**Univerzitet u Novom Sadu**

**Tehnički fakultet „Mihajlo Pupin“**

**Zrenjanin**

SEMINARSKI RAD

**Sistem za raspolaganje robom u veterinarskij apoteci**

Razvoj softvera otvorenog koda

Profesor: Doc. dr. Kazi Zoltan Student: Eduardo Bojer SI 6/17

Zrenjanin, 2022. godina

Sadrzaj:

[1. Uvod 3](#_Toc112252441)

[2. Specifikacija zahteva korisnika 4](#_Toc112252442)

[3. Faze razvoja softvera 5](#_Toc112252443)

[3.1. Tabelarni prikaz slojeva i podslojeva aplikacije 5](#_Toc112252444)

[4. Prikaz softvera 6](#_Toc112252445)

[4.1. Pokretanje softvera i ekranski prikaz 6](#_Toc112252446)

[5. Opis implementacije 9](#_Toc112252447)

[5.1. Opis primenjenih tehnologija 9](#_Toc112252448)

[5.1.1. Microsoft Visual Studio Code 9](#_Toc112252449)

[5.1.2 PHP 9](#_Toc112252450)

[5.1.3 SQLite 10](#_Toc112252451)

[5.2 GitHub 10](#_Toc112252452)

[5.2.1 Kreiranje naloga 11](#_Toc112252453)

[6. Baza podataka 15](#_Toc112252454)

[6.1 Konekcija ka bazi 15](#_Toc112252455)

[7. Zaključak 20](#_Toc112252456)

[8. Literatura 21](#_Toc112252457)

# Uvod

Aplikacija omogućava korisniku programa da može da vodi listu svih proizvoda koje ima u apoteci ili radnji. Mogućnosti: držanja skladišta, kolčina proizvoda, da izbaci proizvode ili da im modifikuje podatke i da ima redni broj odredjenog proizvoda gde taj redni broj bi sluzio kao sifru za kasu. Program omogućava pravljenje naloga za radnike i ulogovanje malopre pomenutih radnika. Nazive tabela za program su items(proizvodi) i users(korisnici).

# Specifikacija zahteva korisnika

Aplikacija obuhvata:

▪ prilikom otvaranja programa korisniku se daje opcija da se uloguje na program ili da pravi sebi nalog da bi mogo da pristupa programu;

▪ ukoliko korisnik treba da pravi nalog zahteva se od njega da ubaci njegov radni kod ime i sifru koju zeli da ima;

▪ ukoliko korisnik ima vec nalog samo na login prozoru unese svoje ime i sifru i onda mu se otvara glavni prozor gde moze da vidi sve proizvode i dugmeta za rad sa podacima i polja za ubacivanje novog proizvoda ili za pretragu;

▪ dugmeti za korisnike su dodavanje, brisanje, azuriranje i ciscenje i ima polje za pretragu i polje gde moze da vidi podatke iz baze;

# Faze razvoja softvera

-specifikacija zahteva korisnika

-dizajn baze podataka

-kreiranje baze podataka

-dizajn korisničkog interfejsa (plan/opis)

-implementacija izgleda

-programiranje logike

-pokretanje softvera

-kreiranje dokumentacije

## 3.1. Tabelarni prikaz slojeva i podslojeva aplikacije

|  |  |  |
| --- | --- | --- |
| SLOJ | PODSLOJ | SEMINARSKI RAD |
| PREZENTACIONI SLOJ | Evidencija proizvoda | Main.py |
|  | Unos Korisnika | Singup.py |
| unosProizvoda | Main.py |
| izmenaProizvoda | Main.py |
| APLIKACIONI SLOJ | methoderada | methoderada |
|  | methoderada | methoderada |
| SLOJ BAZE PODATAKA | Ims.db | Ims.db |

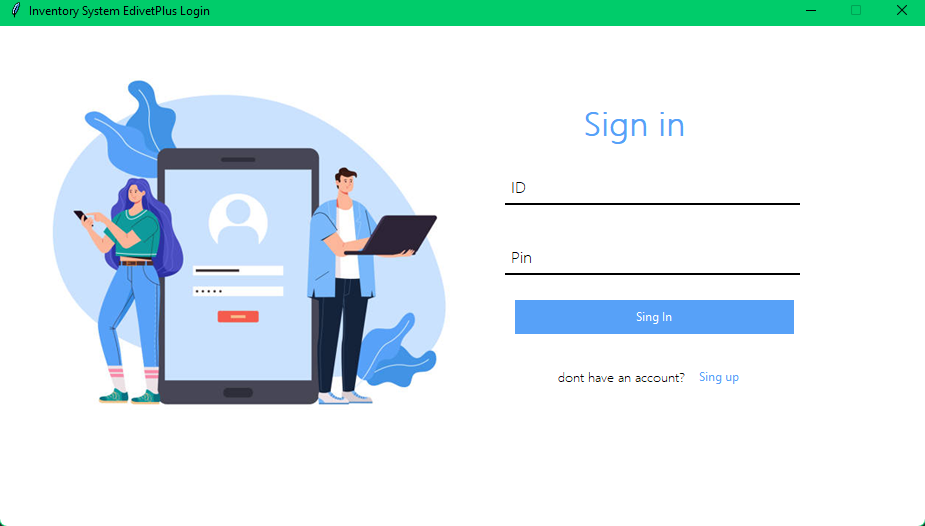
*Tabela 1- Prikaz slojeva i podslojeva aplikacije*

# Prikaz softvera

## Pokretanje softvera i ekranski prikaz

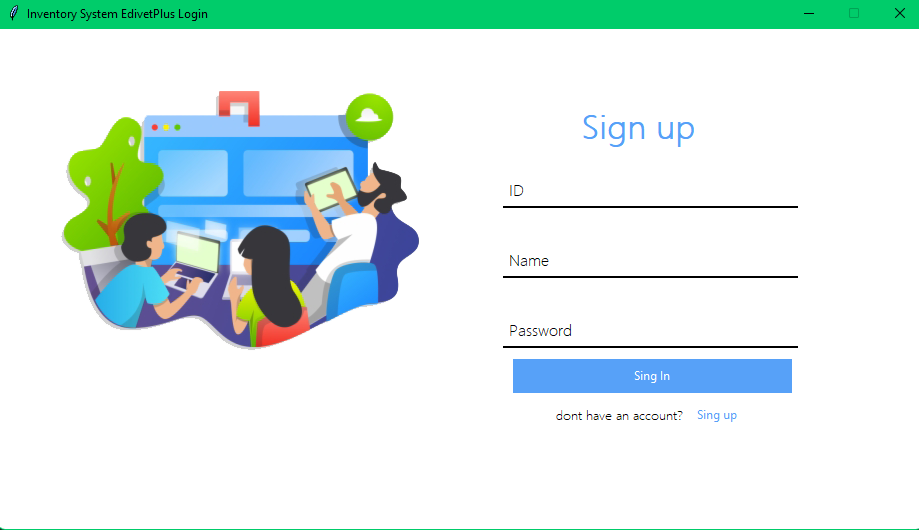
Za prikaz ove aplikacije koristi se python runner, neophodan za njeno pokretanje.

Nakon uspešnog pokretanja aplikacije dobijamo prikaz login prozora. Na ekranu se nalazi polja za unos kredencijala za korisnika.



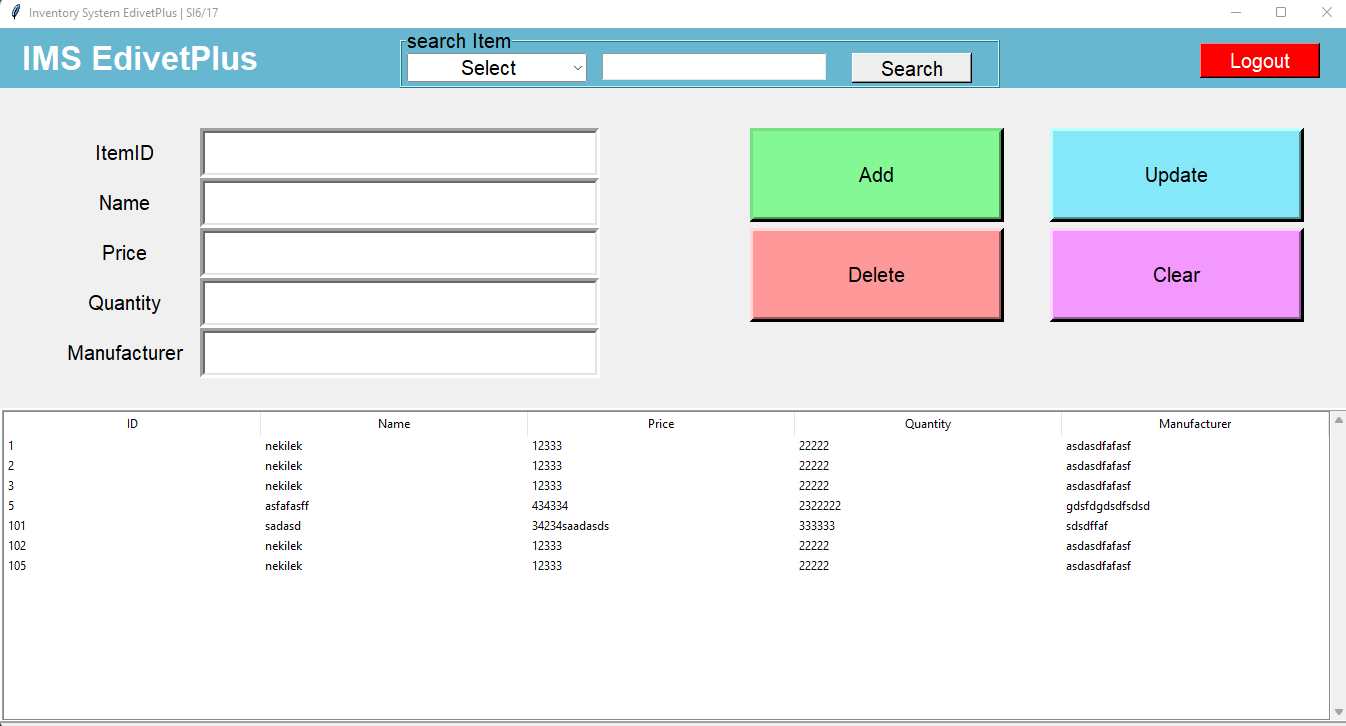
*Slika 1- Snimak ekrana prvog prozora aplikacije*

Nakon klika na dugme za registraciju (Sign up) otvara se prozor za registraciju korisnika.



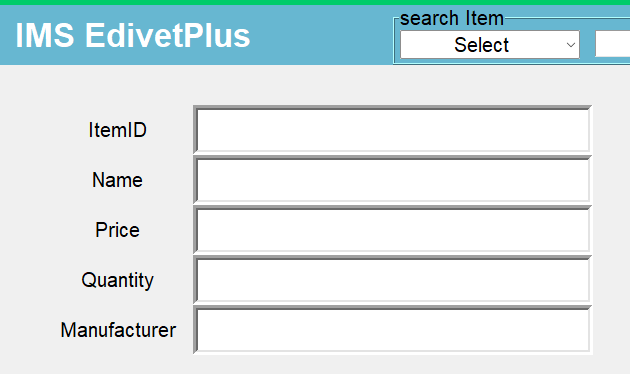
*Slika 2- Prikaz prozora za pravljenje naloga*

Ovaj prozor pokaze kako izgleda kosnicki prozor gde moze da krene da radi sa podacima.



*Slika 3- Prikaz prozora nakon pregistracije*

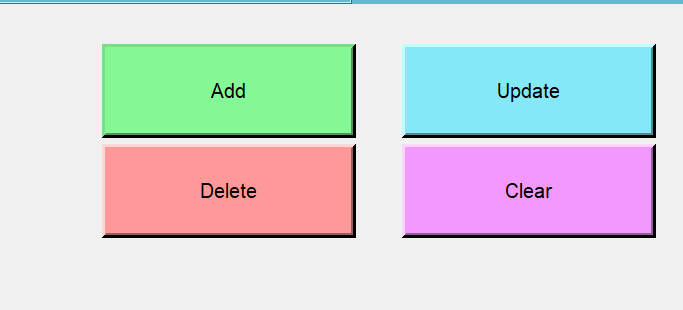
U ovom prozoru korisnik moze da unese podatke za nove prizvode i da unese novi proizvod u bazu.



*Slika 4 – Prikaz ekrana unos proizvoda*

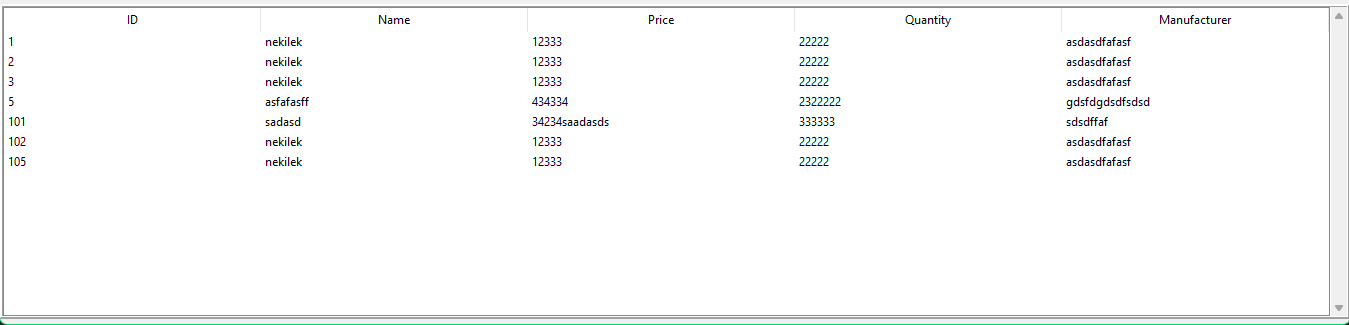
Ovaj prozor sadrži formu za unos podataka. Podaci koji se unose su ID proizvodas, Ime proizvoda, cenu proizvoda,kolicina proizvoda i proizvodac.

U ovoj slici korisnik ako zeli da ubaci nove podatke potrebno mu je da unese podatke u malopre pomenuto polja i onda stisne na dugme add a ako zeli da radi neku od druga dva napomenute opcije update ili delete onda mora da selectuie elemenat sa liste i da klikne onda na koju opciju zeli



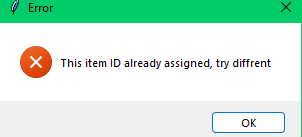
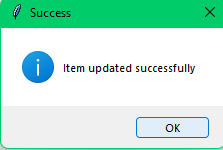
*Slika 5 – Prikaz dugmeta za rad sa podacima*

Ovo je polje gde moze korisnig da vidi sve prozvode koje su unesene u bazu i samo sa jednim klikom na red koji zeli da promeni moze samo sa klikom da oznaci to polje i u polja za unos podataka bi se pokazala podaci pa korisnik moze da radi na podatke .



*Slika 6- Polje za prikaz podataka*

Pravljenje su opcije da se pokaze prilikom gresaka unosa podataka da iskoce greske ako korisnik nije nesto uredu radi u programu.

*Slika 7– Izgled gresaka i uspesnih popapova*

# Opis implementacije

## Opis primenjenih tehnologija

### Microsoft Visual Studio Code

Visual Studio Code ( skraćeno VS Code) uređivač je izvornog koda koji je razvio Majkrosoft za Windows, Linuks i Mac OS.Podržava otklanjanje grešaka, ugrađenu git kontrolu i GitHub, isticanje dijelova sintakse, inteligentno dovršavanje koda, odlomke i refaktorisanje koda. Veoma je prilagodljiv, omogućava korisnicima da menjaju temu, prečice na tastaturi, podešavanja i instaliraju priključke za dodatne funkcionalnosti. Izvorni kod je besplatan i otvoren i objavljuje se pod popustljivom MIT licencom.Kompajlirani binarni fajlovi su besplatni i slobodni za privatnu ili komercijalnu upotrebu.

### 5.1.2 PHP

Pajton (engl. Python) je programski jezik visokog nivoa opšte namene. Podržava, u prvom redu imperativni, objektno-orijentisan i funkcionalni stil programiranja. Sintaksa jezika Pajton omogućava pisanje veoma preglednih programa. Jezik se brzo i lako uči.

Programi pisani u Pajton jeziku se najčešće interpretiraju. Uz interpretator se obično isporučuje i veoma razvijena standardna biblioteka modula.

Autor ovog jezika je Gvido van Rosum sa Univerziteta Stičing u Holandiji..

### 5.1.3 SQLite

SQLite je ACID-kompatibilan ugrađen sistem za upravljanje bazama podataka sadržan u relativno maloj (~225 KB) C programskoj biblioteci. Izvorni kod za SQLite je u javnom vlasništvu.

Za razliku od klijent-server sistema za upravljanje bazama podataka, jezgro SQLite-a nije samostalan proces sa kojim aplikacija komunicira. Umjesto toga, SQLite biblioteka je uvezana i postaje sastavni dio aplikacije. Biblioteka može biti dinamično pozivana. Aplikacija koristi SQLite funkcionalnost kroz proste funkcionalne pozive, koji smanjuju latenciju u pristupu bazama podataka kako su funkcionalni pozivi unutar jednog procesa efikasniji od međuprocesne komunikacije. Kompletna baza podataka (definisanje, table, indeksi i sami podaci) kao jedan međuplatformni fajl na mašini domaćinu. Ovako prost dizajn je postignut zaključavanjem kompletnog fajla baze podataka na početku transakcije.

SQLite je ugrađen u sve veći broj popularnih programa. Na primjer, Mozila fajerfoks pohranjuje mnoštvo konfiguracionih podataka, (bukmarkovi, kukiji, itd.), u interno upravljanu SQLite bazu podataka. Kao drugi primjer, Guglov Android operativni sistem za mobilne telefone i druge male uređaje sadrži SQLite.

## 5.2 GitHub

GitHub je veb-baziran hosting servis za kontrolu verzije, Git. Pruža Git funkcionalnosti: distribuiranu kontrolu revizija i menadžment izvornog koda (Source Control Management - SCM), dodajući dodatne funkcije.

Za razliku od Git-a, koji je striktno alat koji se koristi iz komandne linije, GitHub pruža veb grafički interfejs, radnu površinu i mobilnu integraciju. Takođe pruža kontrolu pristupa i nekoliko funkcija za saradnju, kao što su praćenje grešaka, zahteve za dodavanje novih karakteristika, upravljanje zadacima i mogućnost pravljenja viki dokumentacije za svaki projekat.GitHub pruža planove za privatna skladišta kao i besplatne naloge, koji se obično koriste kao hostovi za sofverske projekte otvorenog koda.

Projektima na GitHub-u se pristupa i manipuliše koristeći standardni Git interfejs iz komandne linije preko kojeg su dostupne i sve standardne Git komande. GitHub dopušta registrovanim i neregistrovanim korisnicima da pregledaju javna skladišta na sajtu.

GitHub sajt pruža funkcije slične društvenim mrežama kao što su: dovodi (engl. feeds), pratioci (engl. followers), viki (koristeći softver Gollum) i grafikone saradnje koji ukazuju kako programeri rade na svojim verzijama (&quot;račvama&quot;) skladišta i koja račva je najnovija.

Korisnik mora da napravi nalog kako bi doprineo stranici, ali javna skladišta mogu biti pregledana i preuzeta od strane bilo koga. Sa registrovanim korisničkim nalogom, korisnici mogu da diskutuju, upravljaju skladištima, prave nova skladišta, postavljaju doprinose drugim skladištima i pregledaju izmene u kodu.“

GitHub se najviše koristi za izvorne kodove. Osim izvornog koda, GitHub podržava sledeće formate i funkcije:

- Dokumentaciju, uključujući automatski generisane README fajlove u raznim Markdown fajl formatima (pogledati README fajlovi na GitHub-u),

- Praćenje problema (uključujući zahteve za novim karakteristikama) sa labelama, prekretnicama, zastupnicima i pretraživačem,

- Viki podrška,

- Zahteve za pregled izmena sa recenzijama koda i komentarima,

- Istorija izvršavanja,

- Grafikoni (puls, saradnici, frekvencije koda, mreže, članovi),

- Adresar integracija,

- E-mail notifikacije,

- Opcija prijave za notifikacije o nekome, koristeći znak @,

- Emoji podrška,

- Ugnježdene liste zadataka unutar fajlova,

-Vizuelizacija geoprostornih 3D podataka koji se mogu pregledati koristeći novointegrisani STL fajl pregledač koji otvara fajlove na 3D platnu,

-Photoshop-ov nativni PSD format se može pregledati i porediti sa prethodnim verzijama istog fajla. [7]

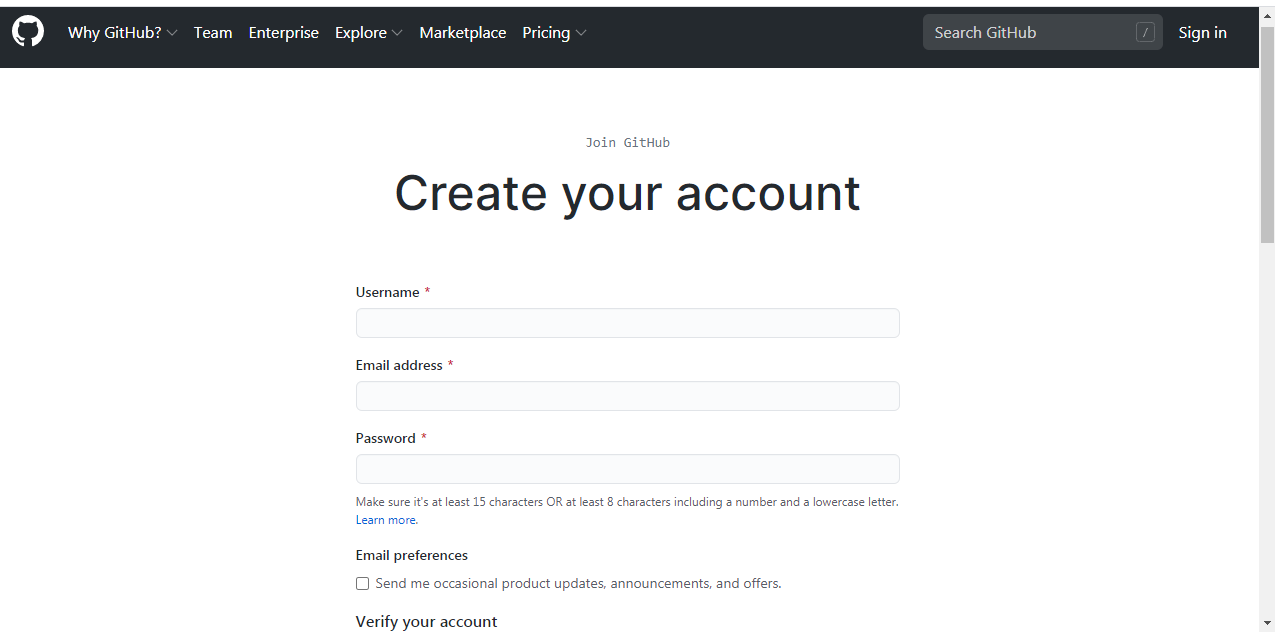
### 5.2.1 Kreiranje naloga

Nalog na GitHubu može se kreirati koristeći jedan od dva sledeća linka:  
▪<https://github.com/>

▪<https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home>

Prilikom kreiranja potrebno je uneti korisničko ime (eng. username), email adresu i lozinku (eng. password)

Na slici ispod je prikazana stranica na kojoj kreiramo nalog.



*Slika 14 – Kreiranje GitHub naloga*

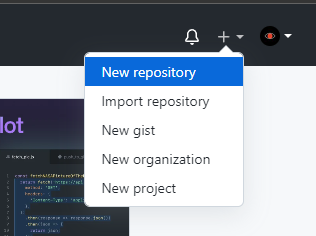
**5.2.2 Kreiranje repozitorijuma**

Svaki korisnik ima mogućnost kreiranja repozitorijuma. Novi repozitorijum se kreira klikom na znak plus koji se nalazi u gornjem desnom uglu.



*Slika 15-Prikaz lokacije znaka plus za kreiranje repozitorijuma*

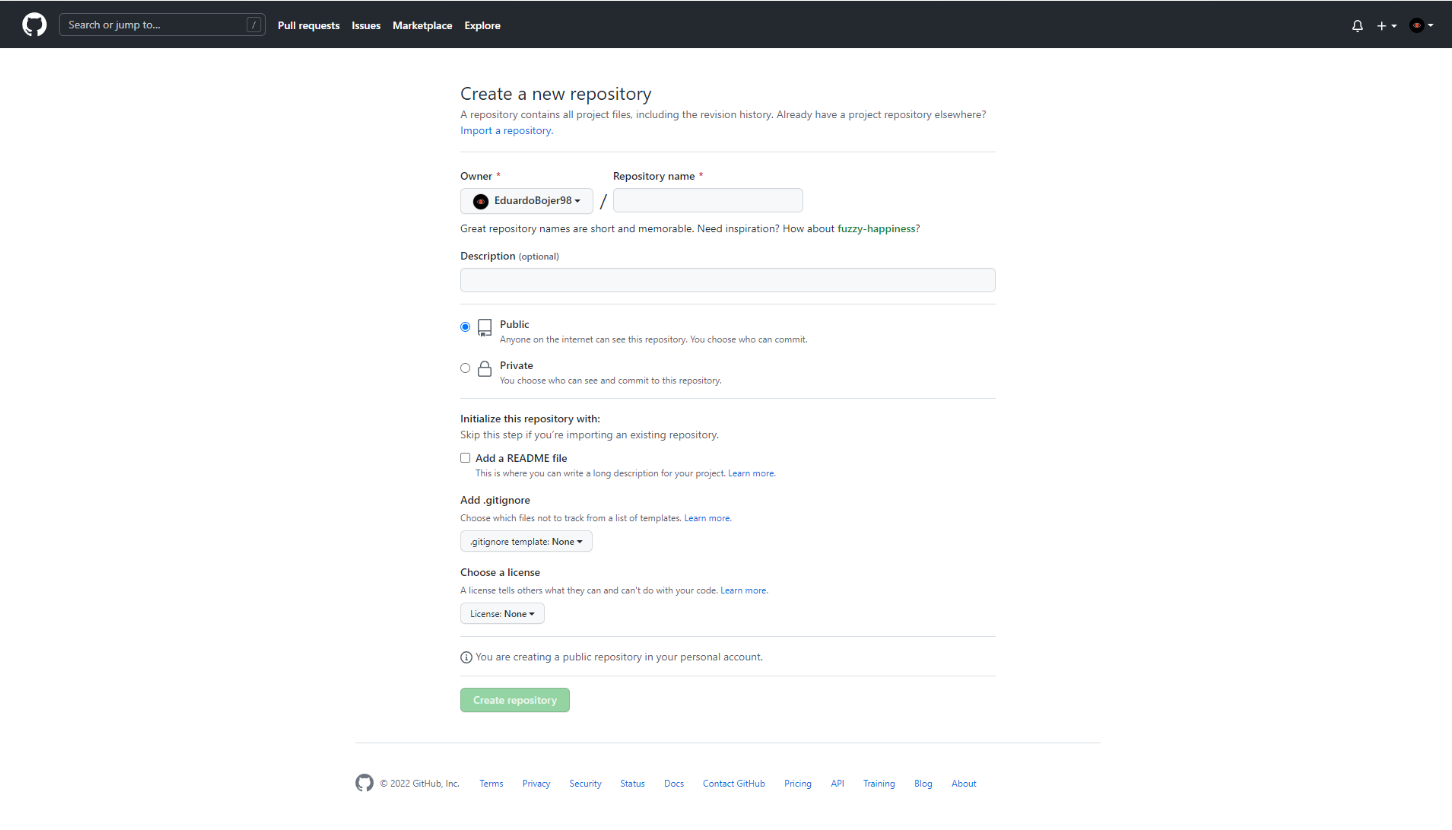
Zatim se u padajućem meniju izabere New repository.



*Slika 16– Izgled padajućeg menija*

Stranica za kreiranje repozitorijuma i podaci koje je potrebno uneti prilikom kreiranja istog prkazani su na sledećoj slici.

Konkretno u ovom seminarskog radu repozitorijum je nazvan Inventory System EdivetPlus.

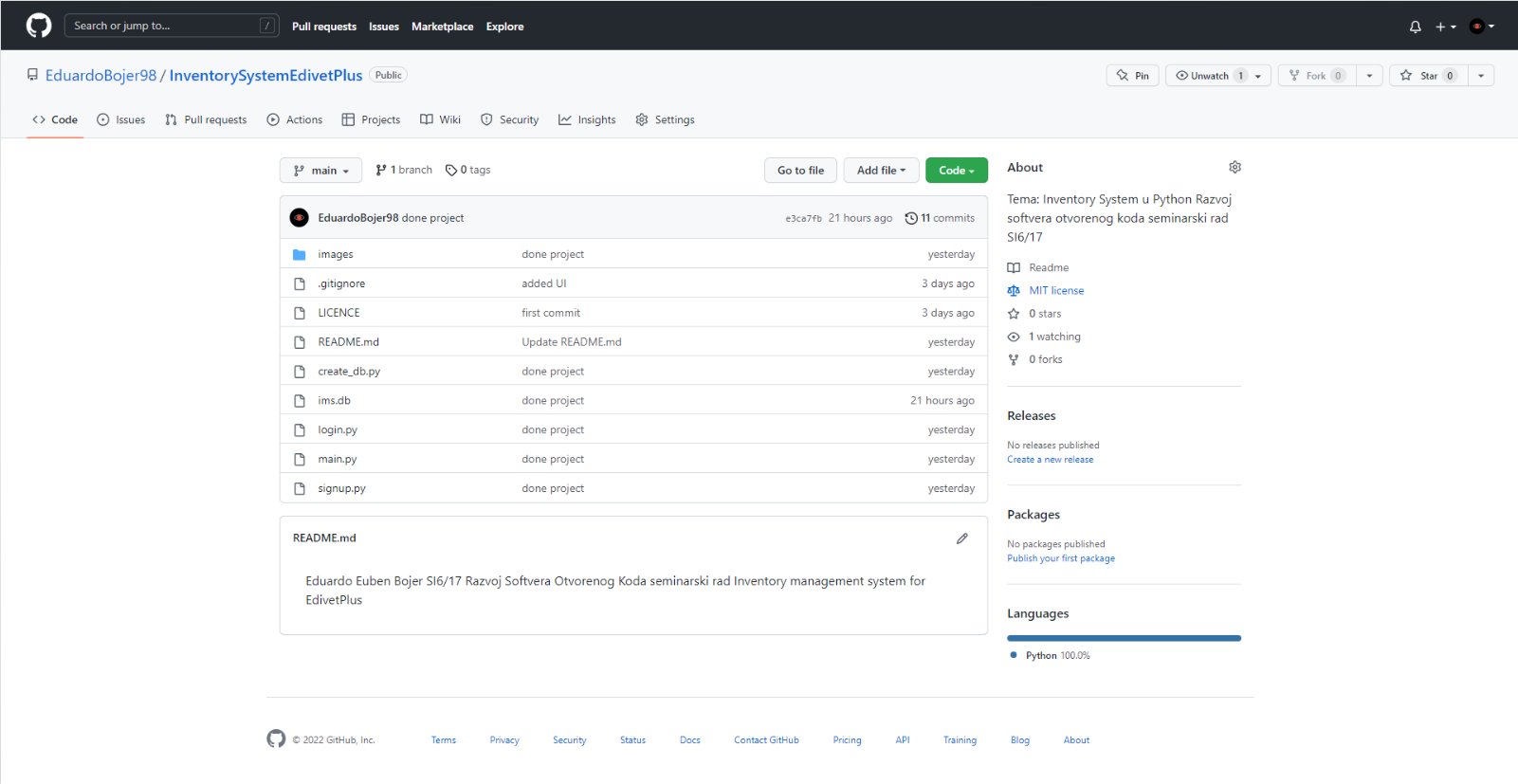


*Slika 17- Stranica na kojoj se kreira repozitorijum*

Repozitorijum može da sadrži sledeće fajlove:

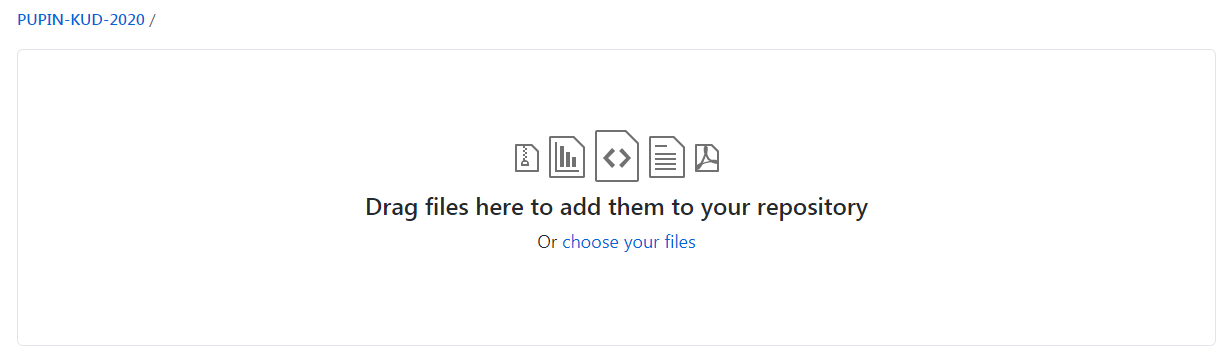
* README,
* LICENSE
* .gitignore.

Nakon kreiranja dobija se sledeći prikaz:



*Slika 18- Izgleda repozitorijuma azil*

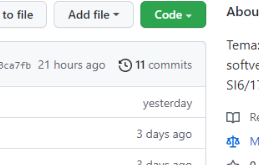
Na slici iznad takodje se mogu videti fajlovi koji su ubačeni u repozitorijum. Fajlovi se mogu dodati klikom na Add file. Add file dugme pruža sve opcija: Create new file i Upload files. Fajlove je moguće dodati biranjem sopstvenih iz foldera ili prevlačenjem.



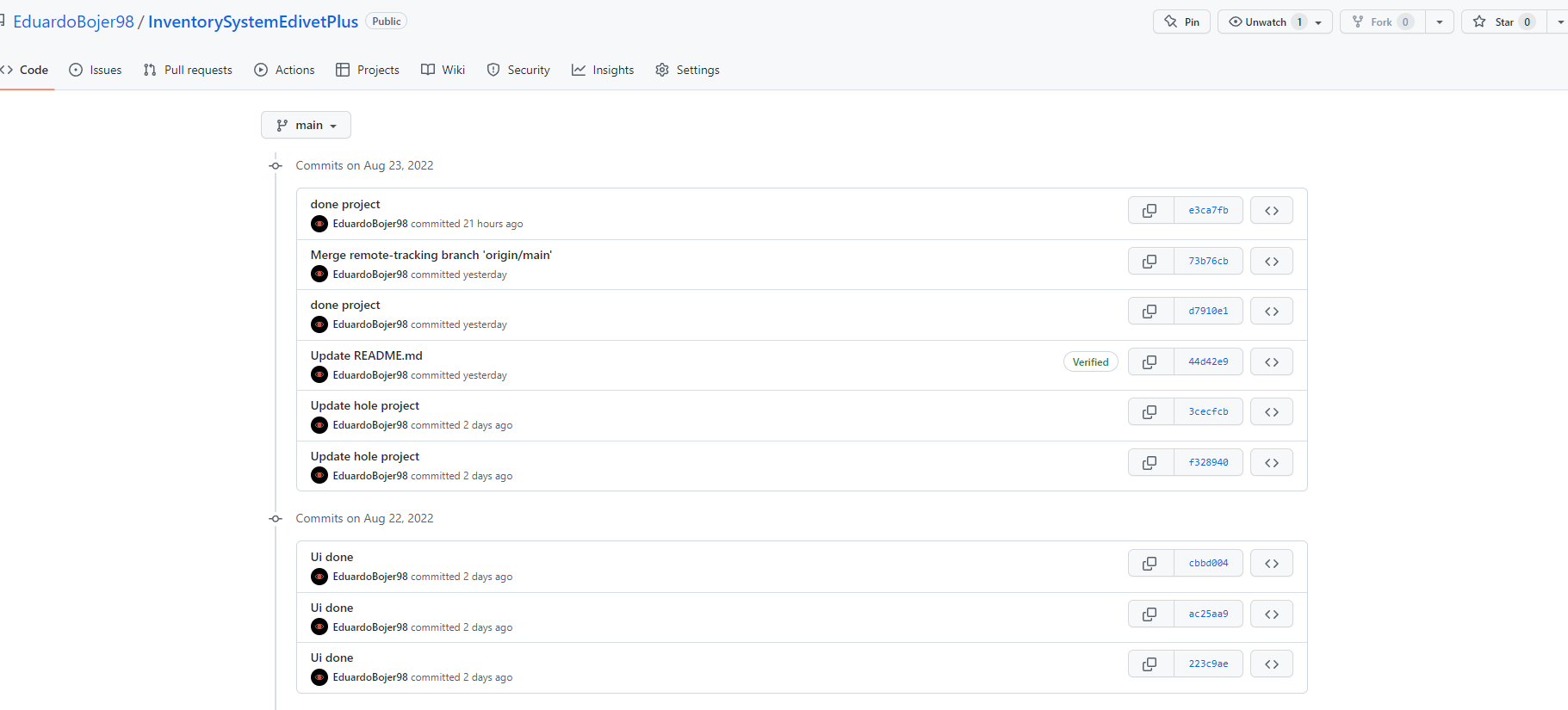
*Slika 19– Dodavanje fajlova*

Repozitoriju pruža uvid u akcije koje su preduzete od strane članova tima.

Potrebno je kliknuti na dugme koje je obeleženo na sledećoj slici.



*Slika 20– Lokacija Commits-a*



*Slika 21– Prikaz rada korisnika*

# Baza podataka

Za potrebe ovog seminarskog rada kreirana je baza podataka pod nazivom ims.db. Ova baza podataka kreirana je pomocu imputa sqlite3 dodatkom za python i komocu create\_db.py skripte se pravila baza i pogled na bazu se radi preko SQLite alatu.

Sastoji se od dve tabele:

* korisnik,
* proizvodi,

Nakon kreiranja tabela, u svaku tabelu dodati su atributi.

Tabela *korisnik* ima sledeće atribute:  
id, ime, sifra.

Tabela *proizvodi* ima sledeće atribute:  
id, ime, cena, kolicina,proizvodjac.

Svaka tabela mora da ima primarni ključ. Kao primarni ključ potrebno je odabrati jedan atribut. U ove dve tabele, primarni ključ je id (u oba slučaja).

Tabela *korisnik*:

CREATE TABLE `korisnik` (

`itemID` int(255) NOT NULL,

`Name` varchar(50) DEFAULT NULL,

`password` varchar(50) DEFAULT NULL

)

Tabela *items*:

CREATE TABLE `items` (

` itemID ` int(255) NOT NULL,

` Name ` varchar(50) DEFAULT NULL,

` Price ` int(255) DEFAULT NULL,

` Quantity ` int(255) DEFAULT NULL,

` Manufacturer ` varchar(10) DEFAULT NULL,

)

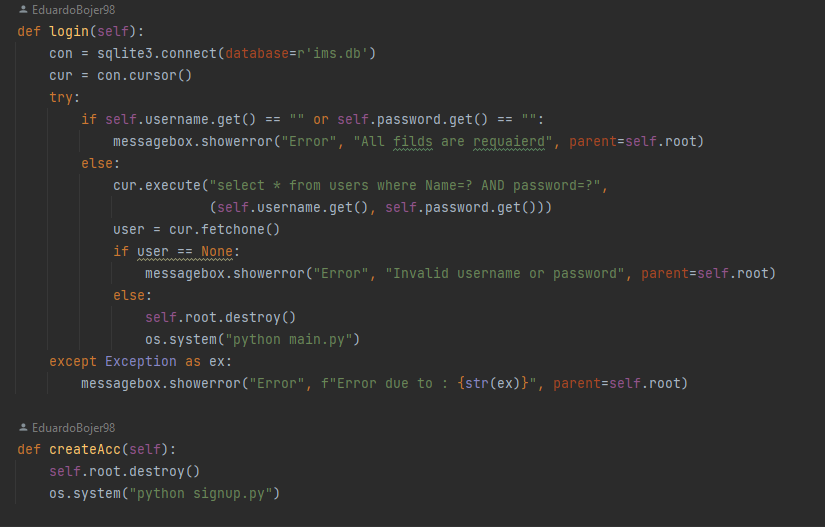
## 6.1 Konekcija ka bazi

Kod za konekciju sa bazom

|  |
| --- |
|  |

*Listing 1- Kod za konekciju sa bazom*

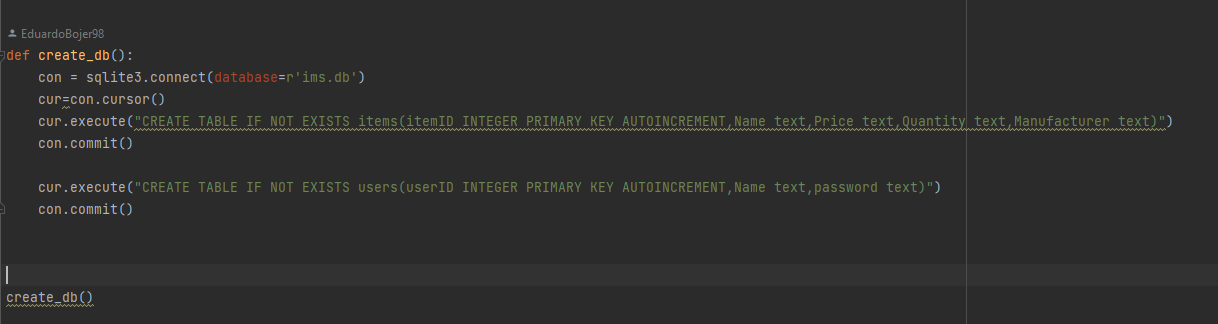
Program sadrzi u sebe kodove metode za logovanje za kreiranje naloga za dodavanje proizvoda za azuriranje brisanje i potraznju.



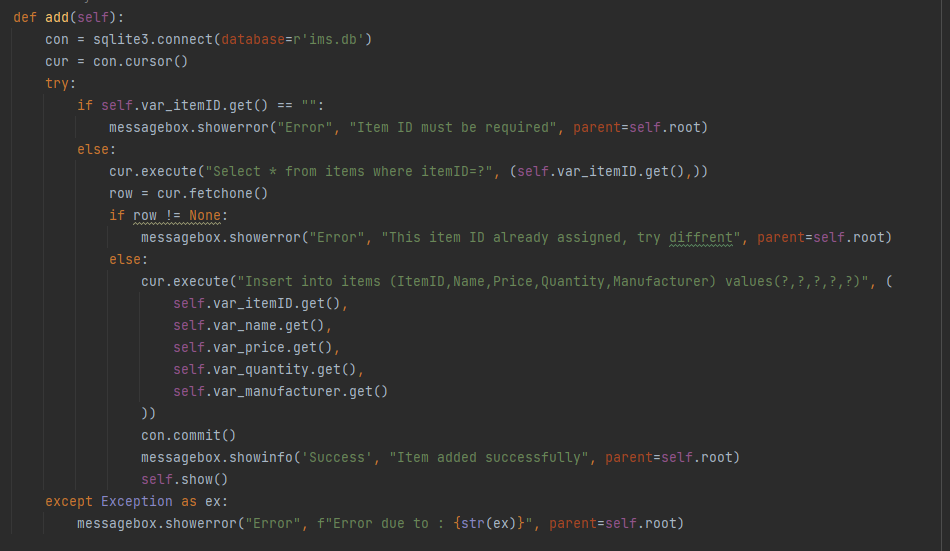
*Listing 2- Kod za logovanje i prebacivanje na prozor za singup*



*Listing 3- Kod dodavanje korisnika*



*Listing 4- Kod kreiranje baze*



*Listing 5- kod za dovanje proizvoda*



*Listing 6- Kod pokazivanje i pozivanje podataka*



*Listing 7- Kod azuriranje podatke iz baze*



*Listing 8- Kod za obrisanje podataka i za ciscenje prozora za unos*



*Listing 9- Kod pretragu podataka po kolokni i metoda za prebacivanje na login stranicu*

# Zaključak

Ovaj seminarski rad upoznaje nas sa GitHub-om koji ljudima omogućava da uče, dele svoje znanje i saradjuju kako bi izgradili novi softver. Pomoću aplikacije IMS EdivetPlus savladano je korišćenje i snalaženje na GitHubu.

# Literatura

[1]: <http://www.tfzr.uns.ac.rs/Content/files/0/GITHUB%20uputstvo%20FINALNO.pdf>

[2]: <http://www.tfzr.uns.ac.rs/Content/files/0/RSOKVezbe%2020192020%20-%20dokument%20FINALNO.pdf>

[3]: <https://sr.wikipedia.org/sr-el/GitHub>