ugostiteljstva uz mnoštvo ugostiteljskih obrta poprima velik broj aplikacija za lakše razvijanje, ugađanje korisniku, napredovanje, efikasnost. Tako se razvila i naša ideja za razvijanje i napredovanje studentskog restorana kako bi se ugodilo na efikasan način studentu kako još jednoj „mladoj nadi“ za bolju budućnost. Možemo zaključiti da su ciljano tržište studenti, točnije studentski restorani koji će uvesti modernizaciju na ovaj način. Kao studenti na području Pule, bazirali smo se na dostavi hrane iz Studentskog restorana Pula no u budućnosti bismo htjeli da aplikacija bude korištena na razini cijele Hrvatske i šire. Razlog predviđanja uspješnosti aplikacije je zadovoljstvo sa studentske strane te financijska rast proračuna studentskih restorana ujedno i smanjenje velikih redova koje može dovesti i do pada cijelog sustava uz ostale nedostatke. Aplikacija omogućuje kreiranje narudžbe po vlastitom odabiru s obzirom da je student dužan uzeti cijeli posluženi menu kada se nalazi u restoranu, a ovako će moći izabrati djelomičan menu čime se smanjuju velike količine bačene hrane. Također omogućuje online mogućnost plaćanja što uvelike ubrzava cijelu situaciju jer se mnogo puta dogodi da student nema gotovine kod sebe ili da zaposlenik u menzi nema dovoljno sitnog novca za izvratiti.

Nudimo brojne pogodnosti koje će biti objašnjene u nastavku rada.

# MOTIVACIJA

Kao što smo prije naveli, naše ciljano tržište su studentski restorani te studenti. Aplikacija je trenutno prilagođena samo studentima Republike Hrvatske pa je i njen službeni jezik hrvatskki, no u skoroj budućnosti nam je cilj proširiti u ostale države gdje bi naravno imala mogućnost odabira jezika ovisno o tome od kuda studennt naručuje. Kako je ovo ujedno i inovacija, tj. ovakva ideja još ne postoji ili nije zaživjela, očekujemo razvoj na području Hrvatske no s vremenom i dalje, ovisno o vanjskom obrazovnom sustavu. Trenutno nemamo konkurencije pa bismo voljeli da u budućnosti ovo zaživi i postane obavezan dio koji će koristiti svi studentski restorani. Kako je studentski restoran također dio ugostiteljstva, ne možemo reći da su nam „klasični“ restorani konkurencija s obzirom da koriste skroz drukčiji sustav i politiku, a ni cijene nisu usporedive. Aplikacija ne zauzima puno mjesta i kada je korisnik skine odmah je spremna za korištenje.

- jednostavan odabir i pregled podataka

- zauzima malu količinu memorije

- brzina

- pregled starih i novih narudžbi

- pregled menu-a u

realnom vremenu

- mali broj funkcionalnosti

- mogućnost korištenja samo na hrvatskom jeziku

- nedostatak marketinških vještina

- jednostavnost plaćanja

- dodavanje novih

funkcija i mogućnosti

- unapređenje poslovanja

- zahtjevnost korisnika

- aplikacija ne ostvaruje očekivane rezultate te korisnici nisu zainteresirani za nju

- mogućnost kopiranja

- teško plasiranje na tržište

T

O

W

S

Slika 1. SWOT analiza aplikacije

Iz prikazane SWOT analize vidljive su nam neke od glavnih prednosti uvođenja aplikacije u poslovanje kao jednostavnost odabira narudže te pregled istih te da korisnici ne bi trebali imati nikakvih problema pri baratanju aplikacijom zbog njene jednostavnosti i konkretno navedenih stvari. Također, uz prednosti postoje i rizici na koje možda i ne možemo utjecati poput npr. teškog plasiranja na tržište s obzirom da je ovo ujedno i inovacija pa ne možemo znati koliko će aplikacija biti učinkovita i da li će se zbilja korisnici zainteresirati oko ovog načina prehranjivanja. Isto tako, iako su u današnje vrijeme svi više manje dobro informatički pismeni, još uvijek postoje oni koji imaju slabo poznavanje informatičke tehnologije pa će mu samim time biti teže koristiti aplikaciju.

Za uvođenje aplikacije u poslovanje potrebna je jedina predispozicija, a to je računalo s bilo kojim operativnim sustavom. Kod aplikacije je također dobro što služi za kontinuirano korištenje, što znači kad netko skine aplikaciju i ako je počne koristiti, vjerojatno će je koristiti na dulje vrijeme.

Aplikacija služi ujedno i zaposlenicima u menzi kao i Studentskom centru, a ne samo studentima jer ipak od nje profitira cjelokupno poslovanje poduzeća.

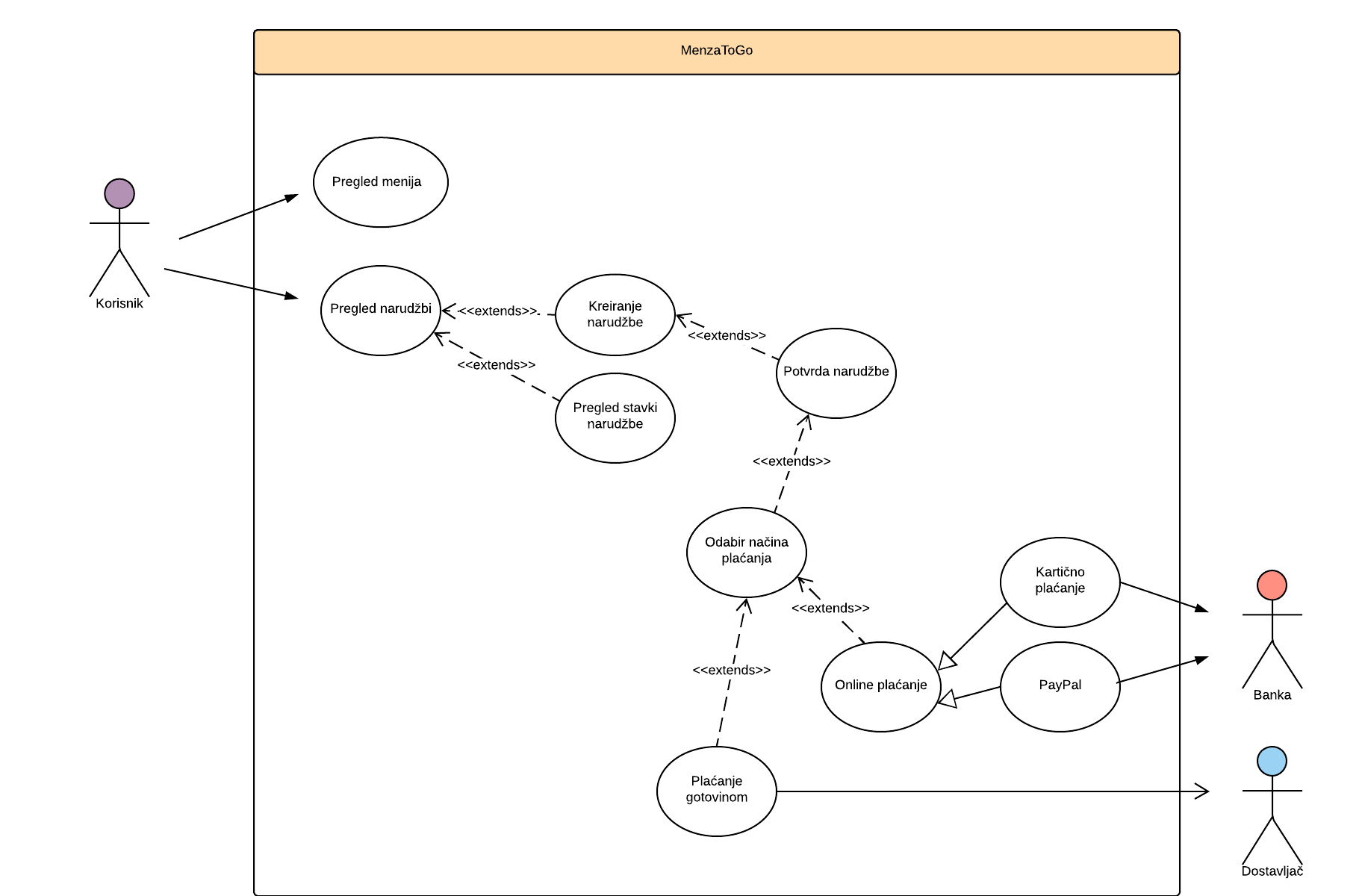
## Buduća poboljšanja

Pri poboljšanju aplikacije dodali bismo nove funkcionalnosti koje bi se odnosile na poboljšanje korisničkog iskustva u aplikaciji, rješavanje bugova prema korisničkim pritužbaama i vlastitom analizom aplikacije. Željeli bismo dodati „Studentski fond“ kod načina plaćanja gdje bi korisnik mogao uplatiti neki mjesečni, polugodišni, godišnji iznos sa kojega bi se odbijao iznos narudžbe i ne bi morao razmišljati da li je platio, ima li dovoljno sredstava na kartici, hoće li imati dovoljno sredstava sljedeći mjesec i sl. (npr. uplati 1000kn, iznos narudžbe je 15kn i s fonda mu se odbije tih 15kn, te mu je preostalo 985kn). To bi bilo idealno za korisnike koje još uvijek financiraju roditelji bi im tako mogli uplatiti određeni iznos za hranu pa ne bi dolazilo do toga da student dobije novac za hranu koji uopće na kraju ne ode na hranu pa do kraja mjeseca gladuje. Ovo bi ujedno moglo biti i jedna vrsta ograničenja i kontrole. Također, uveli bismo sakupljanje kupona na određeni broj dostava pa npr. na 15 dostava korisnik ima jednu dostavu gratis i sl. Ideja nam je napraviti i okvir kod svakog menu-a u kojem bi se nalazili svi sastojci koji se nalaze u istom jer jako puno korisnika je alergično na određeni sastojak ili ga ne prakticira pa bi tako mogao i pratiti određenu količinu u svakom menu-u. Povezali bismo aplikaciju s internetom kako bi korisnik mogao pratiti gdje se dostavljač nalazi i za koliko vremena stiže na odredište.

# RAZRADA FUNKCIONALNOSTI

Aplikaciju će koristiti svi korisnici s AAI@EduHr elektroničkim identitetom. Prvo ćemo objasniti kako funkcionira naša aplikacija na use-case dijagramu i na koji način se obavlja interakcija korisnika i aplikacije. Slijede dijagrami slijeda u kojima ćemo spomenuti detaljnije pojedine funkcije koje sačinjavaju aplikaciju. Nakon toga će biti prikazan prototip sučelja za aplikaciju u kojem se razrađuju svi zahtjevi i sve funkcionalnosti, te prvi dojam kako bi aplikacija trebala izgledati. Na posljetku će biti prikazan i objašnjen klasni dijagram za objekte iz domene aplikacije.

## Use-Case dijagram

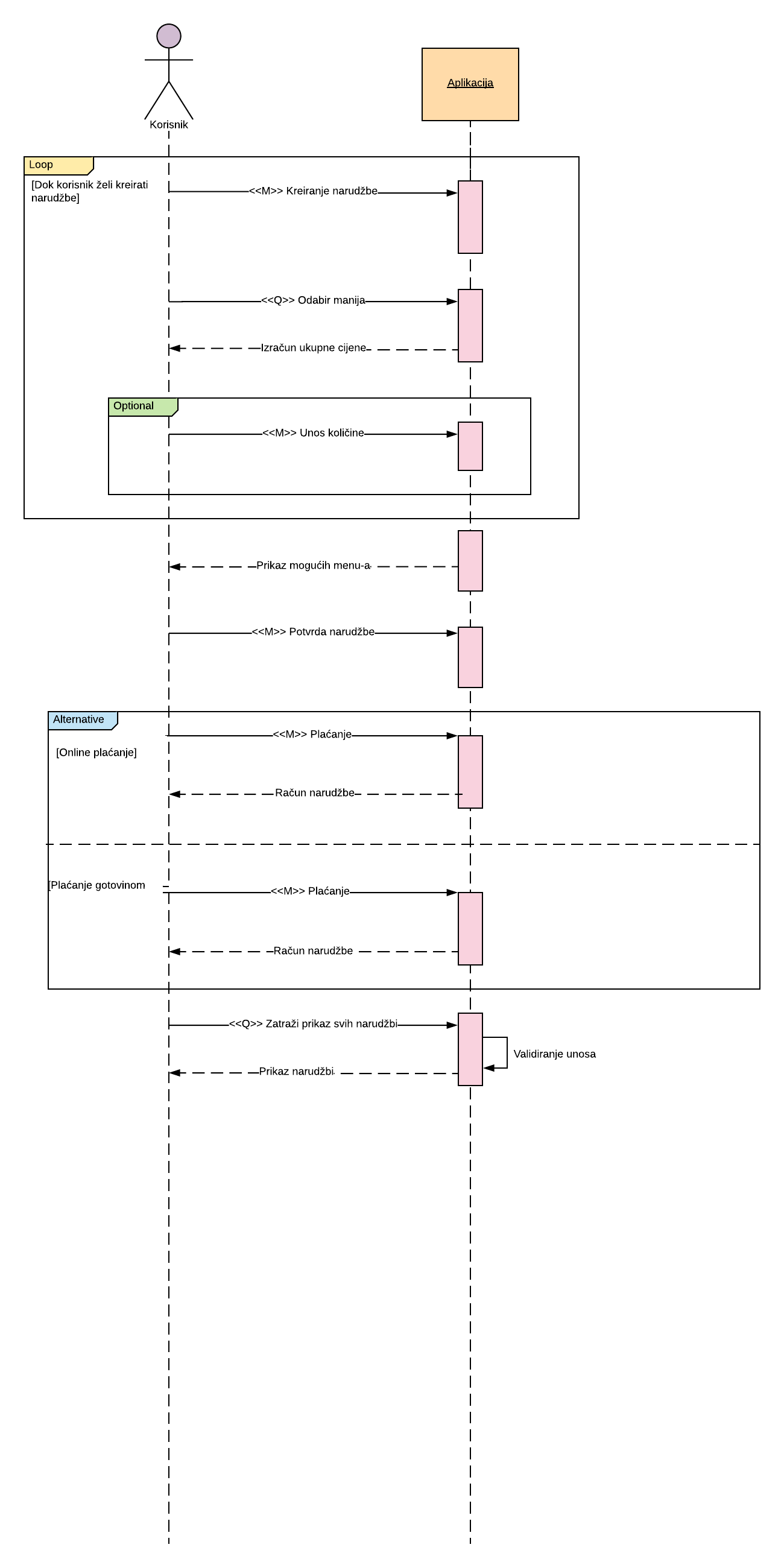


Slika 2. Use-Case dijagram

Dijagram slučajeva korištenja (eng. Use Case Diagram) opisuje što sustav radi, s pozicije vanjskog promatrača. Pritom valja napomenuti da nama kod izrade dijagrama slučajeva korištenja nije važno kako sustav funkcionira iznutra, već je u prvi plan stavljena korisnikova interakcija sa sustavom, te prikaz same interakcije.

Iz priloženog sustava možemo vidjeti da sustva koristi korisnik te da ima mogućnost pregled menu-a i pregled narudžbi. Kod pregleda narudžbi, korisnik može kreirati novu narudžbu ili pregledati stavke narudžbe. Ukoliko kreira novu narudžbu, svojom potvrdom odlazi na način plaćanja gdje mu sustav nudi odabir između gotovinskog ili online plaćanja. Pri gotovinskom plaćanju korisnik dovršava svoju transakciju kod dostavljača kada njegova dostava stigne na točnu adresu, a kod online plaćanja ima mogućnost kartičnog ili PayPal plaćanja. Kod kartičnog plaćanja izabire jednu od 4 vrste kartica koje banka nudi, a kod PalPal-a se prijavljuje na svoj račun s kojeg mu također banka skida iznos narudžbe.

## Sequence dijagram

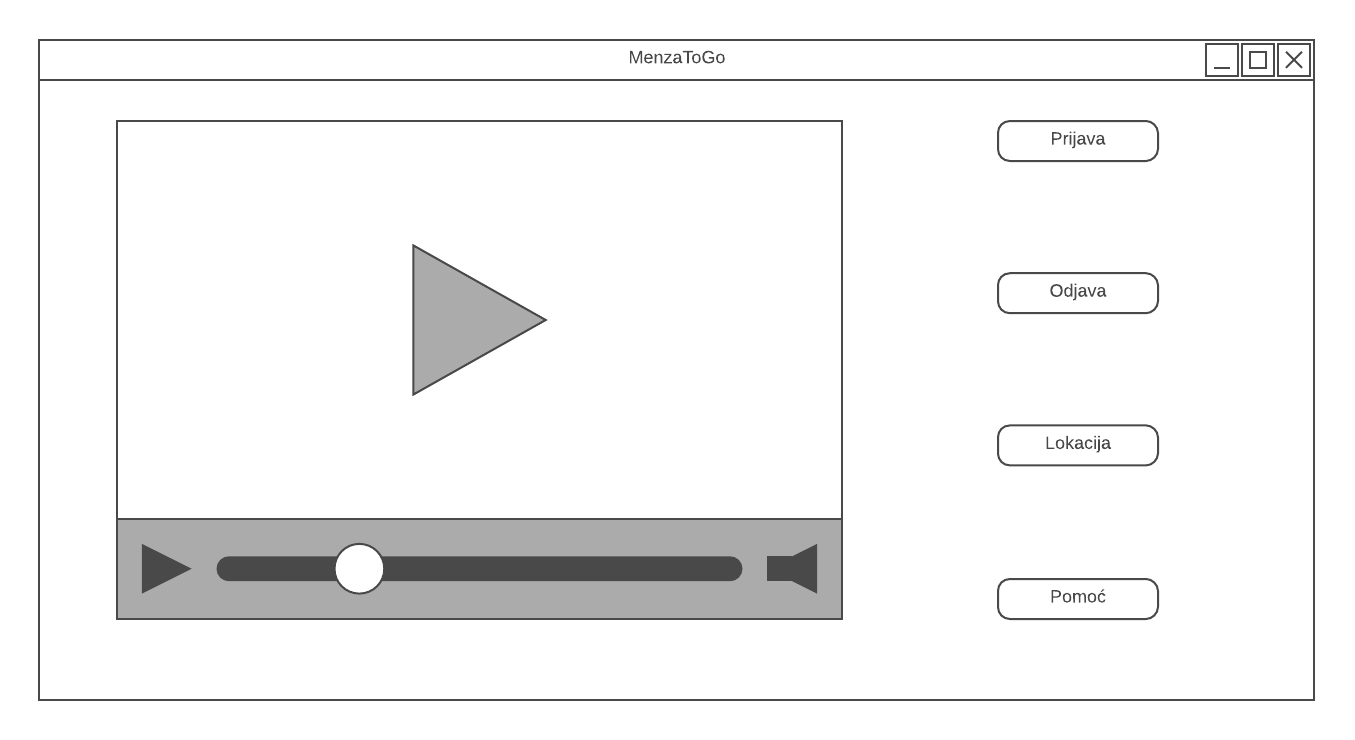


Slika 3. Sequence dijagram

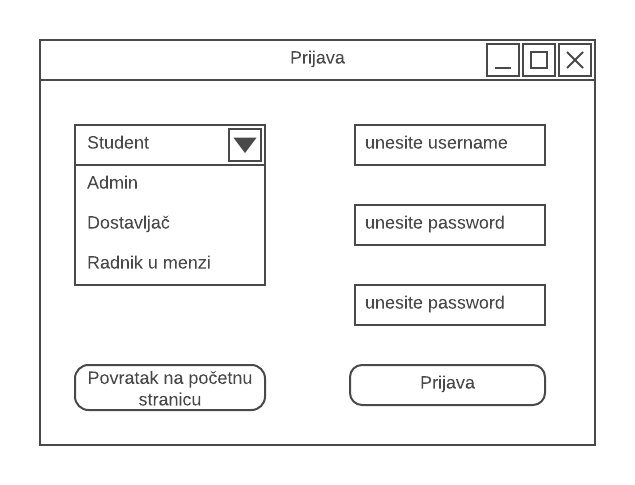
Kroz Use Case Sequance dijagram ćemo prolaziti kroz funkcionalnosti, a u ovom projektu jedna od ključnih funkcionalnosti je kreiranje narudžbe. Prilikom crtanja dijagrama slijeda, objekti koji sudjeluju u razmjeni podataka smješteni su na samom vrhu dijagrama. Učesnik koji pokreće akciju smješteni je skroz lijevo, dok ostale objekte koji igraju sporednu ulogu smještamo desno od učesnika. Kao što je napomenuto i ranije, objekti imaju vertikalnu liniju života. Ta vertikalna linija reprezentira vrijeme i ono prolazi silazno u dijagramu. Spomenuli smo na početku da promatramo izmjenu poruka. Ta izmjena poruka je prikazana na dijagramu strelicama između vertikalnih linija života. Poruke koje se izmjenjuju, tj. šalju i primaju kreću se u skladu s vremenom od vrha prema dnu.

Nakon prijave pomoću AAI@EduHr elektroničkim identitetom korisnik dolazi na početnu stranicu gdje kreira narudžbu tako da odabere željeni menu i sadržaj koji želi naručiti. Tijekom odabira istog, sustav mu automatski vraća ukupan iznos narudžbe tako da korisnik u bilo kojem trenutku kreiranja narudže može znati koliko će njegova ukupna narudžba iznositi s troškom dostave. Kada odabere sadržaj, slijedi unos količine. Ukoliko je korisnik zaboravio što je ponuđeno u svakom menu-u, sustav mu nudi prikaz menu-a u realnom vremenu. Korisnik je kreirao svoju narudžbu do kraja te odlazi na sljedeći korak, no prije toga sustav traži od korisnika da potvrdi svoj odabir narudžbe. Kada korisnik potvrdi narudžbu, dolazi do načina plaćanja gdje može birati između online ili gotovinskog plaćanja. Odabirom načina plaćanja korisniku se izdaje narudžbenica (račun) na kojoj može vidjeti sve podatke koje je odabrao u sustavu. Na samom kraju, korisnik ima pravo od sustava zatražiti pregled svih svojih narudžbi do sada ili se odjaviti/odabrati ponovan povratak na početnu stranicu za izradu nove narudžbe.

## Prototip

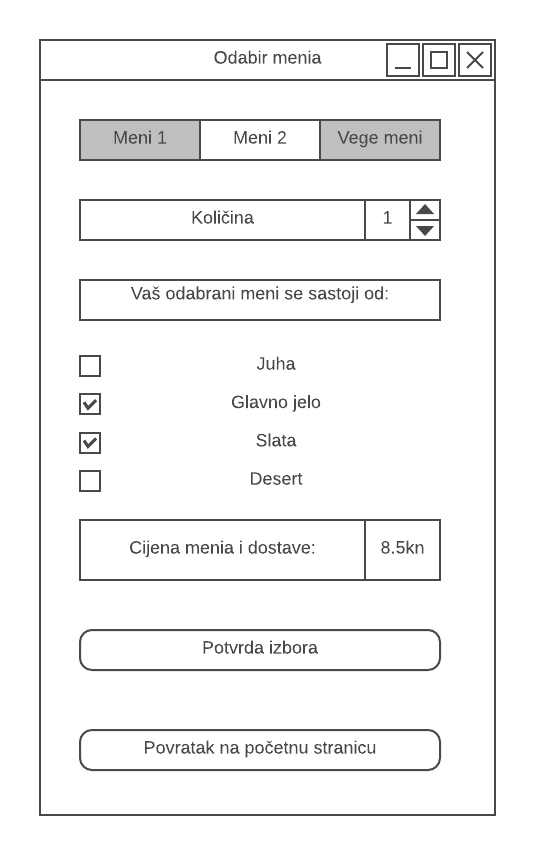


Slika 4. Prikaz prototipa pri ulasku u aplikaciju



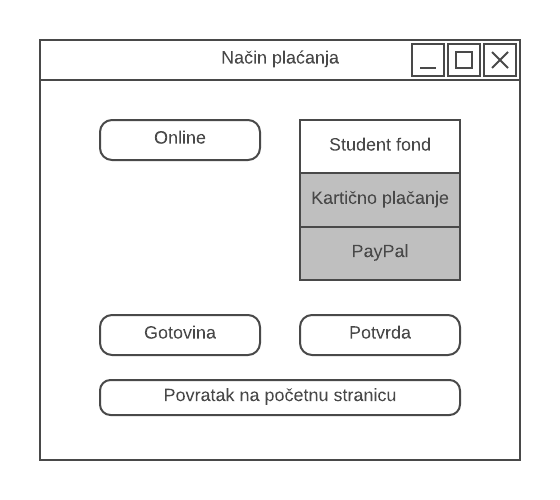
Slika 5. Prikaz prototipa obrasca za prijavu

Slika 4. i 5. nam prikazuju prototipe pri ulazak u aplikaciju i prijave. U našoj završnoj aplikaciji odlučili smo se da se na početku aplikacije nalazi samo prijava bez odjave zbog toga što je odjava ponuđena na kraju aplikacije, a i korisnik se ne može odjaviti dok nije prijavljen. Također, napravili smo okvire za upis email-a i lozinke bez mogućnosti odabira statusa korisnika iz tog razloga što svaki korisnik ima svoj AAI@EduHr elektronički identitet i svi njegovi podaci su u bazi ukoliko su točni tako da se svi mogu s time prijaviti. U prijavi provjeravamo da li je unešen e-mail važeći, točnije da li su e-mail i lozinka u bazi podataka, ako ne, izbacuje se poruka da korisnik nije pronađen. Isto tako, radimo provjeru ukoliko je prazno polje. Uz samu prijavu se nalazi Sveučilišni video u kojem se prikazuje vrijeme studenata u Studentskome domu, restoranu i sl. Video bi pomogao novim korisnicima (npr. brucošima) kojima je cijelo Sveučilište novo i nepoznato.



Slika 6. Prikaz prototipa za odabir menu-a

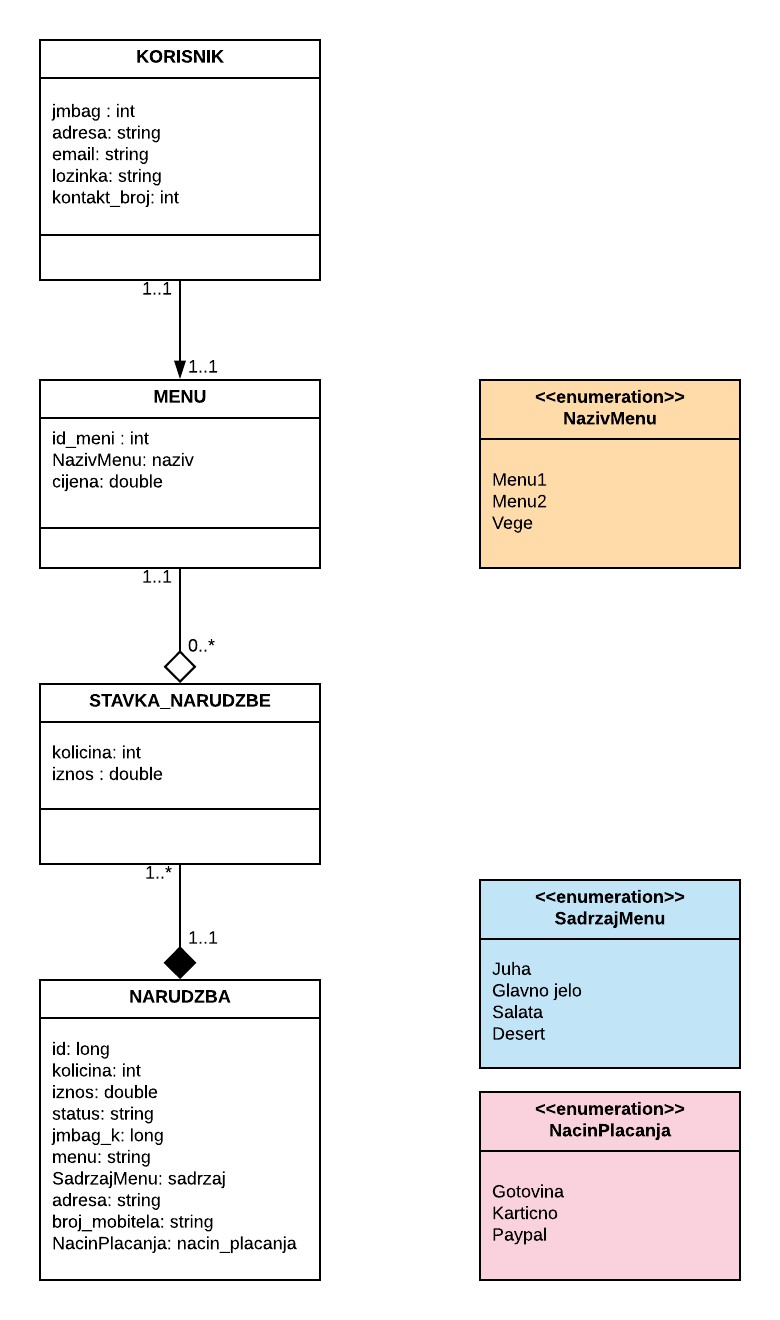
Slika 6. prikazuje prototip obrasca za odabir menu-a koji se otvara kao početna stranica aplikacije nakon što se korisnik prijavi u sustav. Korisniku su na raspolaganju tri vrste menu-a u kojima bira želi li sav sadržaj ili može odabrati samo onaj sadržaj koji ga interesira. Korisnik može odabrati količinu koju želi naručiti te nakon toga korisnik potvrđuje svoj odabir prelaskom na novi prozor.



Slika 7. Prikaz prototipa odabira načina plaćanja

Na samom kraju, korisnik odabire hoće li platiti svoju narudžbu gotovinom ili online plaćanjem. Ukoliko odabere gotovinsko plaćanje, sustav mu izadaje narudžbenicu koja odlazi u bazu podataka te korisnik narudžbu plaća pri dolasku dostavljača. Ukoliko odabere kartično plaćanje, otvara se novi obrazac za upis podataka o kartici te odabiru kartice i iz banke se skida iznos narudžbe. Kod PayPal plaćanja, otvara se obrazac za prijavu u PayPal račun te se sve šalje na banku koja također odbija iznos narudžbe. Studentski fond bi trebao funkcionirati tako da korisnik uplati određeni iznos u fond (npr. 2000,00kn) i svaki put mu se iznos narudžbe odbija sa studentskog fonda pa samim time više ne treba razmišljati hoće li imati dovoljno na kartici ili PayPalu. Na kraju svakog odabira se izbacuje narudžbenica (račun) sa svim podacima koje je korisnik odabrao.

## Klasni dijagram



Slika 8. Prikaz dijagrama klasa

Valja napomenuti da dijagram klasa ili razreda ne opisuje događaje, stanja, aktivnosti ili bilo kakvu vremensku promjenjivu karakteristiku sustava koji se modelira. Klasni dijagram opisuje način na koji su klase povezane u našem programu, tj. predstavlja skup elemenata strukture kao što su klase, sučelja, objekti i naravno veze između samih klasa. Dijagram se sastoji od klasa Korisnik, Menu i Narudzba. Sve klase sadrže ID koji se povlači iz baze podataka i po njima se može pretraživati jednako točno kao i u bazama podataka. Jednom kada iz baze povučemo sve podatke više ne moramo čitati iz nje. Novi se podaci jednostavno stavljaju u liste podataka koji su pročitani iz baze.

Klasa Korisnik sadrži referencu na klasu Menu, ujedno je i klasa koja se povezuje s bazom podataka, te obrađuje i izvršava zadane uvjete (upite) koji manipuliraju bazom podataka. Menu sadrži atribut nabrajanja NazivMenu koji može poprimiti tri vrijednosti: Menu1, Menu2 i Vege. Veza klase Korisnik prema klasi Menu je 1 i samo 1 što bi značilo da točno jedan korisnik može odabrati točno 1 menu pri izvršavanju jedne narudžbe. Točno 1 menu može imati niti jednu ili više stavki narudžbe, a jedna ili više stavki narudžbe može sadržavati točno 1 narudžba. Narudžba sadrži atribute nabrajanja SadrzajMenu koji sadrži četiri stanja: Juha, Glavno jelo, Salata i Desert, te NacinPlacanja koji sadrži 3 vrijesnosti: Gotovina, Kartično i PayPal. Narudžba povezuje sve podatke narudžbe i sprema je za za slanje narudžbe na samu izlaznu jedinicu aplikacije. Prima podatke sadržane u košarici te podatke koje je korisnik sam upisao. Ona ažurira bazu podataka svaki puta kada se neka narudžba izda.

Entitet StavkaNarudzbe postoji kako bi se riješio problem veze više-više koja je potrebna da se prikaže veza entiteta Menu s entitetom Narudžba.

Koristimo kompoziciju jer stavka narudžbe mora biti u narudži s obzirom da nema smisla da ne pripada nigdje. Takva veza označava da je jedan objekt sastavljen od više drugih, podređenih i ako se on obriše sve veze se kidaju i podređeni objekti također. To znači da podređeni objekti ovise o glavnom. Također, koristimo i agregaciju jer stavka narudžbe sadrži menu.

# IMPLEMENTACIJA

Kroz ovu točku opisati ćemo kako smo implemetirali aplikaciju kroz kod. Aplikacija se bazira na klasama koje se podudaraju s bazama podataka te služe kako bi se podaci mogli prezentirati korisniku. U nastavku slijedi tablica glavnih klasa i njihovih sastavnica kao i funkcionalnosti koje one omogućuju.

**Naziv klase Sastavnice Funkionalnosti**

|  |  |  |
| --- | --- | --- |
| Korisnik.cs | private long jmbag;  private string adresa;  private string email;  private string lozinka;  private int kontaktBroj; | * Služi za punjenje liste korisnika * Sadrži sve podatke o korisniku koji se nalaze u bazi podataka kako bi se korisnik mogao prijaviti s valjanim elektorničkim identitetom |
| Menu.cs | private long id;  private string naziv;  private double cijena; | * Služi za spremanje i prepoznavanje različitih vrsta menu-a pri naručivanju |
| Narudzba.cs | private long id;  private int kolicina;  private double iznos;  private string status;  private long jmbag\_k;  private string menu;  private string sadrzaj;  private string adresa;  private string broj\_mobitela;  private string nacin\_placanja; | * Služi za punjenje liste narudžbi * Koristi se u listi za držanje svih narudžbi u bazi podataka |

Tablica 1. Prikaz glavnih klasa

Gore navedene klase su klase koje omogućuju ponajprije čuvanje podataka koje dohvaćamo iz baze. Nakon što podatke dohvatimo iz baze podatake i pohranimo ih u odgovarajuće objekte, moramo ih nekako prikazati korisniku. Podatke ispisujemo određenim funkcijama koje se nalaze u klasama koje ćemo prikazati u nastavku. Te klase imaju posebnu svrhu u radu aplikacije jer ne služe niti za prikaz niti za držanje podataka već pomoću njih dolazimo samo do funkcija s kojima baratamo tijekom rada aplikacije.

**Naziv klase Sastavnice Funkionalnosti**

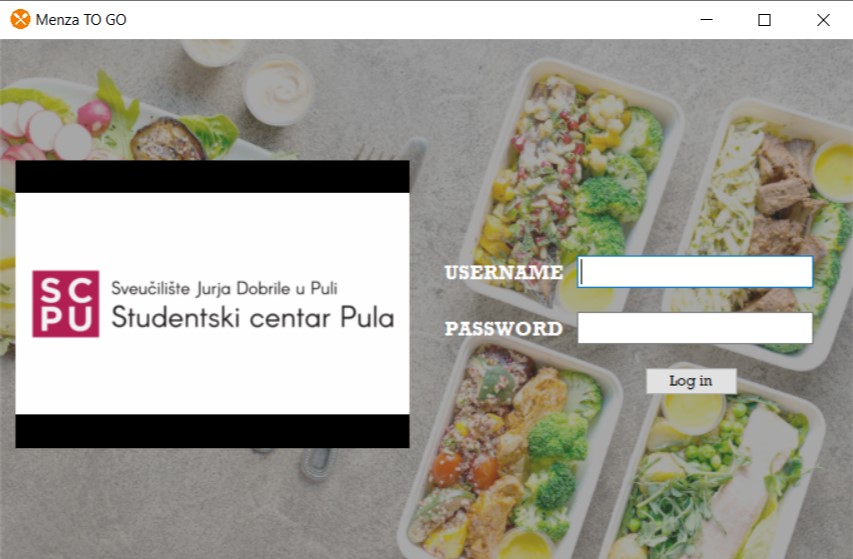
|  |  |  |
| --- | --- | --- |
| BP.cs |  | * Ova klasa je statična i služi povezivanju aplikacije s bazom podataka * U njoj sadržimo metode za otvaranje i zatvaranje konekcije, mogu se pozvati iz bilo kojeg kuta aplikacije i služe kako bi se pozivale različite procedure iz baza podataka |
| BPkorisnik.cs | public static void Spremi(Korisnik k);  public static List<Korisnik> DohvatiSve();  public static List<Korisnik> DohvatiPrijava(string email, string lozinka); | * Pomoću ovih metoda spremamo korisnika u bazu podataka * Možemo dohvatiti listu svih korisnika koji se nalaze u bazi i onih koji su prijavljeni |
| BPnarudzba.cs | public static void Spremi(Narudzba n);  public static List<Narudzba> DohvatiSve();  public static int DohvatiBroj();  public static List<Narudzba> Dohvati(long jmbag); | * Ova klasa nam služi za spremanje i dohvaćanje korisnikovih narudžbi iz liste * Također, imamo i metodu kojom manipuliramo da nam se za svakog korisnika broji koja je narudžba po redu * Pored svih narudžnni spremljenih u bazi, možemo dohvatiti i narudžbe koje je napravio samo trenutni korisnik |

Tablica 2. Prikaz klasa povezanih s bazom podataka

Kod obje klase, Dohvati() i DohvatiSve() uzimaju najažurnije verzije popisa koji su potrebni za prvu opciju aplikacije i proslijeđuje ih aplikaciji. Preuzimaju parametre dobivene od aplikacije i nakon toga se uvrštavaju u upit koji vraća rezultate. Te rezultate je potrebno poslati nazad aplikaciji kako bi se prikazali korisniku.

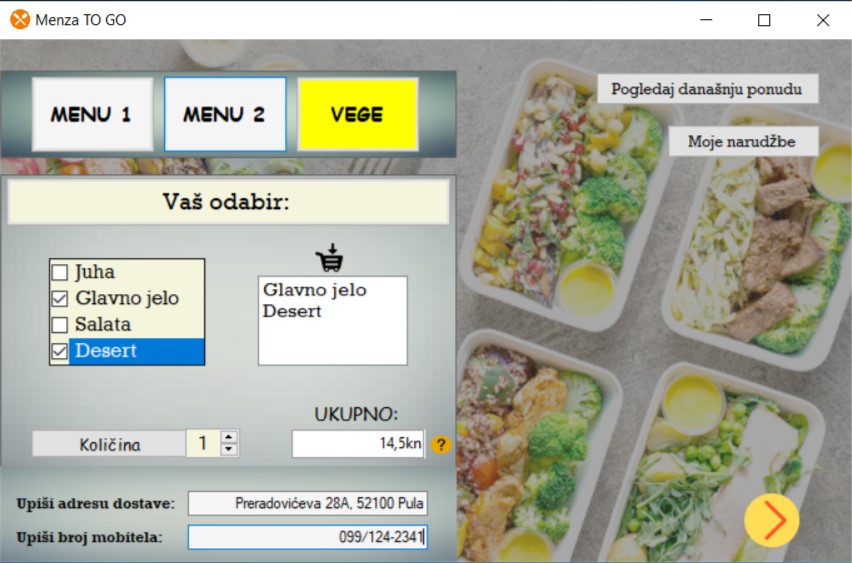
# KORISNIČKE UPUTE

Prije samog ulaska u glavni okvir aplikacije potrebno se prijaviti putem AAI@EduHr elektroničkog identiteta.



Slika 9. Prikaz prozora za prijavu

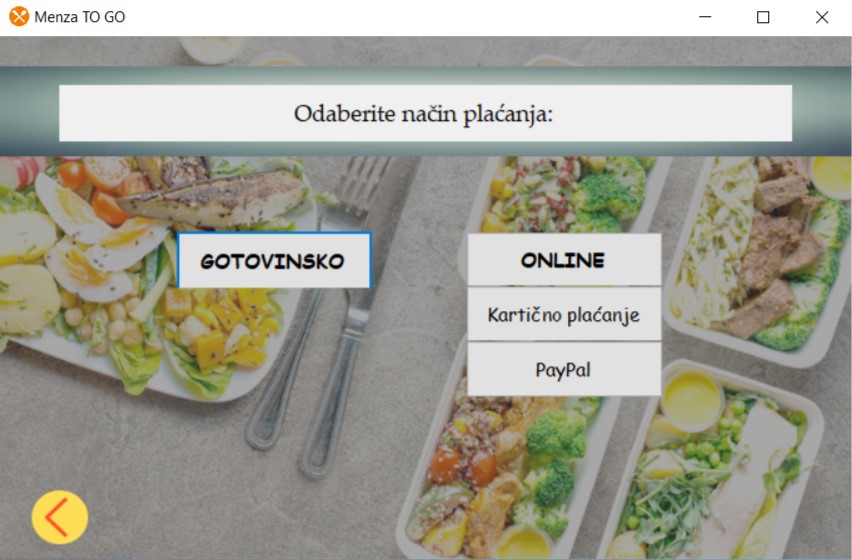
Prilikom prijave korisnik unosi svoj e-mail i lozinku koji se nalaze u bazi podataka svih studenata i zaposlenika na Sveučilištima u Hrvatskoj. Ukoliko korisnik ne unese valjano korisničko ime ili lozinku, sustav mu izbacuje poruku da je upisano krivo korisničko ime/lozinka. Također, ukoliko korisnik ništa ne upiše sustav izbacuje poruku da je korisničko ime/lozinka nisu unešeni. Korisniku se prikazuje Sveučilišni video. Video može doprinjeti npr. brucošima koji nisu još dovoljno upoznati sa Studentskim restoranom, domom, centrom i sl. jer na njemu mogu vidjeti kako sve skupa funkcionira.



Slika 10. Prikaz početnog prozora za kreiranje narudžbe

Nakon prijave u sustav, korisniku se nudi mogućnost odabira jednog od tri menu-a. Možemo vidjeti sa slike 10. da je element menu-a žute boje nakon što korisnik odabere isti. Ukoliko korisnik ne odabere jedan od tri menu-a sustav izbacuje poruku u kojoj navodi da je potrebno odabrati menu. Korisniku su ponuđeni sadržaji menu-a tako da ne mora naručiti cijeli menu ukoliko ne želi (što je ujedno i prednost s obzirom da možda nešto ne jede i onda se hrana bezveze baca). Sustav provjerava da li je korisnik označio barem jedan sadržaj jer ukoliko nije prikazuje poruku da je obavezno odabrati sadržaj menu-a. Pri označavanju sadržaja, isti se dodaje u košaricu na desnoj strani te se ujedno automatski izračunava ukupna cijena narudžbe. Početna cijena je odmah postavljena na 10,00kn, a ukoliko korisniku nije jasno zašto piše takav iznos prije odabira sadržaja, može pogledati prelaskom na upitnik sa strane gdje se nalazi objašnjenje kako je dostava uračunata u cijenu narudže. Još jedna od mogućnosti je promjena količine pa tako korisnik može naručiti više od jednog, ali najviše pet menu-a u jednoj kreiranoj narudžbi. Promjenom količine automatski se mijenja i cijena ali samo za menu, ne i za dostavu jer je ona uvijek fiksna. Nadalje, korisnik ne može prijeći na novi korak dok ne upiše adresu na koju želi da mu se menu dostavi te broj mobitela kako bi ga dostavljač mogao kontaktirati. Ukoliko jedan od okvira za adresu i broj mobitela ostane prazan, sustav prikazuje poruku da je obavezno popuniti oba polja.

U prozoru nudimo i mogućnost pregleda trenutnih menu-a koji su ponuđeni ukoliko korisnik ne zna što se nalazi npr. pod glavnim jelom. Kada korisnik zatraži pregled današnje ponude, sustva ga spaja na stranicu od Studentskog centa gdje se nalazi ponuda sadržaja svakog menu-a te korisnik može paralelno raditi narudžbu. U slučaju da korisnik želi npr. provjeriti koliko je potrošio u tekućem mjesecu na prehranu, može odabrati pregled svih svojih narudži i vidjeti sve podatke o prethodnim narudžbama. Nakon svih točno odabranih i upisanih podataka, korisnik prelazi na novi prozor.



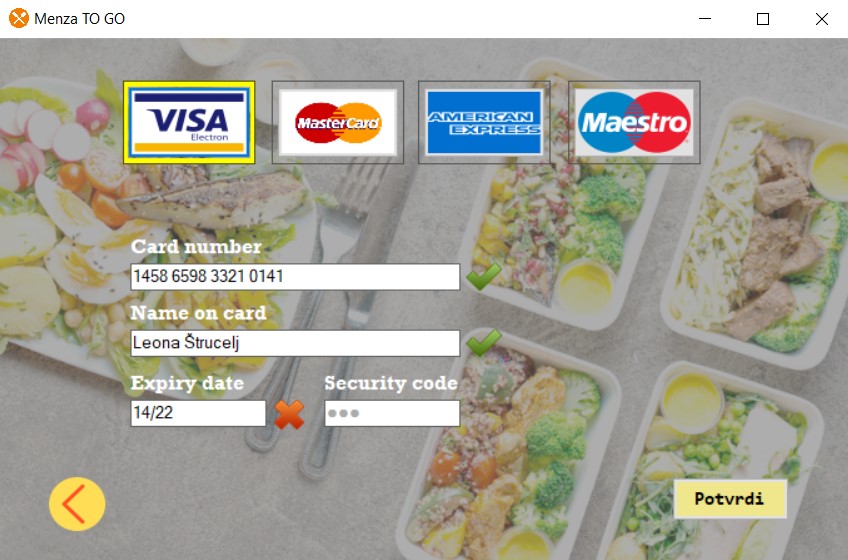
Slika 11. Prikaz prozora za odabir načina plaćanja

Slijedi odabir načina plaćanja kod kojeg korisnik ima dvije mogućnosti: plaćanje gotovinom ili online. Ukoliko odabere gotovinsko plaćanje, sustav mu izbacuje narudžbenicu, točnije potvrdu narudžbe. Kod online plaćanja korisniku se nudi plaćanje PayPal računom ili kartično.



Slika 12. Prikaz plaćanja PayPal računom

Prilikom odabira plaćanja PayPal računom, od korisnika se traži unos korisničnog imena i lozinke koju koristi za prijavu na PayPal račun. Ako korisnik ne upiše u jedan ili oba okvira potrebne podatke, sustav izbacuje poruku da su korisnički podaci neispravni te da ponovi upis. Točnim upisom korisničkom imena i lozinke, korisnik se prijavljuje na Paypal račun koji je povezan s bankoma i izvršava plaćanje nakon čega dobiva potvrdu o uspješnom plaćanju, tj. narudžbenicu. Ukoliko se korisnik predomisli i želi ipak platiti narudžbu kartično, može stisnuti „Pay with debit or Credit Card“ gdje ga sustav vodi na prozor za kartično plaćanje.



Slika 13. Prikaz kartičnog plaćanja

Kod kartičnog plaćanja korisnik odabire vrstu kartice s kojom želi izvršiti transakciju. Ukoliko ne odabere karticu, sustav će mu izbaciti poruku da mora odabrati karticu za plaćanje.

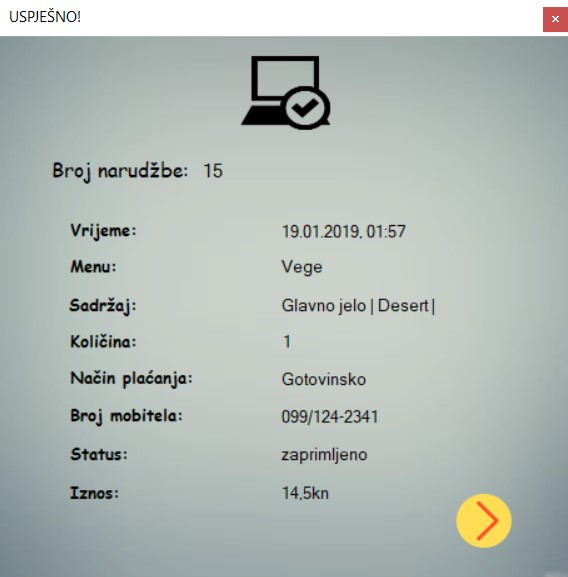
Nadalje, korisnik unosi 16-znamenkasti broj s kartice. Sustav provjerava da li se unešeni broj sastoji od 16 znamenki od 0-9. Ukoliko je sve u redu, sustav izbacuje zelenu kvačicu kao znak potvrde točnog unosa. Ukoliko korisnik nije dobro upisao, sustav izbacuje crveni X kao znak da je upis netočan ili nepotpun.

Korisnik unosi ime i prezime s kartice. Sustav provjerava da li je unešeno ime i prezime u dobrom formatu. Ukoliko korisnik upiše samo ime ili samo prezime, sustav će mu izbaciti X.

Slijedi upis datuma i godine isteka kartice koji mora biti u formatu koji je napisan kao primjer prije unosa podatka u okvir. Sustav provjerava da li je upisani mjesec manji od 13 s obzirom da godina traje 12 mjeseci te da li je godina veća od 2030 jer korisnik ne može dobiti karticu na više od 10 godina.

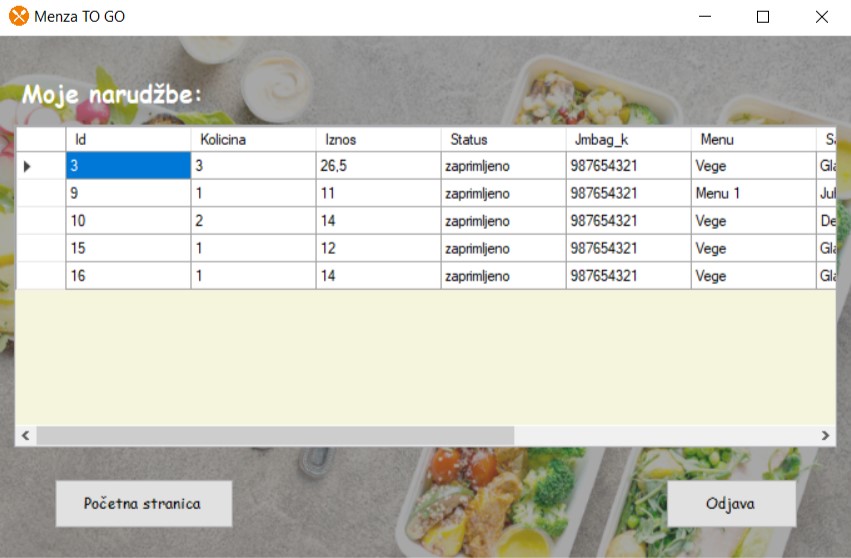
Na samom kraju upisa, od korisnika se traži da upiše 3-znamenkasti kod koji se nalazi na poleđini kartice.

Po završetku unosa, sustav provjerava da li su svi podaci ispravni, točnije da li korisnik ima 4 zelene kvačice kao potvrdu za točan unos podataka. Zadovoljeći sve uvijete kod kartičnog plaćanja, korisnik odabire naplatu i izdaje mu se potreba o narudžbi.



Slika 14. Prikaz potvrde o narudžbi (narudžbenice)

Nakon uspješno kreirane narudžbe korisnik dobiva od sustava narudžbenicu kao potvrdu. U njoj može vidjeti sve potrebne podatke, od vremena kad je narudžba provedena, vrste i sadržaja menu-a, količine, načina plaćanja (koji je bitan za dostavljača kako bi znao da li određena narudžba mora biti plaćena ili samo dostavljena), broja mobitela (također potreban dostavljaču kako bi ga mogao kontaktirati npr. pri dolasku na adresu) te na kraju ukupan iznos narudžbe.



Slika 15. Prikaz prikaza svih korisnikovih narudžbi

Prelaskom na sljedeći korak nakon potvrde o uspješno izvršenoj narudžbi korisniku se ispisuju sve njegove narudžbe koje je izvršio od prvoga puta ulaska u aplikaciju. Korisnik tako može pratiti npr. mjesečni trošak prehrane i sl.

Ukoliko želi izvršiti novu narudžbu može se vratiti na početnu stranicu odabirom na element u lijevom kutu slike 15., a u suprntnom mu preostaje mogućnost odjave iz sustava gdje ga sustav šalje ponovno na prozor za prijavu.

# POPIS SLIKA I TABLICA

Slika 1. SWOT analiza aplikacije

Slika 2. Use-Case dijagram

Slika 3. Sequence dijagram

Slika 4. Prikaz prototipa pri ulasku u aplikaciju

Slika 5. Prikaz prototipa obrasca za prijavu

Slika 6. Prikaz prototipa za odabir menu-a

Slika 7. Prikaz prototipa odabira načina plaćanja

Slika 8. Prikaz dijagrama klasa

Slika 9. Prikaz prozora za prijavu

Slika 10. Prikaz početnog prozora za kreiranje narudžbe

Slika 11. Prikaz prozora za odabir načina plaćanja

Slika 12. Prikaz plaćanja PayPal računom

Slika 13. Prikaz kartičnog plaćanja

Slika 14. Prikaz potvrde o narudžbi (narudžbenice)

Slika 15. Prikaz prikaza svih korisnikovih narudžbi

Tablica 1. Prikaz glavnih klasa

Tablica 2. Prikaz klasa povezanih s bazom podataka