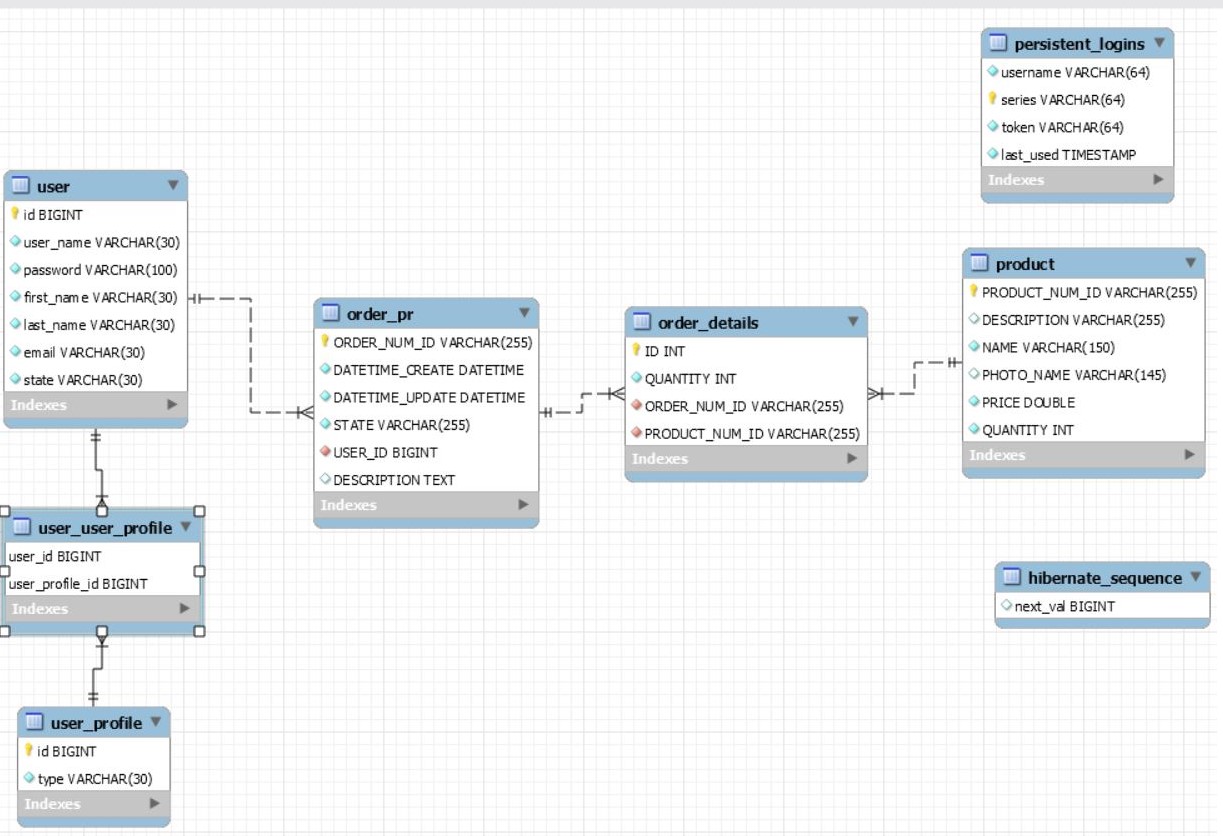
Verzija 1.0

Aplikacija je zamišljena kao neka vrsta *Web Shopa*. U prvoj verziji su kupci poznati. Mogu biti dodavani samo od strane administratora. Tako da je cela aplikacija zatvorenog tipa, neka vrsta glavnog magacina I kupaca koji su njihovi poznati distributeri (prodavnice). ***Home I About us*** su stranice koje su reklamne I dostupne svim posetiocima. Za sve ostale stranice je neophodno ulogovati se, ovlasćenja I pristup su u skladu sa ulogama koje dobijaju korisnici od strane administratora. Aplikacija je kreirana za opstu upotrebu ( ne specializovanu deletnost), pa u cilju slikovitije upotrebe koristim slike, logo, naziv sopstvene firme.

1.0 BAZA

***Baza MySQL 8.0.19***

****

*Slika šeme baze*

U prilogu u folderu mySql dostavljam sve neohodne sql upite za kriranje baze podataka.

Tabela ***user*** čuva sve korisnike sa njihovim osnovnim podacima. Kolana ***password***čuva šivre korisnika koje se čuvaju enkriptovane*.* Povezana je sa tabelom ***user\_profile*** vezom više prema više jer je zamisao da jedan korisnik može da ima više uloga.

Tabela ***user\_profile*** tako sadrži:

1. ADMIN – Uloga administratora.
2. USER – Za sada bez specijalne uloge. Ostavljen za mogućnost brzog prosirivanja.
3. EMPMENAGER – Menadžer zaposlenih.
4. WARMENAGER – Menadžer magacina.(Prodavac).
5. SHOPMENAGER – Mendžer prodavnice. (Kupac).

Prvi korisnik je administrator. Dodaje se upitom. A šifra se generše pomoću klase***QuickPasswordEncodingGenerato****r* aplikacije koja se nalazi u ***generators*** folderu aplikacije. Za kriptovanje koristimo *BCryptPasswordEncoder*. Šifra je 123 a nick name admin.

*INSERT INTO`user` VALUES (1,'admin','$2a$10$94FgHW3muCFTDS4ipVhnm.Z1W1/v2m1PUyYT8LhLW0BJpF.3pbt7K','Nemanja','Matovic','neca@gmai.com','Active');*

Tabela ***hibernate\_sequence*** čuva sledeću vrednost za kreiranje jedinstvenih ID vrednosti u tabelama ***order\_pr I product***.

Tabela ***order\_pr*** čuva sve podatke o kreiranim porudžbinama. Datum kreiranja I izmene porudžbine se automatski kreiraju korišćenjem ***TIMESTAMP*** funkcije MySql-a.

Tabela ***order\_details*** čuva detalje svih porudžbina.

Tabela ***product*** čuva informacije o svim proizvodima.

Tabela ***persistent\_logins*** čuva podatke o kolačicu koji pamti koisnika kada je ukljucena funkcija ***remebe me*** u aplikaci. Čuva ga oniliko dugo koliko je odredjeno aplikacijom. U ovom slucaju sat vremena.

2.0. APLIKACIJA

Za izradu aplikacije je korišćen ***Spring MVC 4.1.6.RELEASE,*** za zaštitu i autentifikaciju ***Spring Security 4.0.1.RELEASE,*** a za integraciju sa ***MySql*** bazom podataka ***Hibernate 4.3.0.Final. CRUD*** operacije se vrše u okviru transakcija. Čuvanje lozinka se obavlja uz pomoć ***Bcrypt-a,*** a funkcionalnost pamćnja korisnika uz pomoć prilagodjene implementacije ***PersistentTokenRepository*** sa ***Hibernate*** ***HibernateTokenRepositoryImpl.*** Za generisanjenje jedinstvenih primarnih ključeva korisi se **SequenceStyleGenerator.** Za prikaz koristimo ***jsp pages, Bootstrap 4.6., Jquery-3.5.1., JavaScrip, HTML, CSS.***

***Pre pokretanja aplikacije neophodno je podesiti application.properties file, koji se nalazi u resource folderu aplikacije radi konekcije ka bazi. Kao I URL putanju do static/photo foldera na kome se cuvaju slike UPLOAD\_LOCATION u kontroleru ProductControlle.***

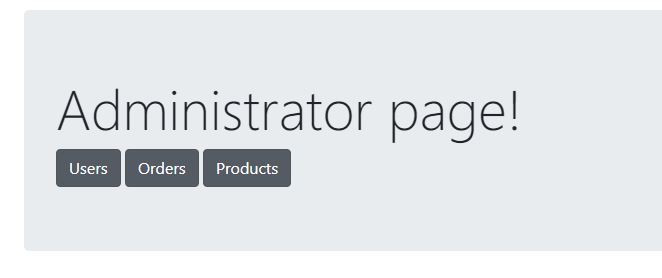
2.1. LOGIN/LOGOUT

Dugme ***Login*** je vidljivo samo na ***Home I About us*** strani. Vodi na stranicu za logovanje. U slučaju da je korisnik već ulogovan klikom na njega se pelazi na početnu stranicu ulogovanog korisnika, ona zavisi od uloge koja je dodeljenja korisniku. Za to je zaslućna kreirana klasa ***TargetUrl*** I njena metoda ***determineTargetUrl*** koja izvlači uloge trenutno prijavljenog korisnika iz objekta za potvrdu indetiteta I konstruiše odgovarajuće URL na osnovu uloga. A klasa ***CustomSuccessHandler*** nasledjuje ***SimpleUrlAuthenticationSuccessHandler*** I gazi njenu metodu ***handle*** u kojoj postavlja podrazumevanu redirect strategiju na dobijeni URL.

Na svim ostalim stranama dugme login ptrelazi u pozdravnu poruku korisniku I dugme Logout koje odjavljuje korisnika I vraća ga na stranu za prijavljivanje uz poruku da je korsnik uspešno odjavljen. Metoda ***logout*** koju obeybedjuje klasa ***PersistentTokenBasedRememberMeServices Spring Sec Security*** nam omogućava: Onemogućava HTTP sesiju, zatim odveyuje sve objekte povezane sa njom. Uklanja potvrdu indetiteta. Brise vrednost konteksta iz trenutne niti. Briše zapamćeni kolčić u bazi.

Korisničko ime ulogovanog korisnika se dobija funkcijom ***getCurrentUser*** , koja se nalazi u klasama kontrolera. Ona vraća User name iz ***Spring SecurityContext-a.***

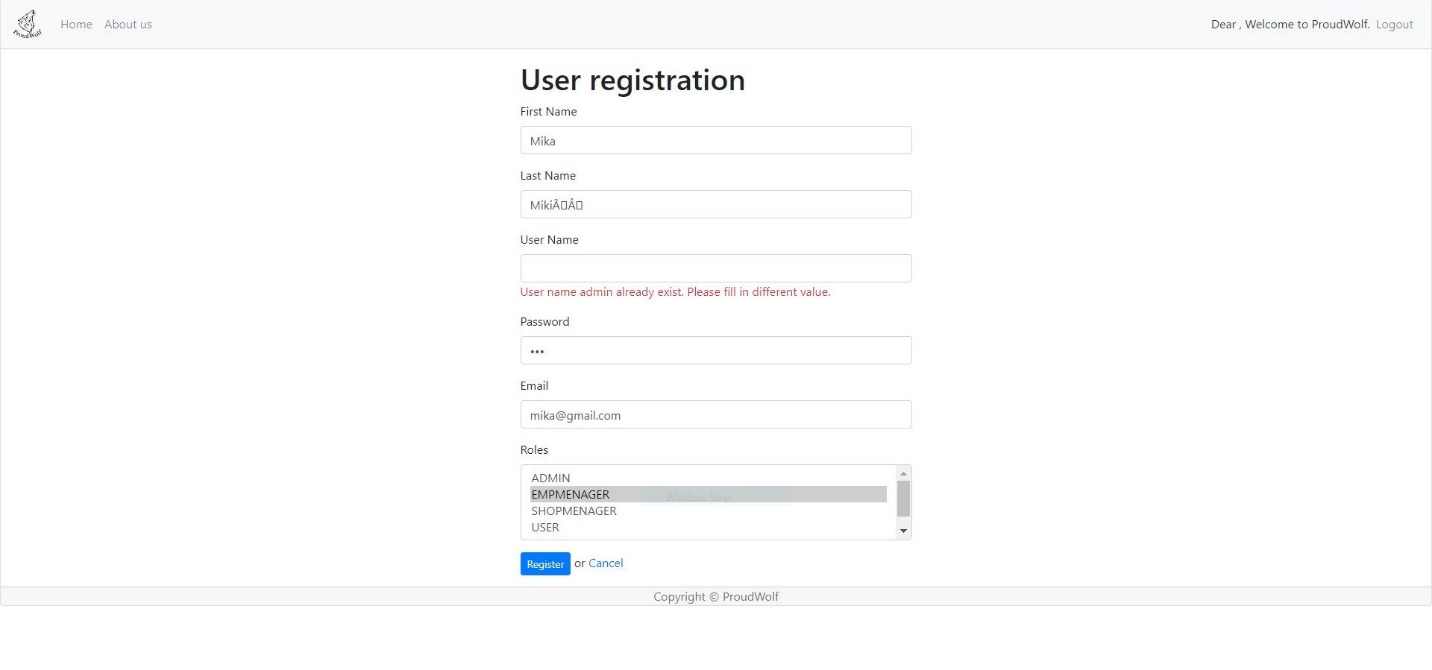
Dugme ***Remember me*** omugaćava da korisnik bude upamćen narednih sat vremena.

2.2. ADMIN

Administratorska strana

U primeru Adminstrator ima sve dozvole I vidi sav sadržaj. Pa će tako većina funkcija bititi objašnjena kroz druge korisnike. Dugme ***Users*** vodi do liste svih korisnika, ***Orders*** do svih porudžbina, ***Product*** do svih proizvoda.

Jedinstvenost administratora je jedino u tome sto može da dodaje, briše i izmenjuje korisnike.



Za kreiranje forme koristimo form tag iz springframework biblioteke tagova, a validaciju na formi omugućavamo uz pomoć anotacija koje dodajemo na polja klase modela. Koristimo ***hibernate-validator 5.1.3.Final*** i ***javax-validation 1.1.0.Final.*** Konfiguracijom ***MessageSource*** ispisujemo željene poruke u slučaju da valadicajia nije odgovarajuća. Jedinstvenost korisničnog imena se postize upotrebom metode***isUserNameUnique*** klase ***UserServiceImpl.***

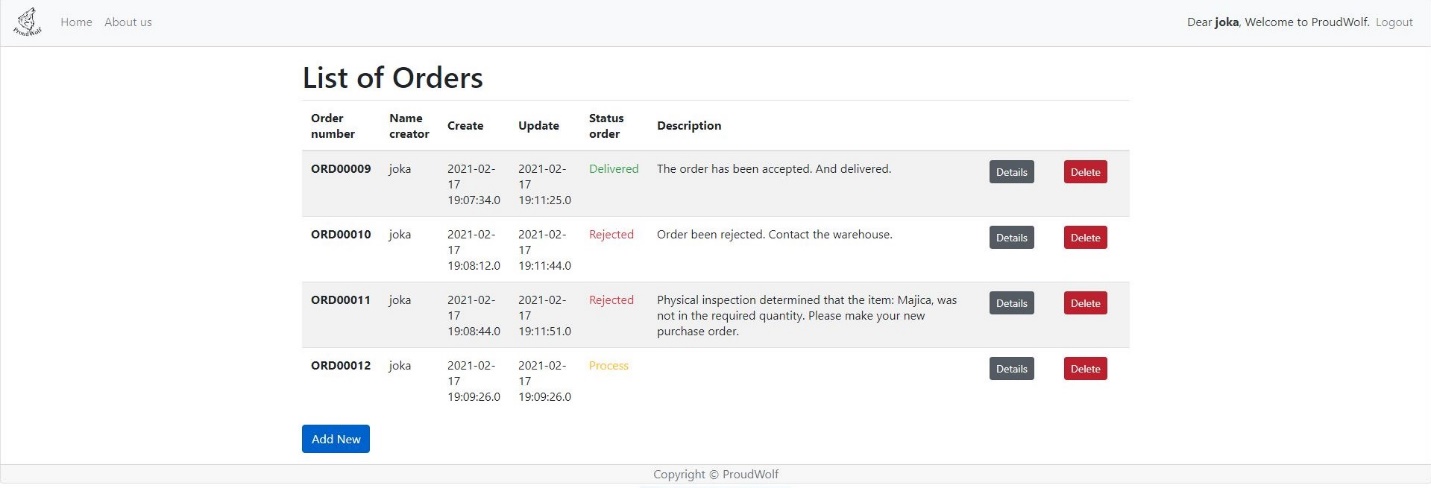
Korisnik se ne može izbrisati ako se njegovi podaci koriste u nekoj od porudzbenica.

2.3. EMPMENAGER

*lista korisnika*

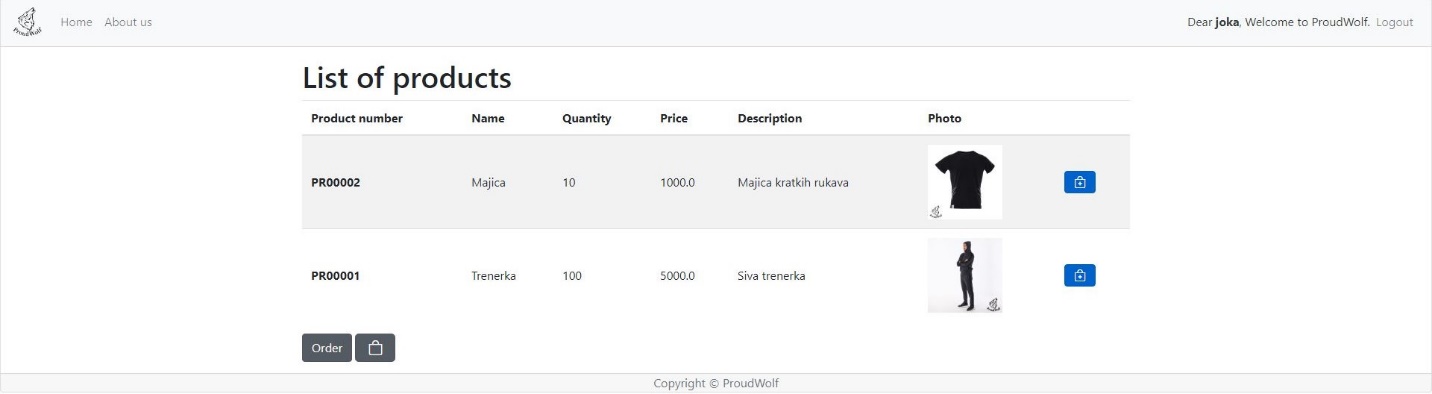
Menadžer zaposlinih ima uvid u sve korisnike (zaposlene) i uvid u sve porudžbine. Klikom na ***User state*** menja sanje korisniku i time mu daje ili oduzima pristup aplikaciji. Administraturu je nemoguće promeniti status.

2.4. SHOPMENAGER

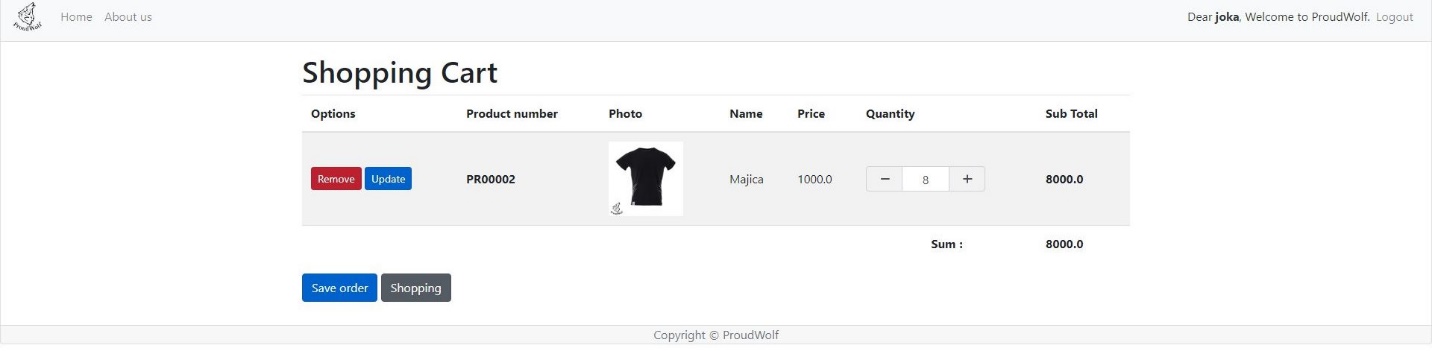


*Lista porudžbina*

Početna stranica, kada se uloguje, menedžera prodavnice (kupca) je lista svih njegovih porudžbina. Može da ih obrise jedino ako su u statusu obrade. Klikom na dugme ***Details*** vidi detalje porudžbenice, a klikom ***Add New*** odlazi na listu svih proizvoda kako bi kreirao novu.

*Lista svih proizvoda*

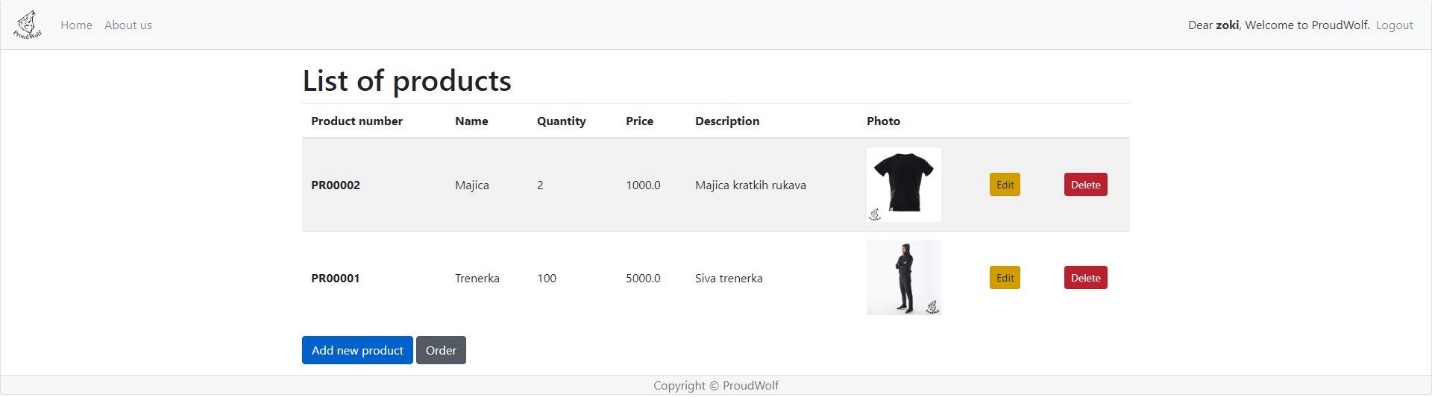
Klikom na plavu korpicu se dodaje proizvod u korpu, a na sivu uvid u korpu ako ima proizvoda u korpi. Ako stavka već postoji u korpi neće biti dodata ista već ze se samo pružiti uvid u korpu. To omugućava metoda ***isExisting*** koja se nalazi u kontroleru ***OrderController.***



*Korpa*

Uvid u korpu i do korpe može doći samo kupac. Lista proizvoda u korpi se kreira na nivo sesije. Aplikacija pamti sve proizvode u korpi dok se ne kreira porudžbina ili prekine sesija. Količina proizvoda ne može biti 0 ili veća od trenutnog stanja u magacinu. Kolona Sub Total računa verdnost zeljene kolićine artikla. Polje ***Suma*** racuna vrednost porudžbine. Klikom na taster ***Update*** se ubdejtuje zeljena količina, ***Remove*** uklanja stavka. Taster ***Shopping*** vraća kupca na listu svih proizvoda kako bih dodao novu stavku. Taster ***Save order*** snima porudžbinu i briše sve stavke iz korpe.

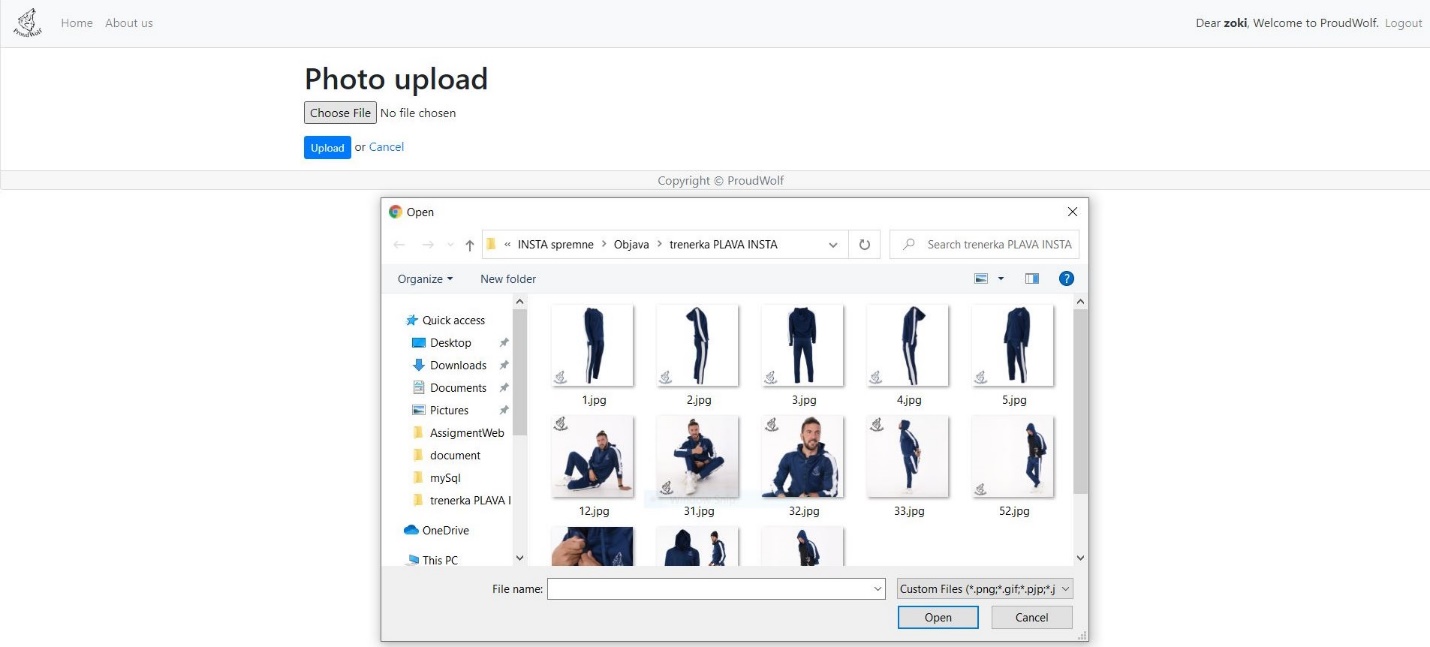
2.5. WARMENAGER



*Lista proizvoda(magacin)*

Početna strana posle logovanja menedžara magacina je lista svih proizvoda. Može da briše proizvode samo čiji se podaci ne nalaze u porudžbenicama. A izmena I dodavanje novih se vrši kreiranjem forme, kao kod korisnika. Koristimo form tag iz springframework biblioteke tagova, a validaciju na formi omugućavamo uz pomoć anotacija koje dodajemo na polja klase modela.

Kada je proizvod dodat u bazu otvara se novi prozor sa tasterom koji nas voidi na formu za upload slike.

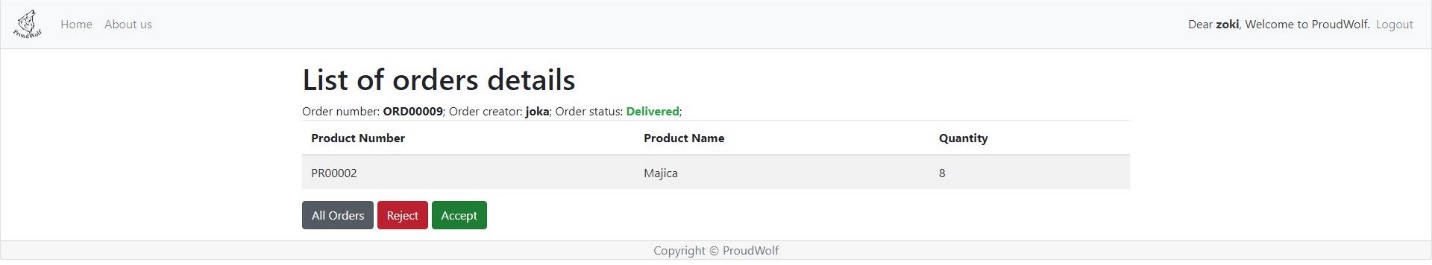


*Dodavanje slike*

Otpremanje se vrši konfiguracijom ***StandardServletMultipartResolver*** bina. Standarnom primenom interfejsa ***MultipartResolver***, zasnovanom na apiju ***Servlet 3.0.***Kreirana klasa ***FileBucket***  je omotac. A njen atribut ***MultipartFile file*** koji nam pruža ***Spring org.springframework.web.multipart.MultipartFile*** prestavlja prikaz učitane datototeke primljene u višedelnom zahtevu.

Učitanu sliku čuvamo na fajl sistemu same aplikacije u folderu ***static/photo***, što je privremeno rešenje dok se ne obezbedi server na kome bi se čuvale slike. Dok naziv slike, koji se sastoji od naziva slike onog ko otprema I jedinstvenog broja proizvoda, čuvamo u bazi podataka. Time smo obezbedili sporo otvaranje ostalih podataka iz baze dok se učita slika.

Za validaciju da je korisnik zaista odabrao sliku željenog formata I veličine koristimo klasu ***FileValidator,*** koju smo kreirali u paketu ***util***. Mora biti do 5MB velečine I formata jpg, png ili gif.



*Detalji porudžbine*

Menadžer magacina peko dugmeta ***Orders,*** gde dolazi to svih kreiranih porudžbina, zatim klikom na dugme ***Details*** dolazi do detalja svake pojedinačne poružbine. Na ovoj stranici dobija mogućnost da klikom na dugme ***Reject*** odbije porudžbinu I ujedno ispiše poruku kupcu u koloni description tabele orders - *Order been rejected. Contact the warehouse.* Klikom na dugme ***Accept,*** poružbina se automatski obradjuje. Ako postoji tražena količina proizvoda, prihvta se I količine se skidaju sa stanja u magacinu, a ispisuje poruka - *The order has been accepted. And delivered.* U slučaju da tržena količina više ne postoji, odbija se I količine u magacinu ostaju nepromenjene a ispisuje s poruka - *Physical inspection determined that the item: Majica, was not in the required quantity. Please make your new purchase order.* Ako je porudybina obradjena nije je moguće ponovo obraditi.