Gregor Stojanović – 2018200935 i Marko Dombaj - 2018200886

gregor.stojanovic.18@singimail.com marko.dombaj.18@singimail.rs

Veb Aplikacija za pretragu recepata

Predmet: Projektovanje Softvera

1. **Opis Projekta:**

Veb Aplikacija za Pretragu Recepata je veb aplikacija koja je namenjana da korisnicima pomogne u njihovim kulinarskim poduhvatima. Uz pomoć administratora koji održavaju, dopunjuju I dodaju recepte korisnici mogu bez brige I sa uživanjem da spremaju svoja jela. Pretraga I pregled recepata preko sastojaka omogućava korisnicima pogodno snalaženje I nalaženje recepata za koje imaju odgovarajuće sastojke.

1. **Razvoj Softvera:**
   1. **Metoda modelovanja**

Za izradu ovog projekta odlučili smo se za korišćenje agilne metode rada. Ovu odluku smo doneli zbog nekiliko različitih razloga gde su glavni vremensko ograničenje I primećene potražnje tržišta (prakse koje smo glledali I za koje smo se prijavljivali su tražile poznavanje Agilnih metoda rada). Veća komunikacija između članova tima nam je pomogla u boljem I bržem razumevanju zahteva.

* 1. **Plan projekta**

Kroz analizu projekta izdvojili smo 4 grupe zahteva. Svaka od grupa ima svoje zadatke i manje celine, ali za same grupe, u našem timu, je zadužen po jedan član tima.

* *Baza podataka* [Gregor Stojanović]*:*
  + Kreiranje modela baze podataka
  + Popunjavanje baze podataka test podacima
  + Kreiranje i provera SQL upita
  + Čišćenje baze podataka
* *Beckend* [Gregor Stojanović]*:*
  + Implementacija CRUD-a:
    - Dodavanje novih recepata, sastojaka i kategorija
    - Brisanje recepata
    - Izmena recepata
    - Čitanje recepata
  + Implementacija prijave korisnika:
    - Provera postojanja korisika
  + Implementacija Registracije novog korisnika:
    - Provera Ispravnosti unetih podataka
    - Kreiranje novog korisnika (ako su podaci ispravni)
* *Frontend* [Marko Dombaj]*:*
  + Dizajn aplikacije
    - Kreiranje mockup-a
    - Kreiranje template-a
  + Povezivanje sa beckend-om
* *Testiranje* [Marko Dombaj]

1. **Analiza Zahteva:**
   1. **Zahtev**

Aplikacija treba da omogući administratorima da postavljaju recepte za razna jela. Samo prijavljeni korisnici, koji se na portal prijave sa ispravnih parametrima naloga administratora sadržaja mogu da pristupe administrativnom panelu portala. U ovom panelu mogu da dodaju novi recept koji se sastoji od naslova, slike, detaljnog opisa postupka pripreme, kao i liste sastojaka koja se, zajedno sa količinom, formira u vidu tagova, tj. obeležja. Svaki sastojak, ako postoji u listi tagova je uvezuje kao jedan od sastojaka koji su potrebni za novi recept. Uz formiranje veze između recepta i sastojka se čuva i podatak o količini sastojka. Ovakve veze se prave za svaki od sastojaka. Ako sastojak ne postoji u bazi, tj. ne postoji taj tag, prvo se kreira zapis u adekvatnoj tabeli, pa se onda kreira veza.

Sa korisničke strane aplikacije, prikazuje se lista recepata sa slikom i nazivom recepta. Svaki recept može da bude upisan u jednu ili više kategorija, što se vrši prilikom dodavanja recepta u bazu od strane administratora. Kada korisnik otvori recept, recept treba da bude prikazan na pregledan način, a lista sastojaka se generiše dinamički na osnovu tagova i količina pripisanih za svaki tag (sastojak). Kada korisnik klikne na neki od tagova u listi, aplikacija treba da ga odvede na listu svih recepata jela u čiji sastav ulazi taj sastojak, tj. za koji je to tag. Treba implementirati i vrstu pretrage gde korisnik može da navede više tagova, tj. sastojaka za koje želi da vrši pretragu baze podataka recepata i aplikacija treba da prikaže samo one recepte za koje su potrebni svi od navedenih sastojaka, tj. sa kojima je vezan svaki od navedenih tagova i to da budu sortirani na taj način da se prikazuju prvo recepti u koje ulazi najveća količina navedenih sastojaka (suma). Grafički interfejs veb sajta treba da bude realizovan sa responsive dizajnom.

* 1. **Uloge**

Iz teksta zahteva uspeli smo da izvučemo dve eksplicitno pomenute uloge koje treba implementirati. Pored dve eksplicitno navedene uloge možemo zaključiti da postoji još jedna ne navedena uloga.

Uloge koje smo izdvojili su:

* Administrator
* Korisnik
* Posetilac (nije eksplicitno navedeno)
  1. **Mogućnosti po ulogama**

Iz zahteva možemo da izvučemo nekoliko različitih funkcionalnosti naše aplikacije. Funkcionalnosti su podeljene po ulogama navedenim prethodno. Pored eksplicitno navedenih funkcijonalnosti kao što je dodavanje novih recepata I izmena postojećih koje moramo da razložimo na manje I jednostavnije celine postoje funkcionalnosti koje smo morali da pretpostavimo kao što je Registracija novih korisnika I postoje neke funkcionalnosti koje smo morali sami da smislimo I dodamo. Jedna od funkcionalnosti je Random recept, ona izvlači jedan random recept I na taj način omogućava posetiocima da vide šta naša aplikacija radi ali za punu funkcionalnost je neophodna registracija, takođe korisnicima omogućava da saznaju za neki novi recept.

Funkcionalnosti po ulogama mogu se videte ovde:

Dodavanje Kategorije Recepta

Dodavanje Podataka za Recept

Brisanje Postojećeg Recepta

Izmena Postojećeg Recepta

Dodavanje Novog Recepta

Pretraga Recepata

Pregled Recepata

Pregled Random Recepta

Registracija Novog Korisnika

Administratorski Panel

Administratorski Panel

Prijava Korisnika

***Administrator:***

* Postavljanje recepata:
  + Podaci recepta
  + Kategorija recepta
* Izmena podataka postojećih recepata
* Brisanje recepata

***Korisnik:***

* Pregled postojećih recepata
* Pretraga recepata
* Pregled Random recepta

***Posetilac:***

* Registracija novog korisnika
* Prijava korisnika
* Pregled Random Recepta
  + 1. **Use case opis procesa**

*Postavljanje recepata:*

Postavljanje recepata je proces u kome Administratorski nalog preko Administratorskog panela daje podatke za recept u vidu popunjavanja forme. Neophodni podaci za recept koji moraju da budu pripremljeni su: Naziv jela, Slika jela kao i način pripreme.

Pored prethodno navedenih detalja neophodno je da se selektuju svi sastojci (tagovi) koji idu u recept koji se postavlja. Ukoliko ne postoji sastojak (tag) već zabeležen neophodno je dodati sastojak i onda ga selektovati. Za selektovane sastojke (tagove) je neophodno da se postavi konkretna količina koja se koristi u samom receptu.

Dodavanje kategorije recepta ima sličan proces kao i dodavanje sastojaka. Neophodno je selektovati najmanje jednu kategoriju kojoj recept pripada. Ukoliko ne postoji kategorija potrebno je dodati kategoriju i onda je selektovati za recept koji se postavlja.

Nakon popunjavanja svih podataka i provere ispravnosti istih treba pritisnuti taster „Create Recipe“. Ta akcija će poslati serveru zahtev za kreiranje novog recepta koju će server obraditi.

Posle obrađivanja zahteva server će poslati odgovor u vidu potvrde uspešno obrađenog zahteva ili u vidu greške pri kreiranju novog recepta.

* 1. **Analiza podataka**

Za čuvanje podataka smo izabrali relacione baze podataka. Koristili smo MySql u okruženju MySql workbench. Na osnovu zahteva zadatka I analize zahteva došli smo do modela baze podataka koji primenjujemo. Analizom smo zaključili da postoje 4 jake tabele I 2 vezne tabele.

**3.4.1 Podela podataka**

**Jake tabele:**

* *Users:*
  + ID
  + Username
  + Email
  + Password (HASH)
  + Profile picture
  + Is Administrator (True=1, False=0)
* *Recipe:*
  + ID
  + Recipe name
  + Food image
  + Preparation
* *Tags (ingridient):*
  + ID
  + Ingridient name
* *Categories:*
  + ID
  + Category name

**Vezne tabele:**

* *Recipe-Tag:*
  + ID
  + Recipe ID
  + Tag ID
  + Ingrident amount
* *Recipe Category:*
  + ID
  + Recipe ID
  + Category ID

1. **Projektovanje Sistema:**
   1. **Korišćene tehnologije, alati i softver**

Upotreba korišćenih tehnologija, programskih jezika I softvera je odabrana zbog jednostavnosti vremenskih ograničenja kao I poznavanja istih od strane članova tima za izradu projekta.

Korišćeno je:

* Za Mock-up:
  + NinjaMock
* Za menadžment I organizaciju:
  + Clubhouse.io
* Za kontrolu verzije I skladištenje fajlova projekta:
  + Git/GitHub (<https://github.com/MDombaj/Ispitni-zadatak-recepti>)
* Za Baze podataka:
  + MySql
  + MySql Workbench
* Za Beckend:
  + PHP
* Za Frontend:
  + JavaScript
  + HTML
  + CSS
  1. **Arhitektura Sistema**

Koritsti se Klijent-Server arhitektura, jer klijenti i administratori (nalozi klijenata i administratora) šalju zahteve serveru (za pregled, izmene, pretrage podataka...) , a server nazad šalje rezultate obrađenog zahteva.

**4.2.1 Dijagram arhitekture**

Stilizovanje odgovora servera

Funkcije slanja zahteva

...

Klijent

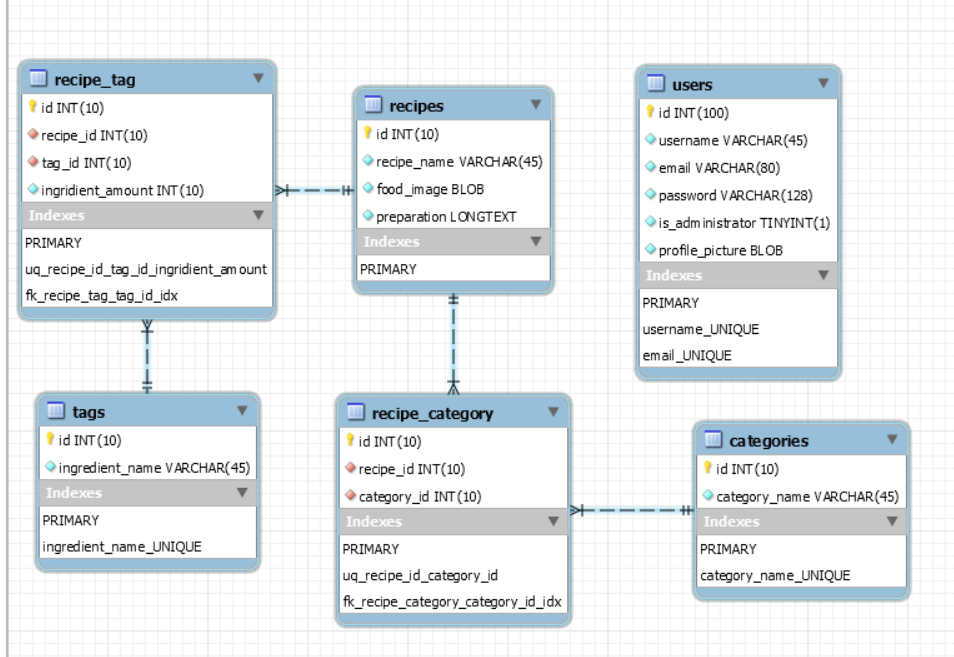
Server

Baza podataka

Funkcije za obradu zahteva

...

* 1. **Struktura programskih modula**
  2. **Projektovanje baze podataka (ERD dijagram baze podatakaka)**



1. **Implementacija Softvera:**
2. **Testiranje Softvera:**
3. **Isporuka Softvera:**