|  |  |
| --- | --- |
| Image result for прва гимнаѕија | **ПРВА КРАГУЈЕВАЧКА ГИМНАЗИЈА**  **КРАГУЈЕВАЦ** |

**МАТУРСКИ РАД**

**Тема:( *Веб-апликација за бележење и упоређивање трансакција*)**

Предметни професор: Ученик:

***Томислав Каљевић***  ***Љубомир Матовић,IVsi***

Крагујевац, јун 2020. године

# САДРЖАЈ

[САДРЖАЈ 2](#_heading=h.gjdgxs)

[1.](#_heading=h.30j0zll) УВОД 3

[2.](#_heading=h.1fob9te) ТЕХНОЛОГИЈЕ КОРИШЋЕНЕ ЗА ИЗРАДУ АПЛИКАЦИЈЕ 4

[2.1.](#_heading=h.1t3h5sf) Речник [6](#_heading=h.3znysh7)

[3.](#_heading=h.2et92p0) ФОЛДЕРСКА СТРУКТУРА 7

[4.](#_heading=h.tyjcwt) СТРУКТУРА БАЗЕ ПОДАТАКА 9

[5.](#_heading=h.3dy6vkm) ФУНКЦИОНАЛНОСТИ АПЛИКАЦИЈЕ 11

[5.1.](#_heading=h.1t3h5sf) Администратор 12

[5.2.](#_heading=h.4d34og8) Корисник 14

[5.](#_heading=h.2s8eyo1) ЗАКЉУЧАК 16

[6.](#_heading=h.17dp8vu) ПРИЛОГ 17

[7.](#_heading=h.3rdcrjn) ЛИТЕРАТУРА 18

# УВОД

У савременом добу све више пажње се посвећује информационим технологијама. Сада више није потребно да се чека у реду у пошти или банци да би се платио рачун; или да се оде у продавницу да би се нешто купило; или забележило нешто у некој свесци или нотесу, већ све то можемо урадити на неком од паметних уређаја коришћењем апликација. Апликација, која ће бити описана у наставку, кориснику омогућава бележење и статистички преглед финансијских трансакција.

Апликација је развијена за коришћење на Вебу. Разликују се две врсте корисника: администратор и корисник. Свакој групи корисника је омогућено да управља различитим ентитетима у складу са улогом у систему која се додељује приликом пријављивања. Релевантни подаци чувају се у бази података, тако да је могуће њихово поновно коришћење у циљу приказивања информација или статистичких анализа.

# Технологије коришћене за израду апликације

**HTML** (Hyper Text Markup Language, језик за означавање хипертекста) је описни језик специјално намењен опису веб страница. Помоћу њега се једноставно могу одвојити елементи као што су наслови, параграфи, цитати и слично. Поред тога, у HTML стандард су уграђени елементи који детаљније описују сам документ као што су кратак опис документа, кључне речи, подаци о аутору и слично. Ови подаци су општепознати као мета подаци и јасно су одвојени од садржаја документа.

**CSS** (Cascading Style Sheets) је језик форматирања помоћу ког се дефинише изглед елемената веб-странице. Првобитно, HTML је служио да дефинише комплетан изглед, структуру и садржај веб-странице, али је од верзије 4.0 HTML-а уведен CSS који би дефинисао конкретан изглед, док је HTML остао у функцији дефинисања структуре и садржаја.

**JS** (енгл. JavaScript) је динамичан, слабо типизиран и интерпретиран програмски језик високог нивоа. Стандардизован је по ЕКМАСкрипт спецификацији језика. Поред HTML-а и CSS-а, Јаваскрипт је једна од три водеће технологије за дефинисање садржаја на Вебу; већина веб-сајтова користи Јаваскрипт а сви модерни веб-читачи га подржавају без потребе за инсталирањем додатака. Комбинован са HTML језиком и CSS-om Јаваскрипт чини DHTML (Dynamic HTML). Јаваскрипт је језик заснован на прототиповима са функцијама првог реда, што га чини језиком вишеструке парадигме који подржава објектно-оријентисани, императивни и функционални начин програмирања. Садржи АПИ за рад са текстом, низовима, датумима и регуларним изразима, али не и улазно/излазне функционалности, као што су повезивање, складиштење података или графичке функционалности, за шта се ослања на окружење у коме се извршава. Иако се на први поглед може рећи да Јаваскрипт има одређене сличности са програмским језиком Јава, као што су назив, синтакса и стандардне библиотеке, они су два потпуно различита програмска језика која се значајно разликују у свом дизајну.

**PHP** (Hypertext Preprocessor) специјализовани је скриптни језик првенствено намењен за израду динамичног веб садржаја и изводи се на страни сервера. Стекао је популарност због своје једноставности и синтаксе наслеђене из програмског језика C. Током времена језик се проширивао и стицао могућности за објектно оријентисано програмирање, нарочито од верзије 5.0. Наликује језику C++ у смислу што дозвољава и чисто-процедурално програмирање, али истовремено омогућава и коришћење класа и других концепата објектно оријентисаног програмирања (наслеђивање, апстрактне класе и методе, интерфејсе итд.).

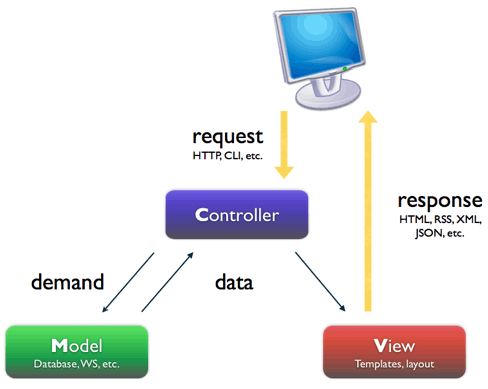
**AJAX** (Asychronous Javascript and XML), данас је више у употреби JSON него XML, је група технологија намењена за динамичко креирање Веб странице. Идеја је да се Веб страница учита само једном и да се свака даља комуникација са сервером одвија асинхроно, без блокирања интерфејса и без поновног учитавања целе стране.

**MySQL** је вишенитни вишекориснички SQL систем за управљање базама података. Систем ради као сервер, обезбјеђујући вишекориснички интерфејс за приступ бази података.

**JQuery** је вишеплатмфорска JavaScript библиотека дизајнирана да поједностави скриптовање са корисничке стране језика HTML**.**

**Bootstrap** представља бесплатни веб фрејмворк отвореног кода, за креирање веб сајтова и веб апликација. Базиран је на HTML и CSS шаблонима за типографију, креирању формулара, дугмади, навигационим и осталим компонентама интерфејса, као и опционим JavaScript додацима. Циљ Bootstrap фрејмворка (енгл. framework) је олакшавање програмирања за веб.

**MVC(Model-view-controller)** архитектура је шаблон који се обично користи за развој софтерских решења заснованог на трослојној подели целокупног решења. Почива на идеји о поновној употреби већ постојећег софтверског кода, олакшавању развоја и каснијем одржавању апликационог софтвера методом раздвајања на посебне компоненте: модел, приказ података (view) и контролeр (управљач), при чему је компонента за приказ информација одвојена од интеракције корисника са тим информацијама. Приказ података представља HTML страну. Модел представља базу података. Контролор ради сву логику и обавља комуникацију између приказа података и модела.



**слика 1- приказ MVC** **архитектуре**

## Речник

**CRUD**(Create-Read-Update-Delete) је акроним насатао од енглеских речи Create, Read, Update и Delete. Подразумева креирање, читање једног или свих, измену и брисање одређеног ентитета над којим се извршава.

**Библиотека** представља колекцију непроменљивих ресурса коју користе програмски језици, нпр. за развој софтвера. Библиотеке могу да садрже податке за подешавање, документацију, помоћне податке, поруке у шаблонима, старе кодове и потпрограме, класе, вредности или типске спецификације.

**OOP(Објектно-оријентисано програмирање)** је парадигма програмирања, која користи објекте као основу за пројектовање рачунарских програма и различитих апликација софтвера. Заснива се на различитим техникама, као што су наслеђивање, модуларност, полиморфизам и енкапсулација.

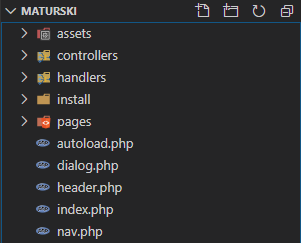
**Framework** је софтверска платформа, радни оквир који програмере приморава да пишу код по тачно одређеним правилима.

**Сторне процедуре** су процедуре које у свом телу садрже **SQL** упите и налазе се у бази података.

**Email** јеназив за мрежни сервис који омогућава слање и примање порука разноврсног садржаја користећи електронске уређаје.

# Фолдерска структура

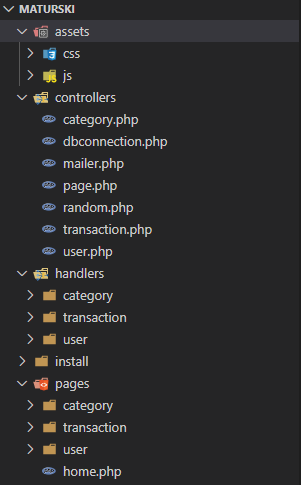
Пројекат се састоји од већег броја директоријума организованим на основу природе њиховог садржаја.



**слика 2-структура датотека матурског рада**

„Глобалне“ датотеке се налазе у кореном директоријуму. Остале датотеке су подељени по директоријумима:

* assets- JS i CSS датотеке
* controllers- класе које су контролери; садрже већи део логике
* handlers- датотеке које обрађују форме, GET и POST захтеве; служе за инстанцирање контролера и позив њихових метода
* install- .sql датотеке које представљају реплику базе података радне верзије пројекта и користе се приликом инсталирања апликације на друге сервере
* pages- странице које крајњи корисник види; углавном садрже HTML код



**слика 3-подела на потдиректоријуме**

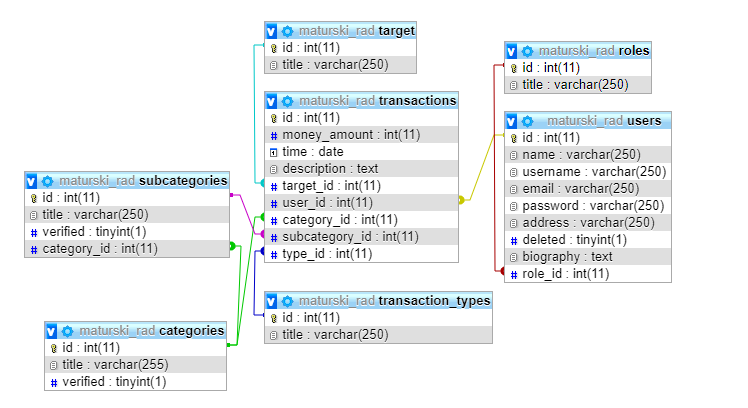
Унутар директоријума pages и handlers постоји подела по функционалностима и то - на потдиректоријуме category, transaction и user, који су главни ентитети (више о њима у наставку).

# Структура базе података



**слика 4-структура базе података**

База података се састоји од релационих табела и сторних процедура.



**слика 5- приказ табела и веза**

Табела **roles** чува податке о улогама које неки корисник може да има (нпр. може да буде обичан корисник или администратор).

Табела **users** чува податке о корисницима (име, username, email, лозинку, адресу, да ли је активан или не, биографију коју може да има али не мора и улогу коју има).

Табела **transaction\_types** чува податке о типовима трансакција(нпр. ПРИЛИВ и ОДЛИВ).

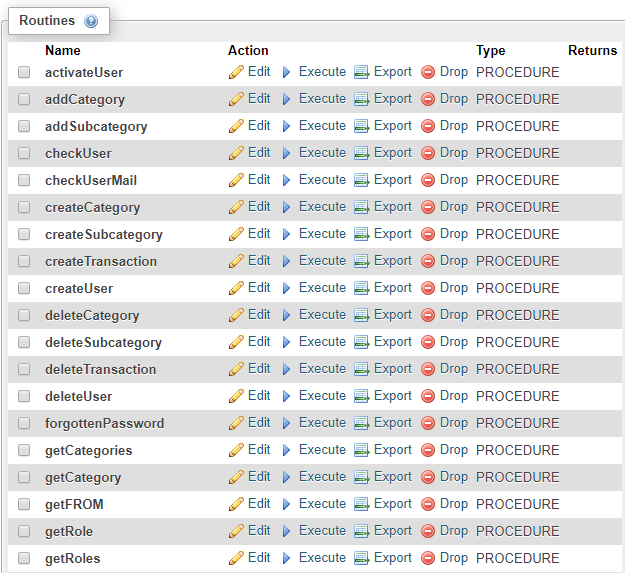
Табела **categories** чува податке о категоријама (њен назив и да ли је одобрена) .

Табела **subcategories** чува податке о поткатегоријама (њен назив, да ли је одобрена, којој категорији припада).

Табела **target** чува податке о особи или компанији којој је уплаћен новац или која уплаћује новац.

Табела **transactions** чува податке о трансакцијама (која сума је у питању, датум када је обављена трансакција, опис трансакције који је опционалан, коме/ко је уплатио/је уплаћен новац, који корисник је забележио трансакцију, којој категорији припада, којој поткатегорију унутар те категорије припада и ког је типа).

Осим наведених табела база података садржи и сторне процедуре (слика 6). Користе се да би се MySQL упити раздвојили од PHP кода.

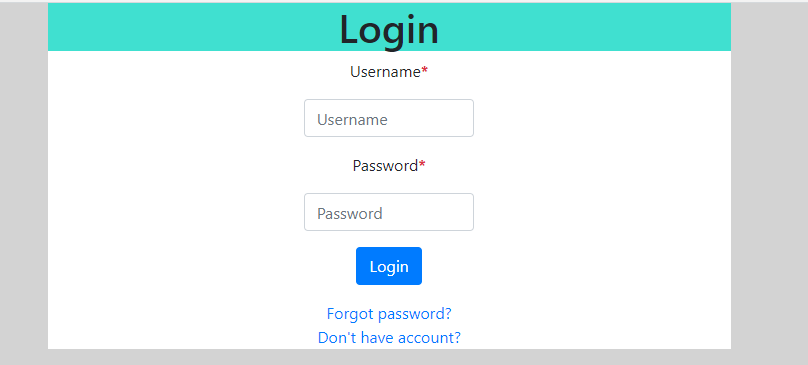


**слика 6- сторне процедуре**

Неке од најзначајнијих сторних процедура су: activateUser, checkUser, checkUserMail, createUser, ... Њихова имена јасно указују на функцију коју врше.

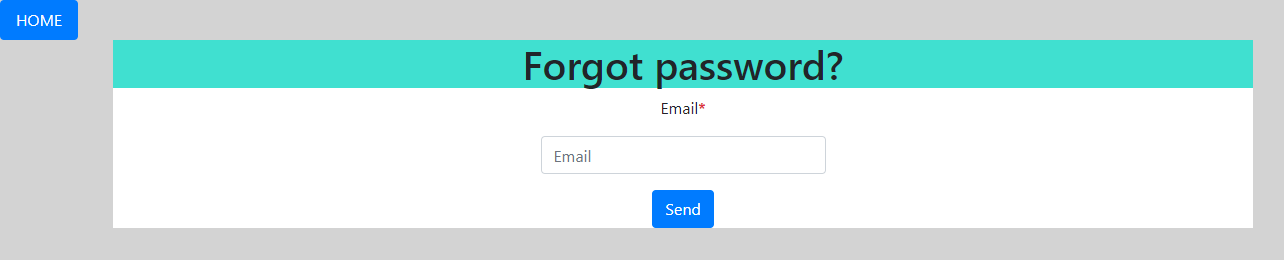
# Функционалности апликације

Приликом учитавања апликације кориснику се омогућава пријава путем форме приказане на слици бр. 7.



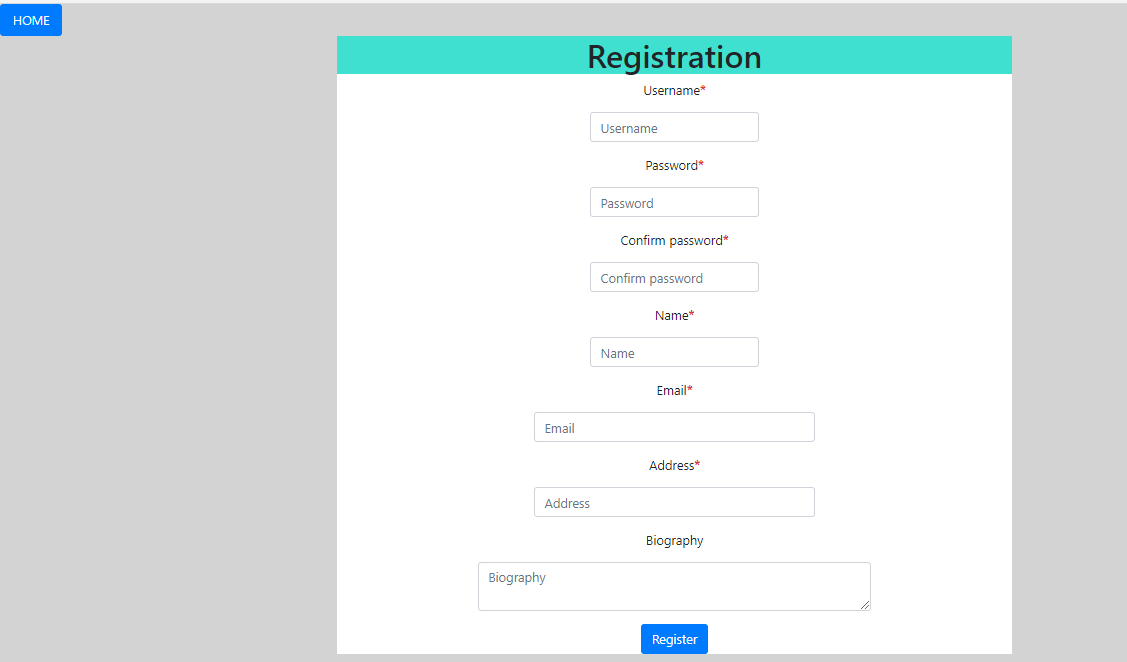
**слика 7-логовање**

Постоје два типа корисника: администратор и корисник. Ако je корисник заборавио лозинку може захтевати поновно креирање лозинке путем форме приказане на слици бр. 8. На еmail ће му бити послата нова лозинка.



**слика 8-заборављена лозинка**

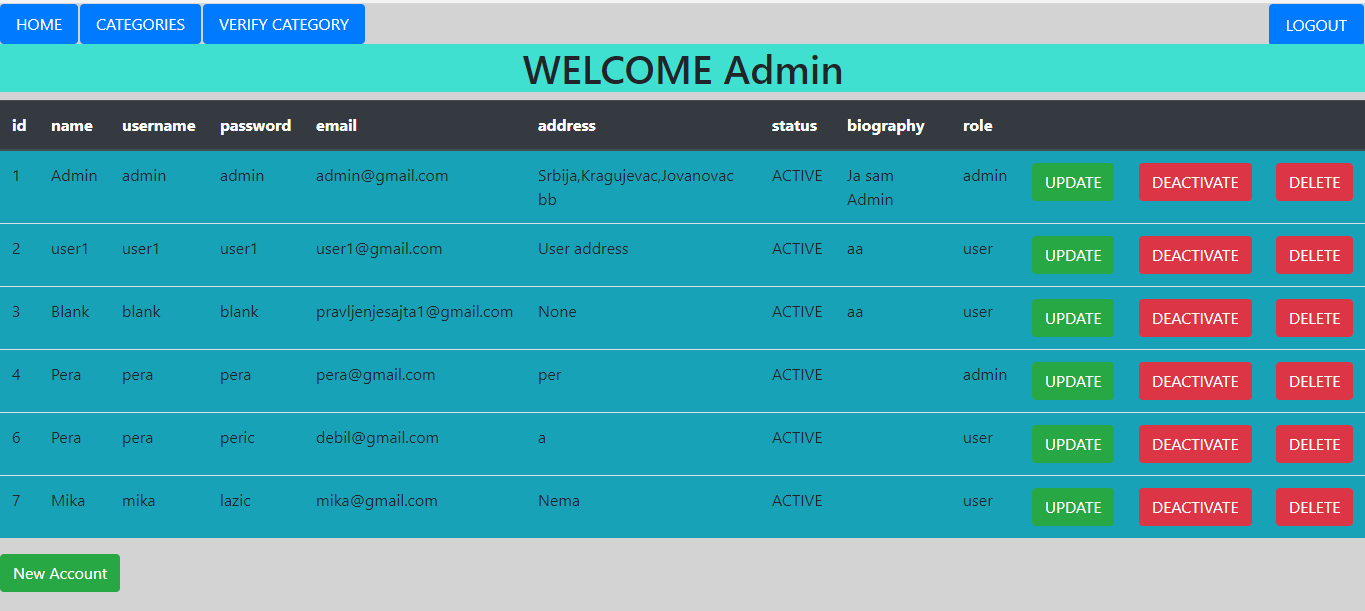
Корисници се могу регистровати. Приликом регистрације користи се форма приказана на слици бр. 9.



**слика 9-регистрација**

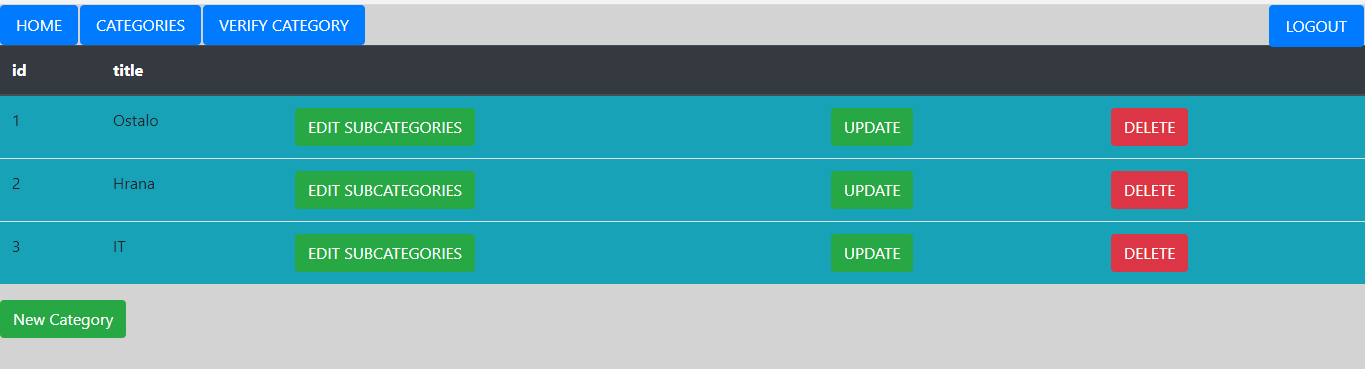
## Администратор

Након пријаве, администраторима ће бити приказан интерфејс као на слици бр. 10.

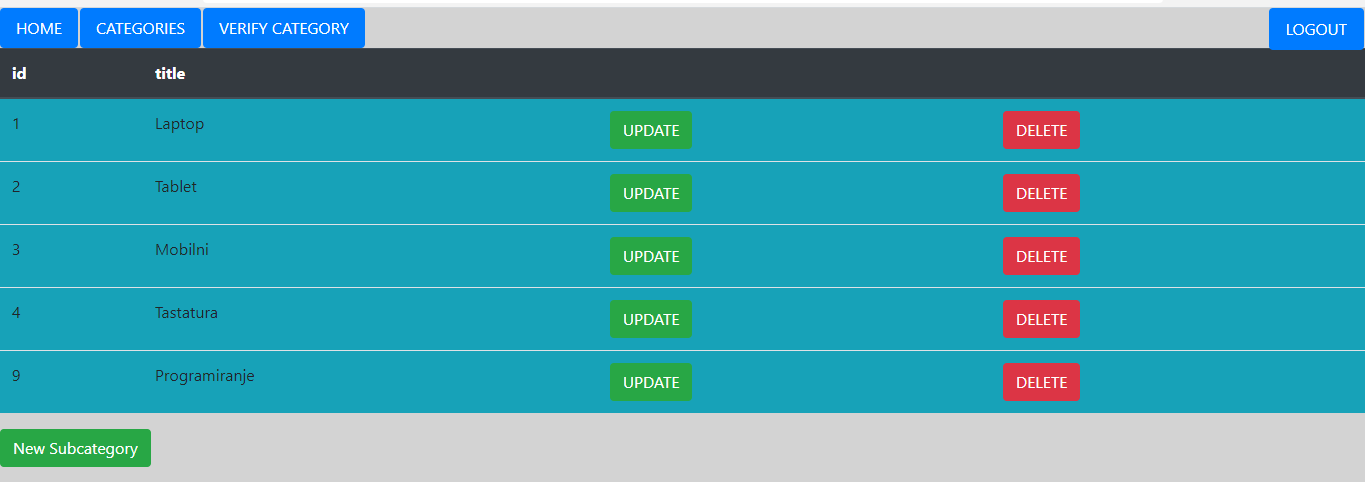


**слика 10-преглед свих корисника**

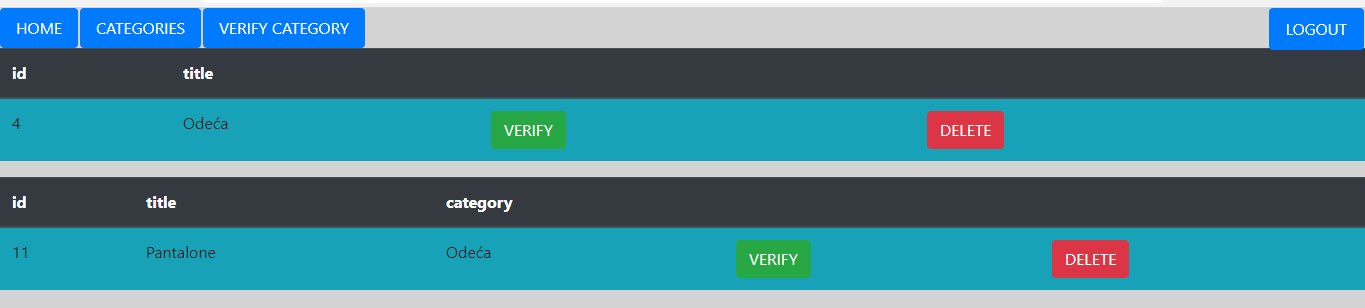
Администратор има CRUD над свим корисницима, категоријама и њиховим поткатегоријама. Може да одобрава и брише категорије и поткатегорије. У наставку су приказани интерфејси који су доступни администратору.



**слика 11-преглед свих категорија**



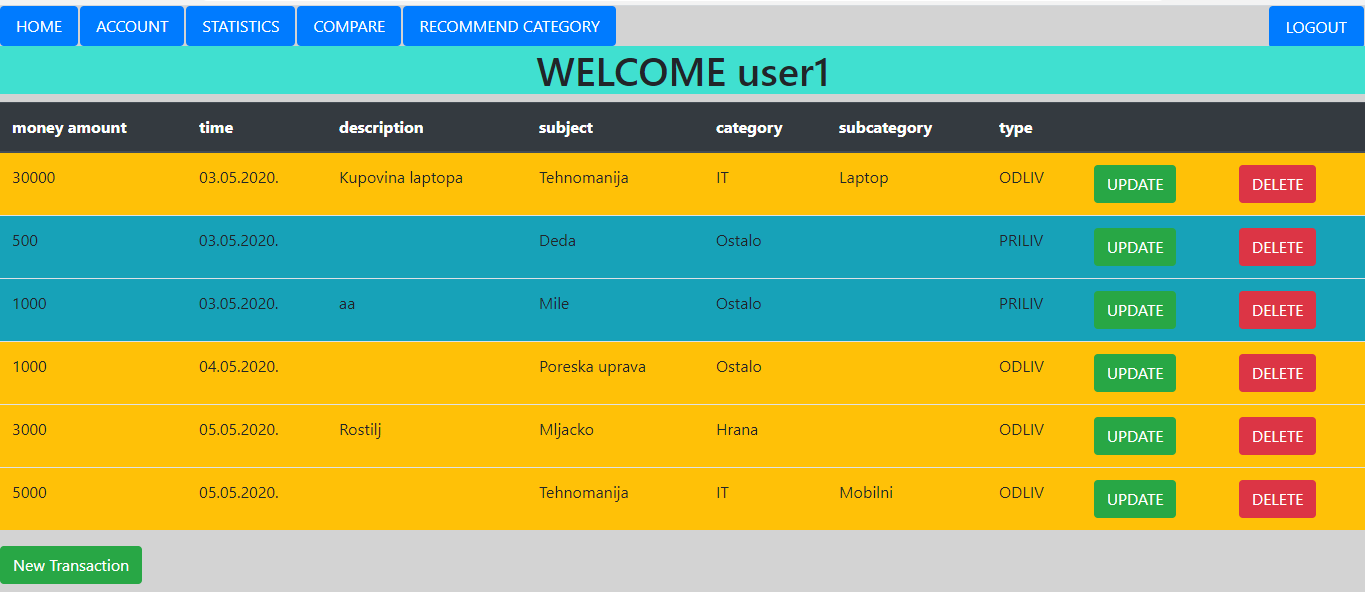
**слика 12-преглед свих поткатегорија категорије ИТ**



**слика 13-одобравање категорија и поткатегорија**

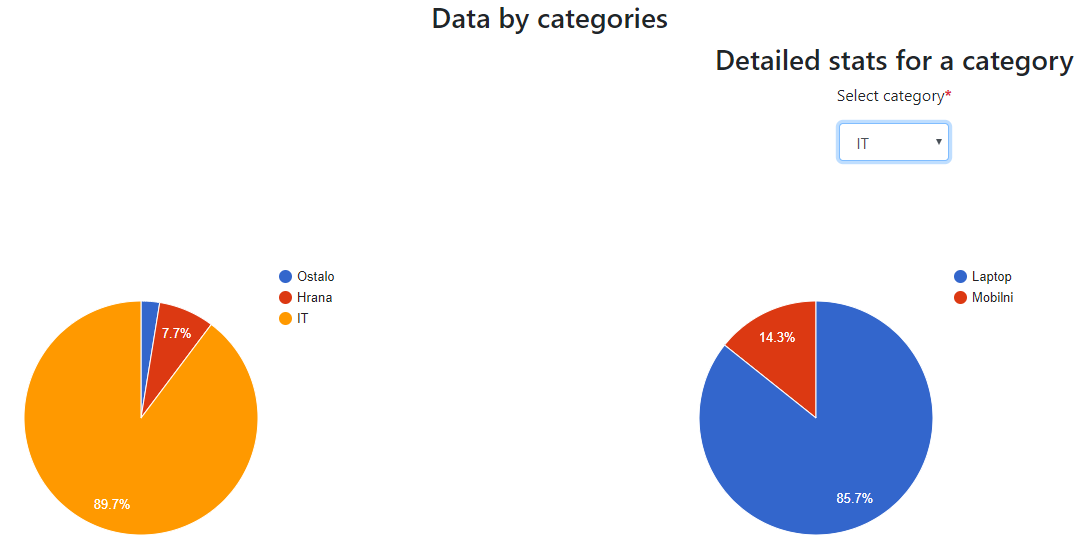
## Корисник

Након пријаве, корисницима ће бити приказан интерфејс као на слици бр. 14.

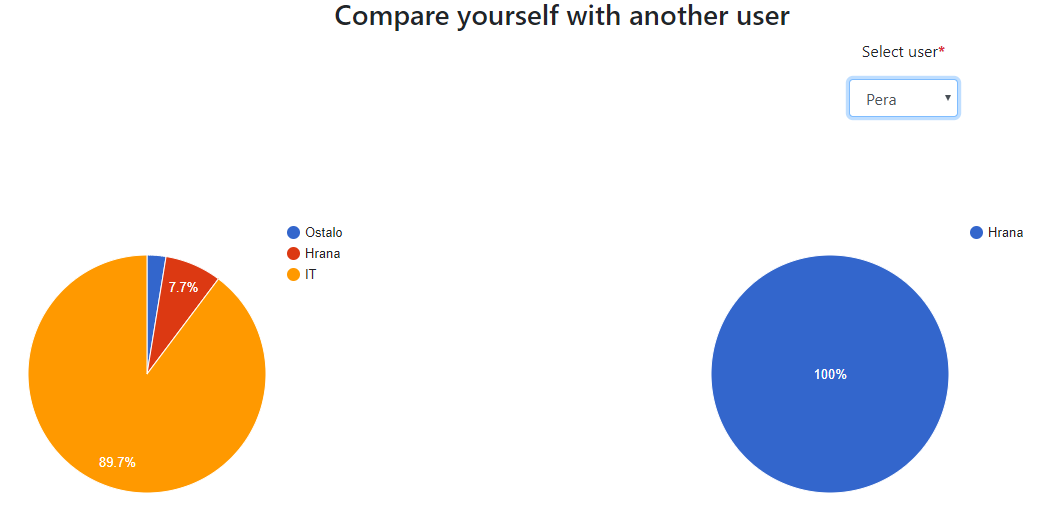


**слика 14-преглед свих трансакција корисника user1**

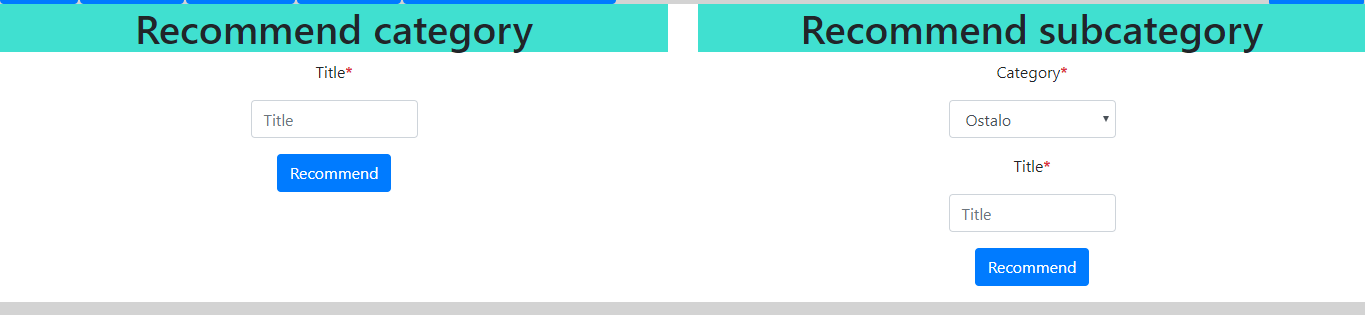
Корисник има CRUD над својим трансакцијама. Може да ажурира свој налог. Има увид у своју потрошњу и потрошњу по поткатегоријама. Може да упореди своју потрошњу са потрошњом другог корисника. Може да предлаже категорије и поткатегорије. У наставку су приказани интерфејси који су доступни кориснику.



**слика 15-статистика улогованог корисника**



**слика 16-упоређивање потрошње улогованог корисника са другим корисником**



**слика 17-предлагање категорија и поткатегорија**

# ЗАКЉУЧАК

Коришћењем веб-апликације за бележење и упоређивање трансакција остварују се вишеструки бенефити. Неки од њих су: прецизност информација, смањена могућност грешке, визуализација података, доступност на већем броју уређаја. Треба напоменути и да су подаци трајно сачувани у бази података па је могуће њихово поновно коришћење као и извоз. На овај начин кориснику се пружа бржи и ефикаснији начин да контролише своје финансијске трансакције, у односу на традиционални приступ без коришћења савремених технологија.

# ПРИЛОГ

1. <https://srdjanmaravic.files.wordpress.com/2014/08/mvc.png?w=584>

# ЛИТЕРАТУРА

1. <https://sr.wikipedia.org/sr/MVC_arhitektura>
2. <https://sr.wikipedia.org/sr-ec/HTML>
3. <https://sr.wikipedia.org/sr-ec/%D0%88%D0%B0%D0%B2%D0%B0%D1%81%D0%BA%D1%80%D0%B8%D0%BF%D1%82>
4. <https://sr.wikipedia.org/sr-ec/PHP>
5. <https://www.webprogramiranje.org/ajax-sa-plain-javascript-om/>
6. <https://sr.wikipedia.org/sr-ec/%D0%91%D0%B0%D0%B7%D0%B0_%D0%BF%D0%BE%D0%B4%D0%B0%D1%82%D0%B0%D0%BA%D0%B0>
7. <https://sr.wikipedia.org/sr-ec/%D0%9E%D0%B1%D1%98%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D1%98%D0%B5%D0%BD%D1%82%D0%B8%D1%81%D0%B0%D0%BD%D0%BE_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%B8%D1%80%D0%B0%D1%9A%D0%B5>
8. <http://sr.magma.rs/blog/recnik/framework>
9. <https://sr.wikipedia.org/sr-ec/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_(%D1%80%D0%B0%D1%87%D1%83%D0%BD%D0%B0%D1%80%D1%81%D1%82%D0%B2%D0%BE)>
10. <https://sr.wikipedia.org/wiki/%D0%91%D1%83%D1%82%D1%81%D1%82%D1%80%D0%B5%D0%BF_(%D1%84%D1%80%D0%BE%D0%BD%D1%82-%D0%B5%D0%BD%D0%B4_%D1%84%D1%80%D0%B5%D1%98%D0%BC%D0%B2%D0%BE%D1%80%D0%BA)>
11. <https://sr.wikipedia.org/wiki/JQuery>
12. <https://sr.wikipedia.org/sr-ec/MySQL>

Датум предаје рада:

Питања /Коментар:



Комисија:

Председник 

Испитивач

Члан

Оцена ( )

Датум одбране: