Web aplikacija za klasifikaciju bijelih krvnih zrnaca

Pavlović, Kristijan

Undergraduate thesis / Završni rad

2017

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:921767

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Sveučilišni studij

WEB APLIKACIJA ZA KLASIFIKACIJU BIJELIH KRVNIH ZRNACA

Završni rad

Kristijan Pavlović

Osijek, 2017.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju

Osijek, 06.09.2017.

Odboru za završne i diplomske ispite

Prijedlog ocjene završnog rada

Ime i prezime studenta:	Kristijan Pavlović
Studij, smjer:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R3684, 23.07.2014.
OIB studenta:	87350576984
Mentor:	Izv. prof. dr. sc. Irena Galić
Sumentor:	Hrvoje Leventić
Sumentor iz tvrtke:	
Naslov završnog rada:	Web aplikacija za klasifikaciju bijelih krvnih zrnaca
Znanstvena grana rada:	**Programsko inženjerstvo (zn. polje računarstvo)**
Predložena ocjena završnog rada:	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	06.09.2017.
Datum potvrde ocjene Odbora:	27.09.2017.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 27.09.2017.

Ime i prezime studenta:	Kristijan Pavlović
Studij:	Preddiplomski sveučilišni studij Računarstvo
Mat. br. studenta, godina upisa:	R3684, 23.07.2014.
Ephorus podudaranje [%]:	0%

Ovom izjavom izjavljujem da je rad pod nazivom**: Web aplikacija za klasifikaciju bijelih krvnih zrnaca**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Irena Galić

i sumentora Hrvoje Leventić

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD **1**

1.1. Zadatak završnoga rada 1

2. KRV **2**

2.1. Medicinska pozadina 2

2.2. Vrste krvnih zrnaca 2

2.2.1. Crvena krvna zrnca 2

2.2.2. Bijela krvna zrnca 3

2.3. Klase bijelih krvnih zrnaca 3

2.3.1. Neutrofili 4

2.3.2. Eozinofili 4

2.3.3. Bazofili 4

2.3.4. Limfociti 5

2.3.5. Monociti 5

3. KORIŠTENE TEHNOLOGIJE **7**

3.1. Ruby on Rails 7

3.2. ActiveRecord 8

3.3. Devise 9

3.4. Pundit 9

4. APLIKACIJA ZA KLASIFIKACIJU BIJELIH KRVNIH ZRNACA **10**

4.1. Dizajn baze podataka 10

4.2. Korisničko sučelje 13

4.2.1. Običan korisnik 14

4.2.2. Administrator 15

4.3. Mogućnosti 16

4.3.1. Osnovne mogućnosti aplikacije 16

4.3.2. Klasifikacija bijelih krvnih zrnaca 17

4.3.3. Mogućnosti administratora 19

4.4. Rezultati 21

5. ZAKLJUČAK **23**

LITERATURA **24**

SAŽETAK **25**

SUMMARY **26**

ŽIVOTOPIS **27**

1. UVOD

Krv se sastoji od krvne plazme, eritrocita (crvena krvna zrnca), leukocita (bijela krvna zrnca) i trombocita. Uloga leukocita u organizmu je obrambena, a njihov povišen broj često upućuje na pojavu bolesti u organizmu. Dijele se na granulocite i agranulocite. Granulociti se dodatno dijele na neutrofile, eozinofile i bazofile, dok su podvrste agranulocita limfociti i monociti. [1] S namjerom utvrđivanja bolesti liječnici na mikroskopskoj fotografiji ručno broje leukocite, a uz prebrojavanje leukocita vrše i njihovu klasifikaciju kako bi se povećala točnost utvrđivanja.

Cilj ovoga završnog rada je izrada mrežne aplikacije koja korisniku omogućuje odabir leukocita na mikroskopskoj fotografiji, odabir njihove klase te pohranu unosa u bazu podataka. Svrha aplikacije je unaprijediti klasifikaciju leukocita obradom baze korisničkih unosa postupkom strojnog učenja kojim bi se dugotrajan ručni postupak sklon greškama naposljetku zamijenio algoritmom koji samostalno klasificira leukocite.

Završni rad sastoji se od 3 poglavlja. Prvo poglavlje opisuje krv, vrste krvnih zrnaca te detaljno opisuje klase bijelih krvnih zrnaca. Drugo poglavlje navodi tehnologije korištene u izradi praktičnog dijela završnoga rada te sadrži objašnjenje funkcionalnosti svake navedene tehnologije. Treće poglavlje opisuje dizajn baze podataka, prikazuje korisničko sučelje, navodi mogućnosti izrađene aplikacije te navodi rezultate njene izrade.

1.1. Zadatak završnoga rada

Zadatak završnoga rada je opisati medicinsku pozadinu krvi i krvnih zrnaca te navesti tehnologije koje su korištene u izradi mrežne aplikacije. Za praktični dio rada potrebno je izraditi mrežnu aplikaciju kojoj je cilj omogućiti klasifikaciju bijelih krvnih zrnaca.

2. KRV

2.1. Medicinska pozadina

Krv je posebno vezivno tkivo s tekućom međustaničnom tvari. Ona je gusta crvena tekućina slana okusa i sladunjavog mirisa sličnog mirisu znoja. U ljudskom tijelu srčano-žilnim sustavom cirkulira između pet i šest litara krvi što je 8% ukupne tjelesne mase.

Krv je funkcionalno najvažnija tjelesna tekućina:

- prenosi kisik iz pluća u tkiva i ugljikov dioksid iz tkiva u pluća
- prenosi hranjive tvari kao što su glukoza, aminokiseline i masne kiseline do stanice
- prenosi hormone do tkiva
- prenosi leukocite, protutijela i druge tvari koje sudjeluju u obrani organizma od štetnih tvari
- regulira količinu vode u stanicama i međustaničnom prostoru
- održava tjelesnu temperaturu prenoseći toplinu po tijelu
- regulira krvni tlak

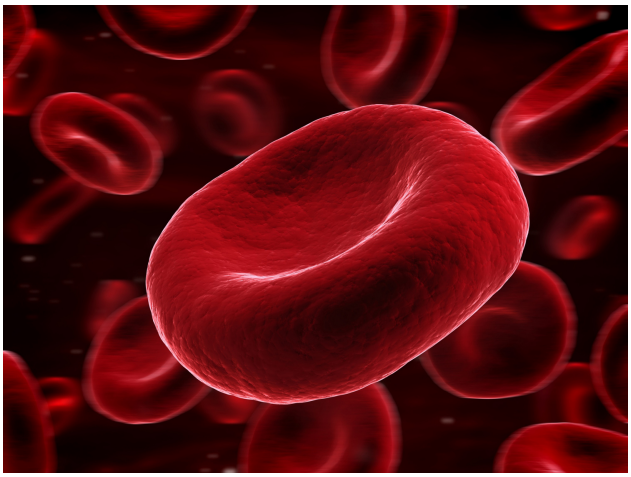
Izvan krvožilnog sustava krv se gruša, stoga se krv namijenjena analizi u procesu vađenja miješa sa antikoagulansima ili sredstvima protiv zgrušavanja.

2.2. Vrste krvnih zrnaca

Analizom krvi utvrđeno je da se sastoji od tekuće krvne plazme (55% ukupne količine krvi) i triju vrsta krvnih tjelešaca (45% ukupne količine krvi). Krvna tjelešca su krvna zrnca - eritrociti (crvena krvna zrnca), leukociti (bijela krvna zrnca) i trombociti.

2.2.1. Crvena krvna zrnca

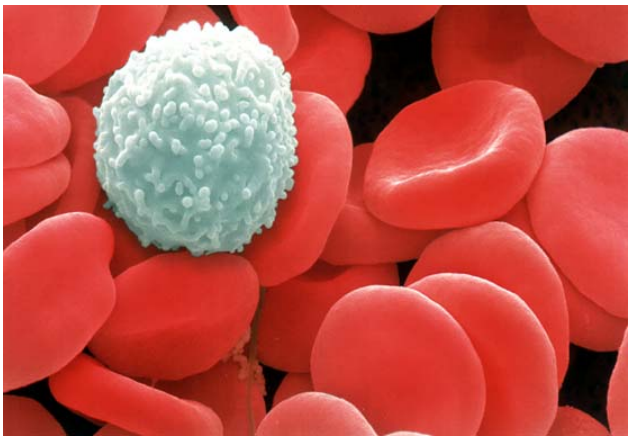
Crvena krvna zrnca (eritrociti) jedine su stanice u ljudskom tijelo koje nemaju jezgru. Membrana im je elastična, što im omogućava prolazak kroz tanke kapilare. Oblikom podsjećaju na disk udubljen u središtu, imaju promjer oko 8 μm i debljine su oko 2 μm (Sl. 2.1.). Osnova uloga eritrocita je prijenos kisika iz pluća do tkiva i ugljikov dioksid iz tkiva do pluća.



Sl. 2.1.: Grafički prikaz eritrocita
Izvor: http://news.rice.edu/wp-content/uploads/2012/09/0914_BLOOD_lg.jpg

2.2.2. Bijela krvna zrnca

Bijela krvna zrnca (leukociti) u organizmu imaju obrambenu ulogu, štite ga od mikroorganizama (bakterija, gljivica, virusa i parazita) te spriječavaju infekcije. Veća su od crvenih krvnih zrnaca i imaju jezgre različitog izgleda (Sl. 2.2.). Povišen broj bijelih krvnih zrnaca često indicira na bolesti. Kod zdravih i odraslih osoba, u jednom kubičnom milimentru krvi, nalazi ih se od 5 do 8 tisuća. Bijela krvna zrnca nastaju u koštanoj srži, ali dijelom i u prsnoj žlijezdi. Njihov životni vijek ovisi o njihovoj aktivnosti u tijelu, a može biti od nekoliko dana do nekoliko godina.



Sl. 2.2.: Grafički prikaz leukocita
Izvor: http://www.just-health.net/images/10416321/image001.jpg

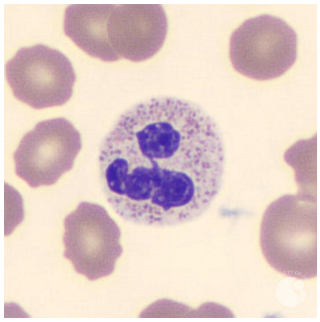
2.3. Klase bijelih krvnih zrnaca

Bijela krvna zrnca dijele se na granulocite i agranulocite. Granulociti su najbrojnija vrsta leukocita te obuhvaćaju više od 50 % svih leukocita u cirkulaciji. Oni sadrže zrnca (granule) u citoplazmi koja se različito boje tako da se razlikuju blijedoljubičasti neutrofili, crveni eozinofili i modri bazofili.

Agranulociti ne sadrže obojana zrnca u citoplazmi, te uglavnom kruže limfnim sustavom. U njih spadaju limfociti i monociti.

2.3.1. Neutrofili

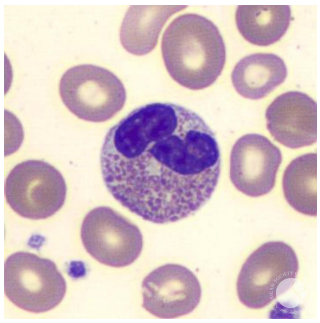
Zadaća neutrofila je borba protiv štetnih bakterija i gljivica te je tijekom infekcije njihov broj povećan. Oni lažnim nožicama obuhvaćaju štetnike i razgrađuju ih u citoplazmi. [1] Granule u citoplazmi su im blijedoljubičaste boje, a jezgra im (nukleus) je sačinjena od više režnjeva. [1] [2] Životni vijek im iznosi od 6 sati do nekoliko dana. [3]



Sl. 2.3.: Mikorskopski prikaz neutrofila
Izvor: https://imagebank.hematology.org/getimagebyid/60395?size=3

2.3.2. Eozinofili

Eozinofili se vežu za parazite te otpuštaju štetne tvari koje ih neutraliziraju. Njihov je broj u krvi povećan kod prisutnosti nametnika u crijevima pri alergijskim reakcijama koje se javljaju udisanjem. Granule u citoplazmi su im crvene boje, a jezgra im je sačinjena od dva režnja. [1] [2] Životni vijek im je od 8 do 12 dana (u krvotoku oko 5 sati). [3]

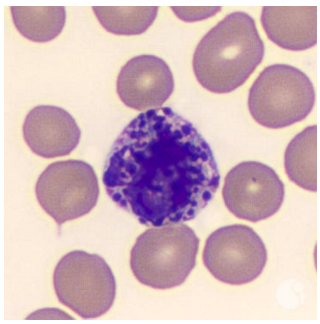


Sl. 2.4.: Mikorskopski prikaz eozinofila
Izvor: https://imagebank.hematology.org/getimagebyid/60934?size=3

2.3.3. Bazofili

Bazofili otpuštaju heparin - tvar koja spriječava zgrušavanje krvi u krvnim žilama te histamin - tvar koja u slučaju infekcije povećava propusnost kapilara za fagocite (neutrofile i monocite) koji prelaze

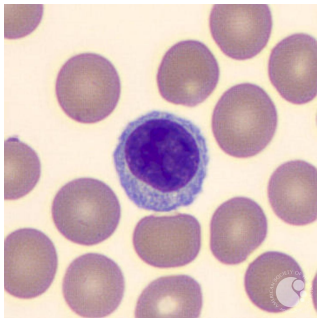
iz krvi u izvanstaničnu tekućinu. Granule u citoplazmi su im modre boje, a jezgra im je sačinjena od dva ili tri režnja. [1] [2] Životni vijek im je od nekoliko sati do nekoliko dana. [3]



Sl. 2.5.: Mikorskopski prikaz bazofila
Izvor: http://imagebank.hematology.org/getimagebyid/60504?size=3

2.3.4. Limfociti

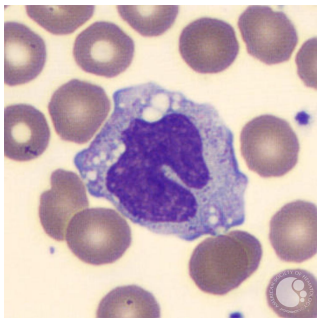
Limfociti imaju sposobnost sinteze protutijela koja se bore protiv različitih mikroorganizama. Imaju važnu ulogu u imunosnim reakcijama. Stvaranje i propadanje limfocita odvija se brzo te se njihova cijela populacija može izmijeniti u roku 24 sata. Imaju veliku jezgru koja ispunjava najveći dio stanice. [1]



Sl. 2.6.: Mikorskopski prikaz limfocita
Izvor: https://imagebank.hematology.org/getimagebyid/60510?size=3

2.3.5. Monociti

Monociti su vrlo velike stanice koje nakon što iz krvnih žila izađu u tkivo stječu sposobnost fagocitoze te proždiru bakterije i druge štetne mikroorganizme. [1] Često imaju oblik kao bubreg. [2] Životni vijek im je od nekoliko sati do nekoliko dana. [3]



Sl. 2.7.: Mikorskopski prikaz monocita
Izvor: https://imagebank.hematology.org/getimagebyid/60935?size=3

3. KORIŠTENE TEHNOLOGIJE

3.1. Ruby on Rails

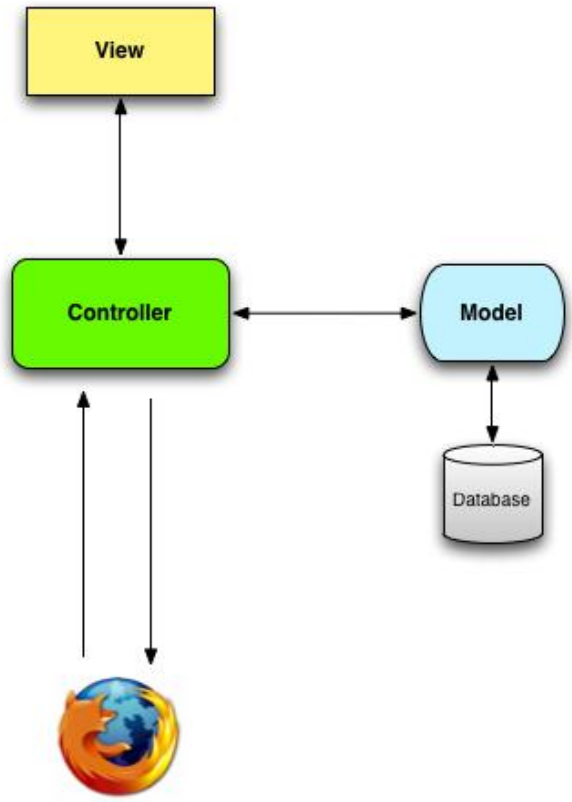
Ruby on Rails, ili jednostavno Rails, je radni okvir (engl. *framework*) napisan u programskom jeziku Ruby koji se nalazi na strani poslužitelja te je korišten za razvoj mrežnih (engl. *web*) aplikacija. Izvorno je nastao 2005. godine te je zaštićen pod MIT lincencom. Dizajniran je da učini izradu mrežnih aplikacija jednostavnom tako da programer piše malo koda, ali njime uspjeva odraditi više nego u drugim programskim jezicima ili radnim okvirima. [4]

Rails je radni okvir otvorenog koda te zbog toga ima vrlo aktivnu i korisnu zajednicu programera koji su napisali velik broj takozvanih dragulja (engl. *gem*). [5] Dragulji se koriste da se proširi funkcionalnost aplikacija i dostupni su svim Rails programerima na korištenje. Tako svaki programer ne mora pisati iznova kod, nego može koristiti već gotovi kod uz blage preinake. Dragulji se upotrebljavaju za neke osnovne funkcionanosti aplikacije kao što su provjera autentičnosti (engl. *authentication*) i autorizacija (engl. *authorization*). Svaki dragulj ima naziv, verziju i platformu na kojoj može biti pokretan (primjer: rake, 12.0.0, ruby), a sastoji se od koda koji izvršava određenu radnju, dokumentacije i specifikacije. [6]

Filozofija Railsa sadrži dva glavna principa izrade aplikacija:

- Ne ponavljaj se (engl. *Don't Repeat Yourself - DRY*) - princip razvoja aplikacija gdje programer ne bi trebao pisati više puta jednak kod, nego koristiti taj već napisani kod. Time se kod aplikacije čini održivijim, proširivijim i s manje greški.
- Radije konvencija nego konfiguracija (engl. *Convention over Configuration - CoC*) - Rails ima "mišljenje" o tome koji je najbolji način za učiniti nešto u mrežnoj aplikaciji te bi se trebalo držati tog načina, a ne konfigurirati po željama ili po nekim starim navikama. [4]

Rails koristi MVC (engl. *Model-view-controller*, odnosno Model-pogled-upravljač) razvoja mrežnih aplikacija koji dijeli podatke u aplikaciji od koda koji ih prikazuje (Sl. 3.1.). Mrežni preglednik šalje zahtjev poslužitelju te ga on prosljeđuje upravljaču koji odlučuje koji je sljedeći korak. U nekim će slučajevima upravljač odmah generirati pogled, što je predložak koji se pretvori u HTML i vrati nazad pregledniku, dok će u drugim slučajevima upravljač stupiti u interakciju s modelom, što je Ruby objekt koji predstavlja određeni element stranice (npr. model je korisnik) i koji je zadužen za komunikaciju s bazom podataka te tek onda generirati pogled. Nakon bilo kojeg od navedenih slučajeva upravljač vraća pregledniku pogled te ga preglednik prikazuje. [5]



Sl. 3.1.: Shematski prikaz MVC principa
Izvor: https://softcover.s3.amazonaws.com/636/ruby_on_rails_tutorial_4th_edition/images/figures/mvc_schematic.png

3.2. ActiveRecord

ActiveRecord je implementacija Active Record uzorka, što je zapravo preslika ORM (engl. *Object Relational Mapping*) sustava.

ORM je tehnika kojom se povezuju objekti aplikacije s tablicama relacijske baze podataka. Koristeći ORM, sva svojstva objekata i veze između objekata u aplikaciji mogu se pohraniti, kao i dohvatiti iz baze podataka bez direktnog pisanja SQL upita.

ActiveRecord daje nekolicinu mehanizama, od kojih su najbitniji:

- predstavljanje modela i njegovih podataka
- predstavljanje veza između modela
- predstavljanje hijerarhije nasljeđa preko povezanih modela
- potvrđivanje ispravnosti modela prije slanja u bazu
- izvršavanje operacija u bazi podataka u objektno orijentiranom stilu [4]

3.3. Devise

Devise je najpoznatiji i najkorišteniji dragulj korišten u svrhu provjere autentičnosti. Baziran je na konceptu modularnosti, gdje svaki programer može koristiti samo onaj dio koji mu je potreban.

Sačinjen je od 10 modula:

- Database Authenticatable - odgovoran za sigurno spremanje lozinke u bazu podataka i za provjeru autentičnosti korisnika prilikom prijave [7]
- Omniauthable - dodaje podršku za OmniAuth koji omogućuje provjeru autentičnosti za mrežne aplikacije na više načina (primjerice prijava putem Google ili Facebook računa) [7] [8]
- Confirmable - šalje e-mail s uputama za potvrdu računa, te provjerava je li korisnički račun potvđen prilikom prijave
- Recoverable - resetira korisnikovu lozinku te mu šalje upute za njenu izmjenu
- Registerable - rukuje s registracijom korisnika te im dopušta da izmjene svoj korisnički račun ili ga obrišu
- Rememberable - sprema ili briše generirani žeton (engl. *token*) u ili iz preglednika koji ima svrhu pamćenja korisnika
- Trackable - prati broj prijava korisnika, vrijeme prijave i IP adresu s koje se korisnik prijavio
- Timeoutable - odgovoran za isticanje sesija koje nisu bile aktivne u određenom vremenskom razdoblju
- Validatable - pruža korisniku mogućnost potvrde e-maila i lozinke
- Lockable - zaključava korisnički račun nakon određenog broja bezuspješnih pokušaja prijave [7]

3.4. Pundit

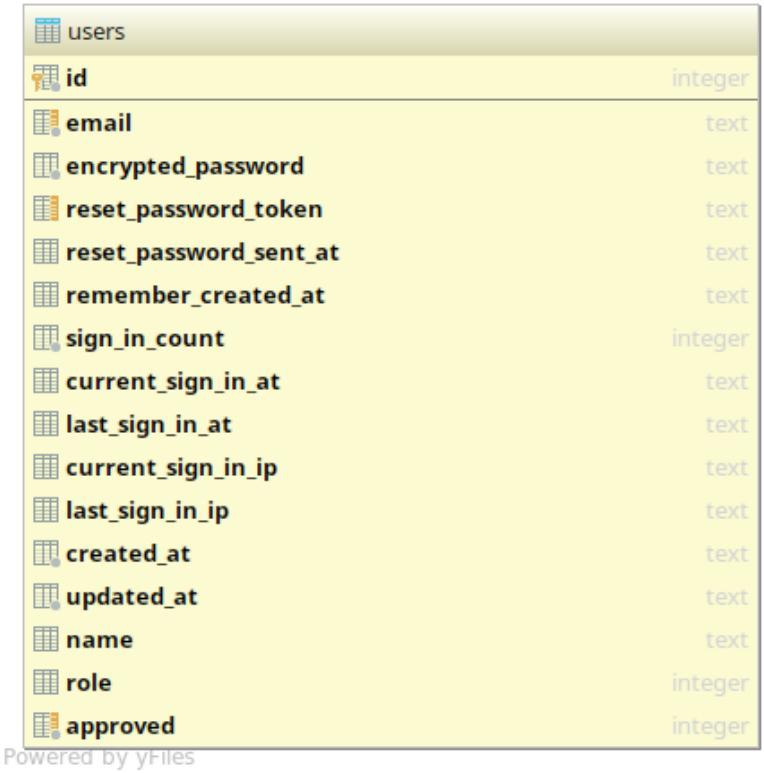
Pundit je jedan od najjednostavnijih i najkorištenijih dragulja koji se koriste u svrhu autorizacije. On koristi Ruby klase i objektno orijentirani način dizajna za izdradu jednostavnog i održivog sustava za autorizaciju. Glavni mu je zadatak dopuštanje ili zabrana korisniku da pristupa određenom dijelu aplikacije (pogled) ili da vrši određene radnje (upravljač). [9]

4. APLIKACIJA ZA KLASIFIKACIJU BIJELIH KRVNIH ZRNACA

4.1. Dizajn baze podataka

Budući da je nužno da aplikacija ima korisnike, od tog se zahtjeva počelo dizajnirati bazu podataka. Kreiran je model User, koji je zbog ActiveRecorda odmah dobio svoju tablicu u bazi podataka pod nazivom "users" (Sl. 4.1.). Od prikazanih atributa najbitniji su šifra ("id"), e-mail ("e-mail"), ime ("name"), kriptirana lozinka ("encrypted_password"), uloga ("role") te je li korisnik potvrđen ("approved").

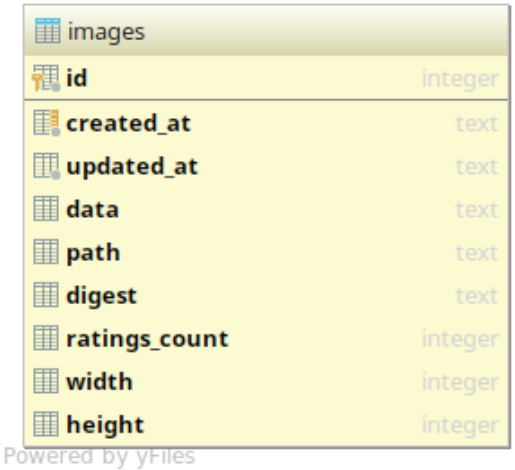
Korisnici su podijeljeni u dvije grupe, tj. imaju jednu od dvije mogućih uloga. Oni mogu biti obični korisnici, ali i administratori. Svaki korisnik nakon registracije nije potvrđen i tek nakon što ga administrator potvrdi, on može pristupiti aplikaciji. Također, svaki korisnik nakon registracije dobiva ulogu običnog korisnika i samo administrator može promijeniti njegovu ulogu.



Sl. 4.1.: Shematski prikaz tablice "users"

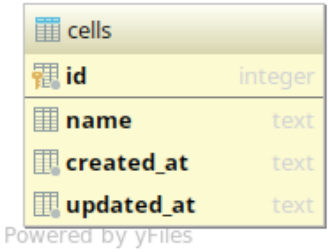
Svaki korisnik mora imati mogućnost odabira leukocita kao i odabira njihove klase na danoj mikroskopskoj fotografiji. Po tim zahtjevima su kreirani modeli Image i Cell, gdje prethodni predstavlja mikroskopske fotografije, a potonji klase krvnih zrnaca.

Model Image je u bazi podataka predstavljen tablicom "images" (Sl. 4.2.) čiji su atributi šifra("id"), putanja do fotografije ("path"), širina ("width"), visina ("height") te broj koliko je korisnika klasificiralo krvna zrnca na toj fotografij ("ratings_count"). Putanja do fotografije je potrebna budući da je ona spremljena na strani poslužitelja, a ne u bazi podataka.



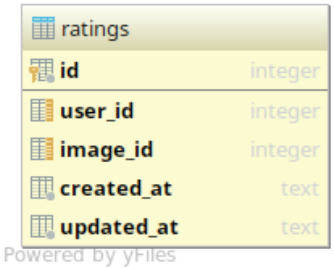
Sl. 4.2.: Shematski prikaz tablice "images"

Model Cell je u bazi podataka predstavljen tablicom "cells" (Sl. 4.3.) čiji su atributi šifra ("id") i naziv klase ("name").



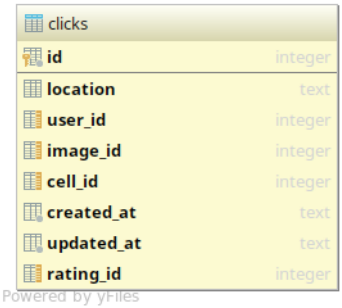
Sl. 4.3.: Shematski prikaz tablice "cells"

Za potrebe spremanja lokacije korisnikovog klika na leukocit i za spremanja klase odabranog leukocita kreirani su modeli Click i Rating. Model Rating je u bazi podataka predstavljen tablicom "ratings" (Sl. 4.4.) čiji su atributi šifra ("id"), šifra korisnika koji je klasificirao leukocite ("user_id") i šifra fotografije na kojoj ih je klasificirao ("image_id").



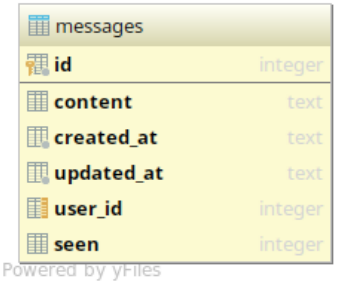
Sl. 4.4.: Shematski prikaz tablice "ratings"

Model Click je u bazi podataka predstavljen tablicom "clicks" (Sl. 4.5.) čiji su atributi šifra ("id"), šifra fotografije na kojoj je korisnik klasificirao leukocita ("image_id"), šifra korisnika koji je to učinio ("user_id"), lokacija leukocita ("location") te šifra klase tog leukocita ("cell_id").



Sl. 4.5.: Shematski prikaz tablice "clicks"

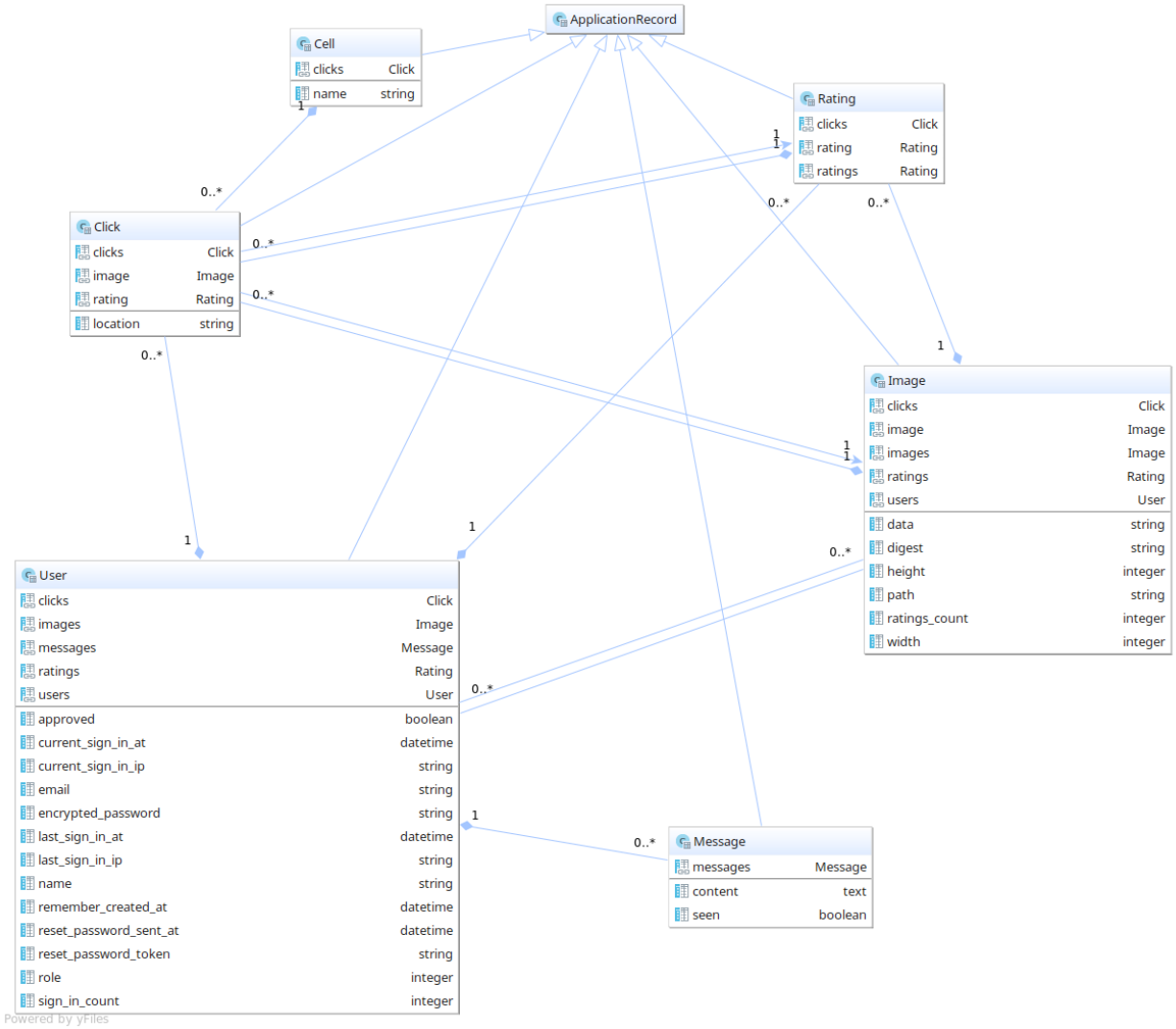
Uz te funkcionalnosti, odlučeno je da će se svaki korisnik obavijestiti u slučaju da su u aplikaciju dodane nove mikroskopske fotografije. Ta se funkcionalnost omogućila kreiranjem Message modela koji je u bazi podataka predstavljen tablicom "messages" (Sl. 4.6.). Tablica kao atribute sadrži šifru ("id"), sadržaj obavijesti ("content"), šifru korisnika za kojeg je obavijest ("user_id") te oznaku koja označava je li korisnik obavijest označio kao pročitanu ("seen").



Sl. 4.6.: Shematski prikaz tablice "messages"

Grafički prikaz svih tablica baze podataka uz označene veze između njenih tablica može se vidjeti na Sl. 4.7.. Može se primjetiti kako je svaka tablica povezana s oznakom ApplicationRecord.

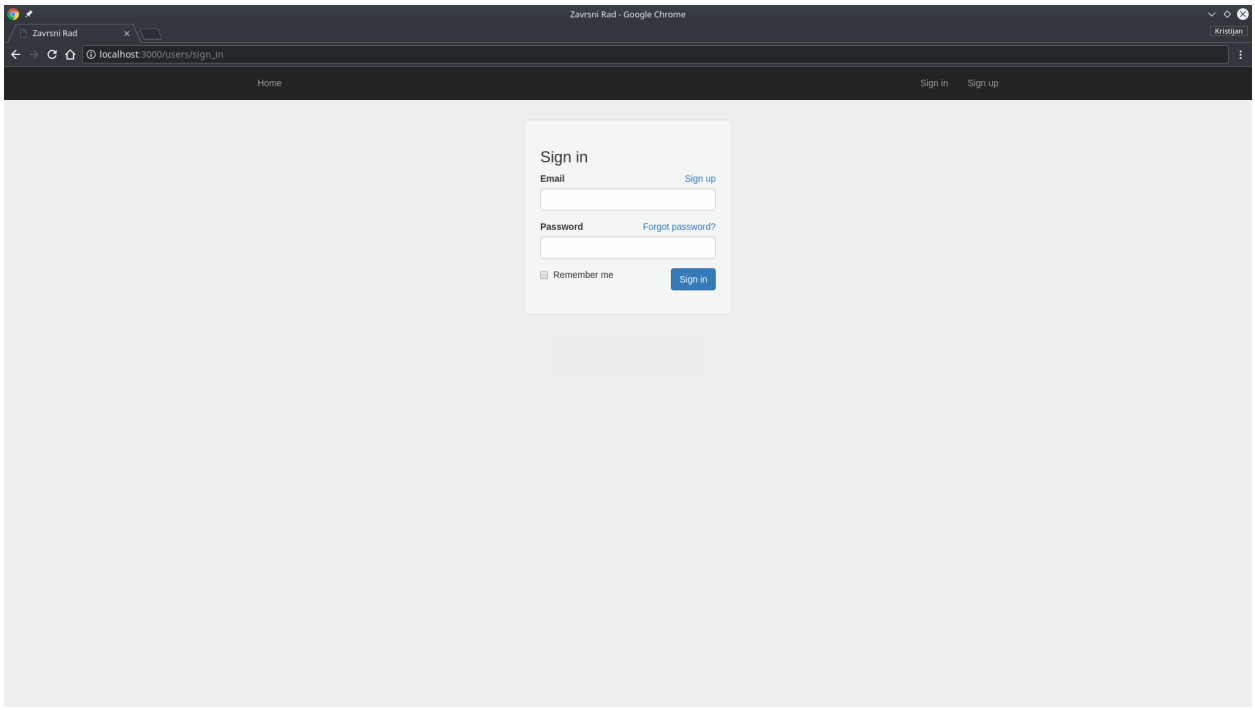
To je zapravo klasa koja nasljeđuje funkcionalnosti klase ActiveRecord::Base koja modelima omogućuje razne funkcionalnosti kao što je opisano u poglavlju 3.2..



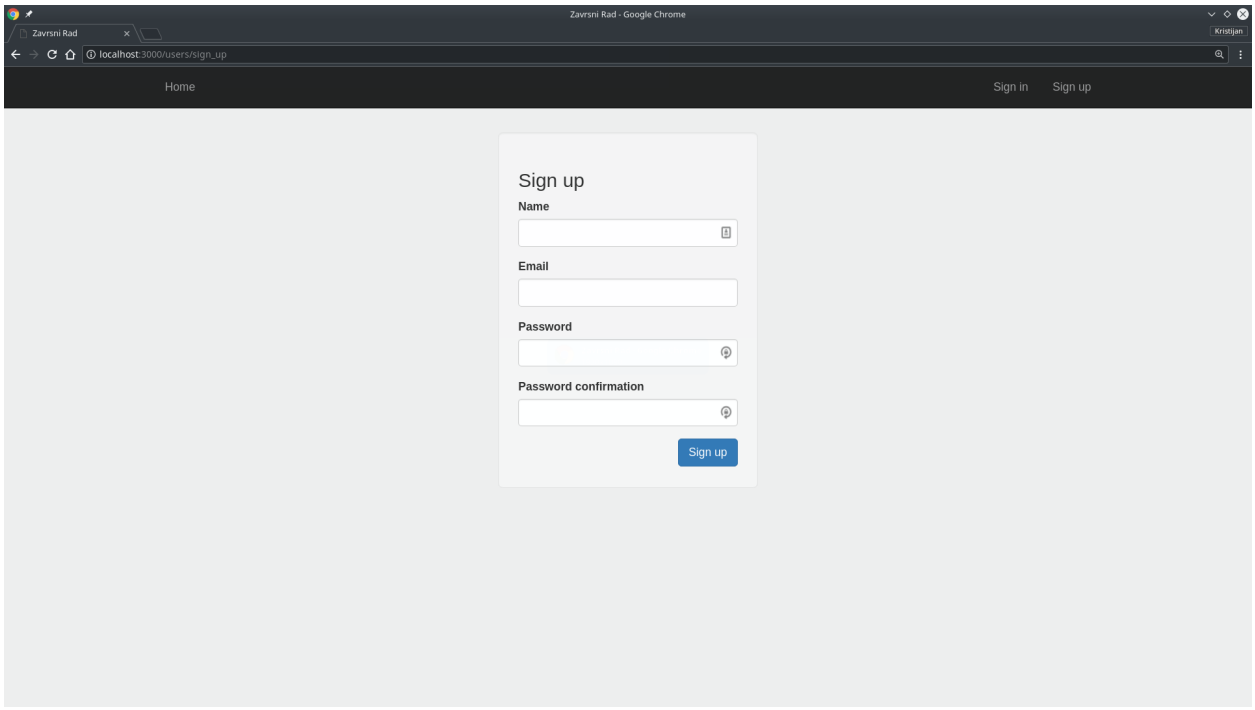
Sl. 4.7.: Shematski prikaz baze podataka

4.2. Korisničko sučelje

Prilikom ulaska u aplikaciju, svaki gost je preusmjeren na stranicu za prijavu (engl. *sign in*) gdje mora upisati tražene podatke korisničkog računa i prijaviti se (Sl. 4.8.) kao korisnik. Ukoliko nema korisnički račun, može se registrirati tako da ode na stranicu za registraciju (engl. *sign up*) klikom na "Sign up" gumb te ispuni tražene podatke prikazane na slici Sl. 4.9..



Sl. 4.8.: Stranica za prijavu korisnika

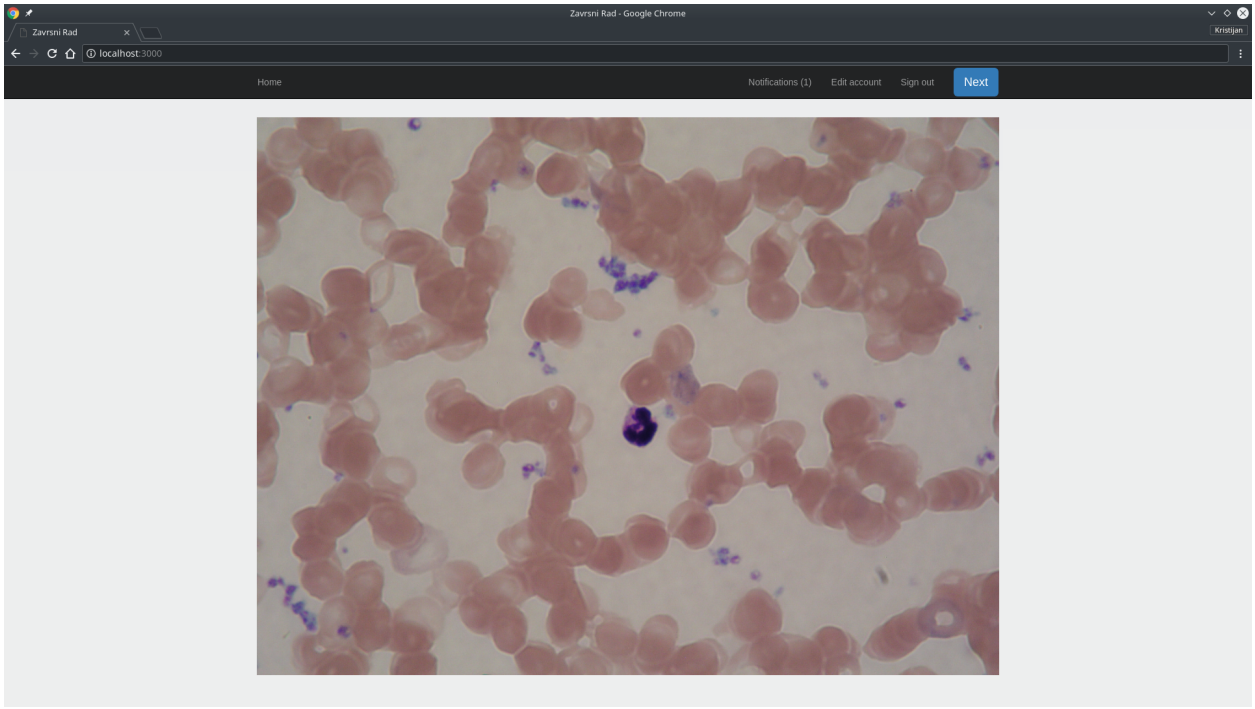


Sl. 4.9.: Stranica za izradu korisničkog računa

4.2.1. Običan korisnik

Nakon prijave, korisniku je prikazana početnu stranicu aplikacije na kojoj je prikazana mikroskopska fotografija krvi (Sl. 4.10.). Klikom na "Notifications" gumb korisniku se prikažu obavijesti koje većinom govore da je postavljeno nekoliko novih fotografija. Korisnik također može izmijeniti

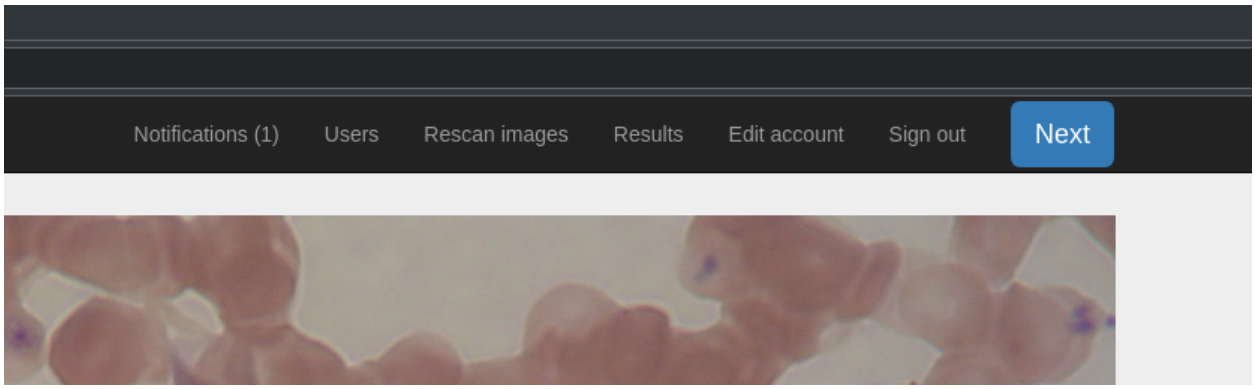
podatke korisničkog računa klikom na "Edit account" gumb ili se odjaviti iz aplikacije klikom na "Sign out" gumb.



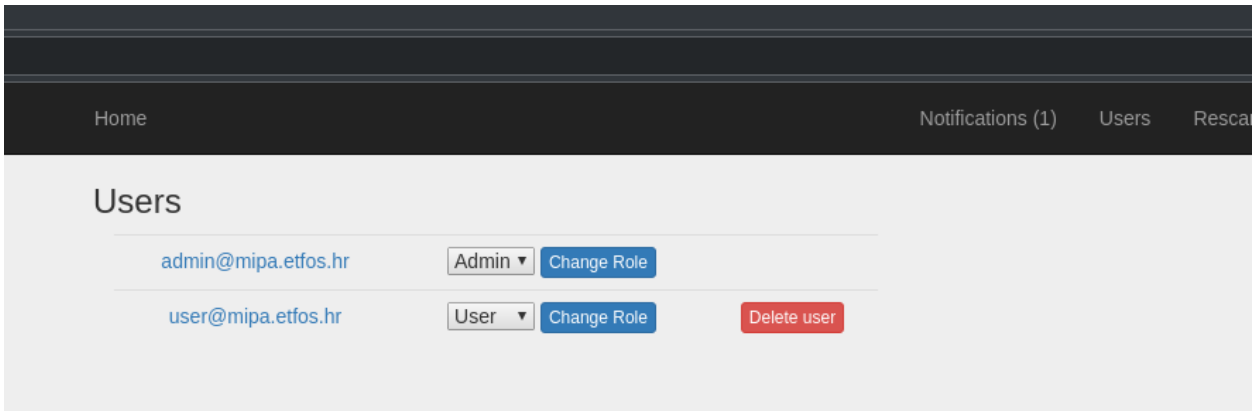
Sl. 4.10.: Početna stranica aplikacije

4.2.2. Administrator

Prikaz nakon prijave administratora je nešto drugačiji kao što je prikazano na slici Sl. 4.11.. On uz sve mogućnosti običnog korisnika ima i dodatne mogućnosti kao što su prikaz svih korisnika klikom na "Users" gumb (Sl. 4.12.), prikaz svih rezultata klikom na "Results" gumb ili pak mogućnost slanja naredbe poslužitelju da provjeri ima li novih fotografija klikom na "Rescan images" gumb.



Sl. 4.11.: Prikaz administratorovog izbornika



Sl. 4.12.: Prikaz svih korisnika

4.3. Mogućnosti

4.3.1. Osnovne mogućnosti aplikacije

Osnovne mogućnosti aplikacije su registracija novih korisnika, kao i prijava, odjava, promjena podataka korisničkog računa ili pak brisanje korisničkog računa postojećih korisnika. Sve navedeno je omogućeno koristeći dragulj Devise koji je opisan u poglavlju 3.3..
Upisivanjem naredbe "rails generate devise:install" u ljusku instalira se Devise na željeni projekt, dok se upisivanjem "rails generate devise User" kreira model imena User (ako već ne postoji) koji se konfigurira po osnovnim Devise modulima. Devise također generira i HTML kodove koji sačinjavaju osnovne poglede povezane s danim modelom. [7] Generairani HTML kod koji sačinjava pogled na slici Sl. 4.8. prikazan je na slici Sl. 4.13..
Na početku svakog upravljača gdje je nužna provjera autentičnosti potrebno je napisati naredbu "before_action :authenticate_user!". Ta naredba znači da će prije svake radnje koju treba izvršiti upravljač provjeriti je li korisnik prijavljen u aplikaciju i ako nije, zabranit će mu se vršenje te radnje. [7]

```
<div class="authform">
  <%= form_for(resource, :as => resource_name, :url =>
  registration_path(resource_name), :html => { :role => 'form'}) do |f| %>
    <h3>Sign up</h3>
    <%= devise_error_messages! %>
    <div class="form-group">
      <%= f.label :name %>
      <%= f.text_field :name, :autofocus => true, class: 'form-control' %>
    </div>
    <div class="form-group">
      <%= f.label :email %>
      <%= f.email_field :email, class: 'form-control' %>
    </div>
    <div class="form-group">
      <%= f.label :password %>
      <%= f.password_field :password, class: 'form-control' %>
    </div>
    <div class="form-group">
      <%= f.label :password_confirmation %>
      <%= f.password_field :password_confirmation, class: 'form-control' %>
    </div>
     <%= f.submit 'Sign up', :class => 'button right' %>
  <% end %>
</div>
```

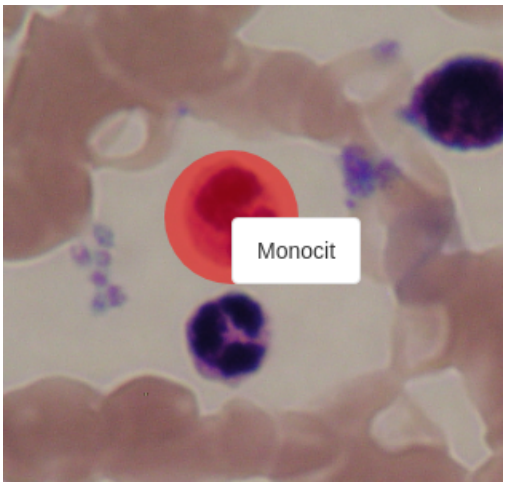
Sl. 4.13.: HTML kod koji sačinjava pogled prikazan na slici Sl. 4.8.

4.3.2. Klasifikacija bijelih krvnih zrnaca

Pri svakom kliku na mikroskopsku fotografiju korisniku se prikaže padajući izbornik s klasama leukocita prikazan na slici Sl. 4.14.. Nakon što korisnik odabere određenu klasu, na fotografiji se klasificirani leukocit označi crvenom bojom te natpisom o kojoj se klasi radi (Sl. 4.15.).



Sl. 4.14.: Padajući izbornik koji sadrži klase bijelih krvnih zrnaca



Sl. 4.15.: Prikaz klasificiranog leukocita

Tijekom korisnikovog klasificiranja leukocita, svaki njegov odabir sprema se u skrivenu formu (Sl. 4.16.) tako što se ona ispunjava s koordinatama klika i klasom leukocita. Pritiskom na "Next" gumb skrivena forma koja sadrži navedene podatke i šifru fotografije se šalje upravljaču RatingsController koji ju obrađuje.

```
<div class="authform">
  <%= form_for(@rating) do |f| %>
    <%= text_area_tag :coordinates, params[:coordinates], class:
        'form-control', style: "display: none", autocomplete: "off" %>
    <%= text_area_tag :cellTypes, params[:cellTypes], class:
        'form-control', style: "display: none", autocomplete: "off" %>
    <%= hidden_field(:image, :id) %>
    <%= f.submit "Submit",
                 id: "submitButton",
                 class: "btn btn-primary btn-lg btn-block",
                 style: "display: none"; %>
  <% end %>
</div>
```

Sl. 4.16.: HTML kod forme

Upravljač RatingsController (Sl. 4.17.) pokušava, koristeći metodu modela Rating (Sl. 4.18.) nazvanu "save_rating_and_clicks", spremiti sve podatke u bazu podataka. U slučaju da uspije, korisnik se preusmjeri na početnu stranicu i prikaže mu se poruka da je fotografija uspješno obrađena, dok mu se u suprotnom prikaže poruka da fotografija nije uspješno obrađena.

```ruby
class RatingsController < ApplicationController
  def create
    @image = Image.find(params[:image][:id])
    @rating = Rating.new
    if @rating.save_rating_and_clicks(current_user, @image,
    params[:coordinates], params[:cellTypes] )
      redirect_to root_path, notice: "Image processed."
    else
      redirect_to root_path, notice: "Unable to process image."
    end
  end
end
```

Sl. 4.17.: Prikaz upravljača RatingsController

```ruby
class Rating < ApplicationRecord
  belongs_to :user
  belongs_to :image, counter_cache: true
  has_many :clicks, dependent: :destroy
  validates :image_id, uniqueness: {scope: [:user_id]}

  def save_rating_and_clicks(user, image, coords, cells)
    ActiveRecord::Base.transaction do
      self.user_id = user.id
      self.image_id = image.id
      coordinates = coords.split(' ')
      cell_types = cells.split(' ')
      self.save!

      coordinates.zip(cell_types).each do |coord, cell|
        click = user.clicks.build(image: image, cell_id: cell, rating_id:
        self.id, location: coord)
        click.save!
      end
    end
  end
end
```

Sl. 4.18.: Prikaz modela Post

4.3.3. Mogućnosti administratora

Glavne mogućnosti administratora su upravljanje korisnicima, provjera ima li novih fotografija te prikaz svih rezultata. Obični korisnici ne mogu pristupiti tim stranicama jer im je to ograničeno koristeći dragulj Pundit (opisan u poglavlju 3.4.). On, nakon dodavanja linije koda "include Pundit" u upravljač ApplicationController i pokretanja naredbe "rails g pundit:install" u ljusci, dobiva mogućnost da određenim korisnicima omogući pristup nekim stranicama ili nekim radnjama, dok drugima zabrani. [9] Koristeći datoteku prikazanu na slici Sl. 4.19. postavljena su pravila koje radnje u upravljaču UserController smije vršiti koji korisnik.

```ruby
class UserPolicy
  attr_reader :current_user, :model

  def initialize(current_user, model)
    @current_user = current_user
    @user = model
  end

  def index?
    @current_user.admin?
  end

  def show?
    @current_user.admin? or @current_user == @user
  end

  def update?
    @current_user.admin?
  end

  def destroy?
    return false if @current_user == @user
    @current_user.admin?
  end
end
```

Sl. 4.19.: Prikaz datoteke gdje su postavljena pravila za Pundit

Prikaz stranice sa svim korisnicima (Sl. 4.12.) omogućeno je samo administratoru. Administrator također ima pravo, koristeći stranicu za prikaz svih korisnika, obrisati pojedine korisnike (osim ako su i oni administratori) kao i promijeniti ulogu korisnika ili ga potvrditi.
Upravljač UsersController (Sl. 4.20.) obrađuje sve zahtjeve vezane za korisnike, gdje se koristeći jednostavnu naredbu "authorize user" provjerava u datoteci prikazanoj na slici Sl. 4.19. je li trenutni korisnik autoriziran za vršenje te radnje. Također, dodavanjem naredbe "after_action :verify_authorized" na početak upravljača osigurava da se programera upozori ukoliko je u datoteci prikazanoj na slici Sl. 4.19. navedena provjera, a on ju nije pozvao pri određenoj radnji u upravljaču. [9]

```ruby
class UsersController < ApplicationController
  before_action :authenticate_user!
  after_action :verify_authorized

  def index
    @users = User.all.order(created_at: :desc)
    authorize User
  end

  def show
    @user = User.find(params[:id])
    authorize @user
    @images = @user.images
  end

  def update
    @user = User.find(params[:id])
    authorize @user

    if params[:approve]
      if @user.toggle!(:approved)
        redirect_to users_path, :notice => "User approved."
      else
        redirect_to users_path, :alert => "Unable to approve user."
      end
    elsif params[:change_role]
      if @user.update_attributes(secure_params)
        redirect_to users_path, :notice => "User's role updated."
      else
        redirect_to users_path, :alert => "Unable to update user's role."
      end
    end
  end

  def destroy
    user = User.find(params[:id])
    authorize user
    user.destroy
    redirect_to users_path, :notice => "User deleted."
  end

  private

  def secure_params
    params.require(:user).permit(:role)
  end
end
```

Sl. 4.20.: Prikaz upravljača UsersController

4.4. Rezultati

Mrežna aplikacija je uspješno izrađena i postavljena na poslužitelja na kojem je njena funkcionalnost dodatno isprobana od strane korisnika. Ona je izrađena koristeći Ruby verzije 2.3.1 i Ruby on Rails verzije 5.0.1, te je u tom obliku postavljena i na poslužitelja.

Administrator može vidjeti sve rezultate, tj. što je na fotografijama i kako klasificirano, klikom na "Results" gumb. Rezultati su sačinjeni od koordinata korisnikovog klika, putanje do fotografije, šifre klase leukocita i naziva klase leukocita. Prikaz rezultata prikazan je na slici Sl. 4.21..



Sl. 4.21.: Prikaz rezultata

5. ZAKLJUČAK

U završnome radu opisana je medicinska pozadina iza krvi i krvnih zrnaca te su opisane tehnologije korištene u izradi praktičnog dijela rada. Uz navadeno, opisan je i prikazan dizajn baze podataka, prikazano je korisničko sučelje i opisane su mogućnosti aplikacije.

Za praktični dio rada izrađena je zadana mrežna aplikacija koristeći radni okvir Ruby on Rails koji je omogućio učinkovitu izradu aplikacije. Njegova filozofija omogućila je pisanje koda za višekratnu upotrebu, te kod koji vrši određenu radnju nije bilo potrebno ponovno pisati. Koristeći njegove dragulje, Devise i Pundit, jednostavno su se implementirale mogućnosti registracije, prijave i odjave korisnika (provjera autentičnosti) kao i mogućnost provjere ima li korisnik prava vršiti određenu radnju ili pristupati određenom dijelu stranice (autorizacija). Nakon kreiranja osnovne aplikacije, dodane su joj mogućnosti specifične za temu rada kao što su odabir bijelih krvnih zrnaca na mikroskopskoj fotografiji, odabir klasa tih bijelih krvnih zrnaca i spremanje odabranog.

LITERATURA

[1] S. Đumlija, M. Heffer, I. Drenjančević, *Biologija 3*. Alfa d.d., Zagreb, 2014.

[2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, P. Walter, *Molecular Biology of the Cell (4th ed.)*. Garland Science, New York, 2002.

[3] P. Wheater, H. Burkitt, V. Daniels, *Functional histology: A text and colour atlas*. Churchill Livingstone, Edinburgh, 1979.

[4] "Getting Started with Rails - Ruby on Rails Guides", Guides.rubyonrails.org. Dostupno na: http://guides.rubyonrails.org [Pristupano: 29. lipnja 2017.].

[5] M. Hartl, *Ruby on Rails Tutorial: Learn Web Development with Rails (4th ed.)*. Addison-Wesley Professional, Boston, 2016.

[6] "Guides - RubyGems Guides", Guides.rubygems.org. Dostupno na: http://guides.rubygems.org [Pristupano: 29. lipnja 2017.].

[7] "plataformatec/devise", GitHub. Dostupno na: https://github.com/plataformatec/devise. [Pristupano: 29. lipnja 2017.].

[8] "omniauth/omniauth", GitHub. Dostupno na: https://github.com/omniauth/omniauth. [Pristupano: 29. lipnja 2017.].

[9] "elabs/pundit", GitHub. Dostupno na: https://github.com/elabs/pundit. [Pristupano: 29. lipnja 2017.].

SAŽETAK

Problem završnoga rada predstavlja izrada mrežne aplikacije za klasifikaciju bijelih krvnih zrnaca. Aplikacija je uspješno izrađena koristeću Ruby verzije 2.3.1 i Ruby on Rails verzije 5.0.1 te je postavljena na poslužitelja gdje je njena funkcionalnost dodatno isprobana od strane korisnika. U radu se opisuje medicinska pozadina krvi i krvnih zrnaca, kao i tehnologije korištene pri izradi aplikacije. Dodatno je opisan i prikazan dizajn baze podataka, prikazano korisničko sučelje i opisano mogućnosti aplikacije. Filozofija Ruby on Railsa sačinjena od dva principa - "Ne ponavljaj se" i "Konvencija prije nego konfiguracija" omogućila je brzu i učinkovitu izradu aplikacije. Brzini izrade aplikacije su također pomogli i dragulji koji omogućuju aplikaciji korištenje gotovog koda koji je dostupan svim Rails programerima. Od dragulja korištenih u ovoj aplikaciji najznačajniji su bili Devise i Pundit koji su korišteni za provjeru autentičnosti i autorizaciju korisnika.

Ključne riječi: bijela krvna zrnca, Devise, leukociti, mrežna aplikacija, Pundit, Ruby on Rails

SUMMARY: WEB APPLICATION FOR CLASSIFYING WHITE BLOOD CELLS

This paper discusses problems encountered while developing a web application for classifying white blood cells. The application has been developed successfully using Ruby version 2.3.1 and Ruby on Rails version 5.0.1. It has been put on a production server where it was additionally tested by users. The paper also describes medical history behind blood and blood cells, as well as technologies used in the development of the application. Additionally, the paper describes and shows the design of the database, user interface and features of the application. The Rails philosophy includes two major principles - "Don’t Repeat Yourself" and "Convention over Configuration", which accelerated the development of the application. Gems, which allowed usage of self-contained solutions to specific problems that are available to all Rails developers, further accelerated the development process. Devise and Pundit are gems which allowed for seamless user authentication and authorization.

Keywords: Devise, leukocytes, Pundit, Ruby on Rails, web application, white blood cells

ŽIVOTOPIS

Kristijan Pavlović rođen je 12. listopada 1995. godine u Frankfurtu na Majni, SR Njemačka. Osnovnoškolsko obrazovanje započinje u Osnovnoj školi Tenja 2002. godine. Nakon završetka osnovne škole 2010. godine, obrazovanje nastavlja u Graditeljsko-geodetskoj školi Osijek koju završava 2014. godine i upisuje preddiplomski studij računarstva na Elektrotehničkom fakultetu Osijek.