

Apoteka

**CS101**

Uvod u objektno orijentisano programiranje

Projektna dokumentacija

31.01.2020

Student: Bogdan Janković

Mentor: Nikola Dimitrijević

Sadržaj

[1. Predlog teme 3](#_Toc491866984)

[2. Opis funkcionalnosti 3](#_Toc491866985)

[3. Struktura aplikacije 4](#_Toc491866986)

[4. Korisničko uputstvo 5](#_Toc491866987)

[5. Zaključak 5](#_Toc491866988)

[6. Literatura 6](#_Toc491866989)

# Predlog teme

Tema koja se obradjuje je program za apoteku.Apoteka poseduje naziv I liste klijenta, porudžbina I stavki apoteke. Ono što je karakteristično je to da je moguće korišćenje I iz ugla farmaceuta koji je zaposlen u apoteci i iz ugla klijenta koji kupuje lekove.

Program poseduje apstraktnu klasu Osoba koju nasledjuju klase Farmaceut koji kao dodatni atribut poseduje pristupni PIN kako bi se ulogovao. Klijenti imaju atribut broj zdravstvene knjižice po kojem se kasnije i pretražuju iz fajla,koji takodje ima ograničene vrednosti.

Svaki lek ima naziv,proizvodjača,jačinu,tip leka koji je nabroji tip (Enum) i može biti tablete,sirup,prašak ili injekcije , cenu kao i atribut tipa boolean da li je lek sa receptom ili bez.

Stavka apoteke i stavka porudžbine poseduju kao atribute lek i kolicinu,koji kasnije koriste metode u pretraživanju i računanju.

# Opis funkcionalnosti

Program nudi klijentima interaktivnu kupovinu lekova odnosno unos naziva kao I željene količine zatim I nastavak ako korisnik želi da kupi još nešto I prikazuje da li je kupovina izvodljiva, I naravno da li kupac želi da nastavi sa kupovinom I odabere neki drugi proizvod odnosno lek.

Odgovorom “ne” se proverava da li je korisnik već u našoj bazi klijenta,ako nije u listi klijenata od njega se traži unos njegovih podataka kako bi se registrovao kao klijent uz to da se mora poštovati dužina broja zdravstvene knjižice .

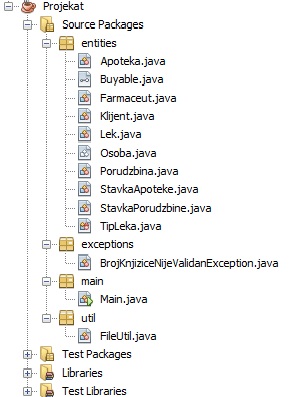
Sve opcije za farmacuta I kljenta se nude pomoću switch-a I metoda koje su vezane za svaki case u switch-u I na kraju se sve to ponovo upisuje u zasebne fail-ove sa svim izmenama koje su se desile I toku rada aplikacije za naredno pokretanje aplikacije.

# 

# Struktura aplikacije

Aplikacija se sastoji iz jednog enuma odnosno nabrojivog tipa u kome su navedeni tipovi lekova I jenog interfejsa koji se implementira u klasi apoteka kao I interfejsa Comparable koji uporedjuje stavke apoteke po nazivu I tako ih sortira..

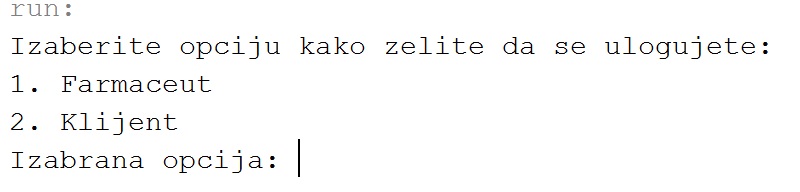
POJO[[1]](#footnote-1)-klase se nalaze u paketu entities I sve one su menjusobno povezane odgovarajućim relacijama-asocijacijama,agregacijom,kompozicijom I generalizacijom[[2]](#footnote-2).



# Korisničko uputstvo

Farmaceut se jedino može ulogovati unosom odgovarajućeg pristupnog PIN-a koji je upisan u fail ali se čita obrnuto pa je tako PIN-1234. Ukoliko dodje do greške postoji mogućnost ponovnog unosa sve dok on ne bude tačan.Kada se farmaceu uloguje on moze da izabere neke od ponudjenjih opcija kao što su da vidi ste stavke u apoteci,sve klijente o kojima apoteka čuva podatke,kao I da unese podatke o nekom novom korisniku, listu porudzbina I na samom kraju da izadje iz programa..

Klijent ima uvid u listu lekova iz koje bira ono što želi da kupi po nazivu pa zatim I količinu tog izabranog proizvoda, Korisnik zatim može da nastavi kupovinu unosom “Da” kada ce mu ponovo biti prikazana lista lekova ili unosom “Ne” ako to ne želi biće mu prikazan pregled kupovine kao I ukupna cena porudžbine.



# Zaključak

Navedeni program omogućava korićenje sa dve strane I u odnosu na njih prikazuje I nudi odgovarajuće metode.Projekat je samo početna vizija jednog programa za realnu apoteku koji u ovoj fazi ima dosta nedostataka.Neka od poboljšanja koja mogu da znacajno unaprede rad su da se od korisnika zahteva I unos odredjenog recepta sa nekim ID atributom koji se mora poklopiti sa atributom u bazi za navedeni lek.Zatim je moguće dodati posebnu klasu kao uputstvo za upotrebu leka ili možda čak metodu koja će farmaceutu omogućiti unos novih lekova I dodavanje odredjene količine leka.

Smatram da ovaj projekat ima dosta mogućnosti za dalji razvoj ali sa trenutnim znanjem on izgleda ovako.Nadam se da će koristiti kao dobra osnova za projekat na narednom predmetu.

# Literatura

**LITERATURA:**

**1.**LAMS-Univerziteta Metropolitan

**2.**Introduction to Java Programming - Y. Daniel Liang

**3.**Effective Java – “Efikasno programiranje u Javi”- Joshua Bloch

**4.**Java 8 Programiranje – Yakov Fain

1. POJO-Plain old Java Object [↑](#footnote-ref-1)
2. U prilogu maila koji se šalje se nalazi detaljan klasni dijagram sa graficki prikazanim relacijama [↑](#footnote-ref-2)