



Univerzitet Singidunum

Tehnički fakultet

Online platforma za prodaju video igara

-Projektni rad-

Predmet:

Web programiranje

Profesor:

doc. dr Nebojša Bačanić Džakula

Predmetni asistent:

mast. Dušan Marković

Student:

Stefan Dimitrijević 2018/200497

Beograd, 2020. godine

Sadržaj

1. UVOD.....	3
2. Opis aplikacije.....	4
2.1 Početna strana.....	5
2.2 Registracija.....	5
2.3 Stranica profil korisnika.....	7
2.4 Stranica Store.....	8
2.5 Stranica „Library“.....	9
2.6 Brisanje naloga.....	11
Zaključak.....	12

1. UVOD

Svrha ovog dokumenta je da opiše i detaljno prikaže projektni rad iz predmeta "Web programiranje", tematika ovog projekta jeste izrada troslojne veb aplikacije, odnosno izrada prezentacionog sloja, sloja poslovne logike i sloja baze podataka. Dokumentacija će obuhvatiti vizuelni prikaz aplikacije, njene karakteristike i funkcionalnosti, tehnologije korišćene za izradu, kao i sam kod korišćen za implementaciju aplikacije.

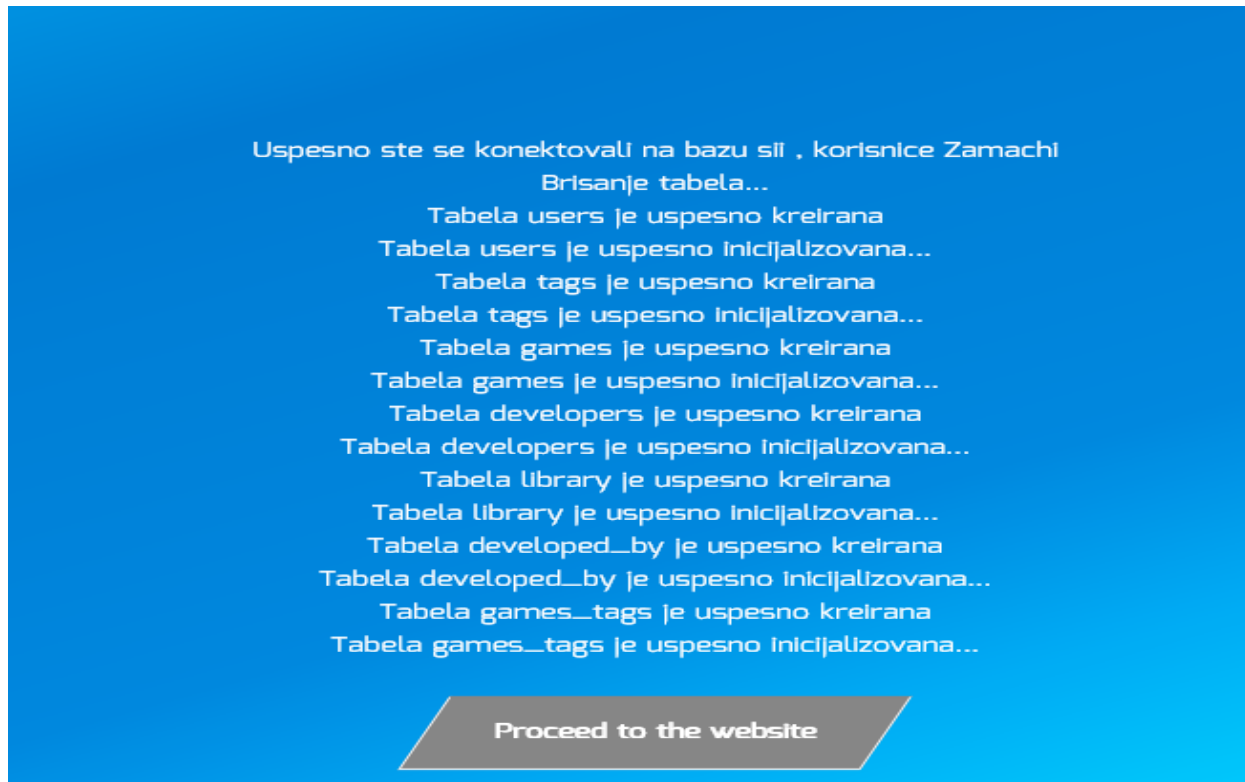
Aplikacija, se fokusira na platformu za prodaju video igara. Treba da omogući korisniku da se registruje i uloguje na sajt pomoću parametara svojeg naloga, doda novac na svoj nalog, da promeni sliku na svom profilu ili lozinku svog profila, da pretražuje video igre na osnovu nekog unosa(npr. Ime ili žanr). Nakon logovanja, korisnik može posetiti prodavnicu, biblioteku video igara koje korisnik poseduje(koja je inicijalno prazna), kao i da sam kupi video igre koje ga interesuju u prodavnici/pretrazi. Korisniku je takođe data mogućnost da obriše svoj nalog.

Aplikacije je rađena u Notepad++ editoru koda i tehnologije koje su korišćene za ovu aplikaciju su:

1. HTML – generisanje statičkog sadržaja
2. CSS – stilizovanje sadržaja
3. PHP – za manipulisanjem unosa korisnika i manipulacijom nad bazom podataka
4. JavaScript – za validaciju formi na strani korisnika i neke dodatne efekte
5. jQuery – za animacije
6. MySQL – za skladištenje podataka, ažuriranje/generisanje sadržaja na aplikaciji

2. Opis aplikacije

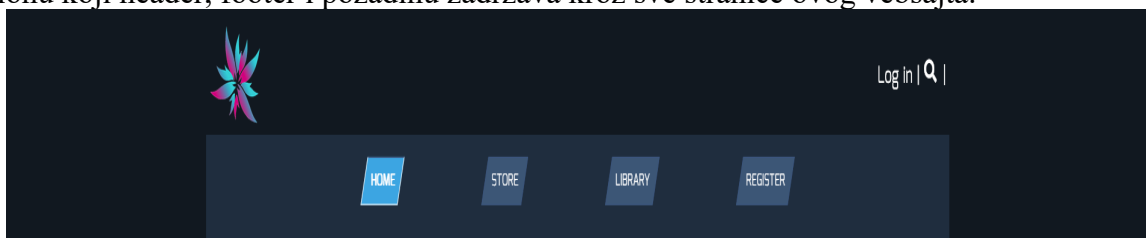
Za potrebe ovog projekta, napravljena je strana loader.php koja briše sve korisničke podatke(slike, tekstualne fajlove...), a zatim sve tabele iz baze podataka, te zatim rekreira sve tabele, zajedno sa statički unetim vrednostima u njoj.



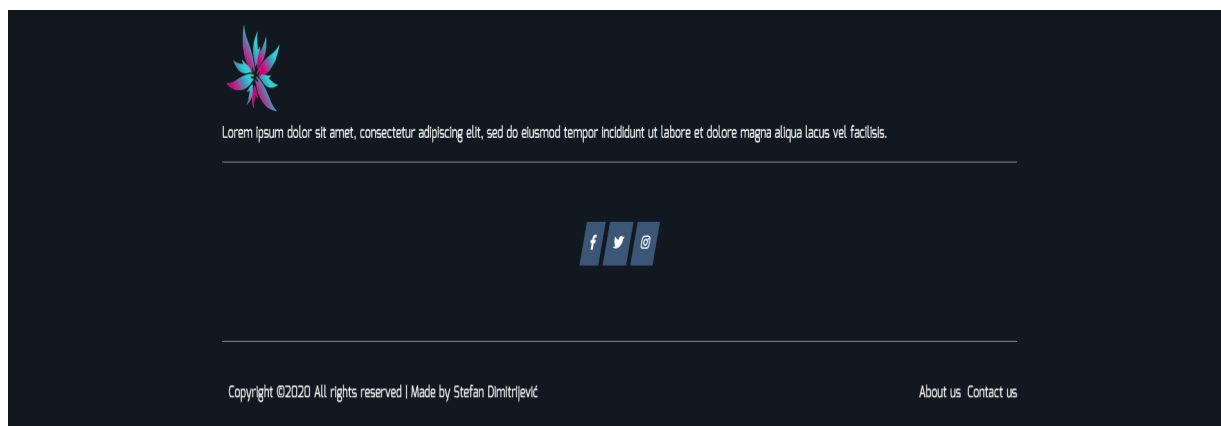
Slika 1. Loader.php stranica koja inicijalizuje bazu

2.1 Početna strana

Veb strane su, radi održavanja konzistentnosti dizajna širom aplikacije, odrađene po šablonu koji header, footer i pozadinu zadržava kroz sve stranice ovog veb sajta.



Slika 2. Header stranice



Slika 3. Footer stranice

2.2 Registracija

Header sadrži logo, dugme koje korisnika preusmerava na stranicu za logovanje, dugme koje otvara pretragu, kao i navigacioni bar koji vodi na sledeće 4 stranice: „home“, prodavnica, biblioteka video igara i stranica za registraciju korisnika. Footer sadrži identičan logo kao i u zaglavlju, linkove ka stranicama društvenih mreža, stranicu „O nama“, kao i stranicu za kontakt.

Da bi korisnik počeo da koristi sajt, mora prvo da se registruje.

Slika 4. registraciona forma

Nakon što korisnik unese sve neophodne parametre(slika je opcionalna) koji zadovolje pravila validacije, napraviće se korisnik u bazi podataka registrovanih korisnika:

```
define("fajloviProjekat","C:/xampp/htdocs/ProjekatLabrupiKolikvijum/img/users/");
$connection=new mysqli("localhost","Zamachi","asdwelhu","sii");

$username=$_POST['username'];
$password=password_hash($_POST['password'], PASSWORD_BCRYPT);
$country=$_POST['country'];
$imageFajla=$_FILES['fileToUpload']['name'];

if(isset($_FILES['fileToUpload'])) {
    if(is_uploaded_file($_FILES['fileToUpload']['tmp_name'])) {
        if(!($_FILES['fileToUpload']['type']=='image/png' || $_FILES['fileToUpload']['type']=='image/jpg' || $_FILES['fileToUpload']['type']=='image/jpeg')) {
            echo "<p style='color: red; font-weight: bold;'> Pogrešan format fajla!</p>";
        } else {
            $sql="
            INSERT INTO users(username, password, country)
            VALUES
            ('$username','$password','$country')
            ";
        }
    } else {
        if(file_exists(fajloviProjekat . $_FILES['fileToUpload']['name'])) {
            $osnovnoIme=basename(fajloviProjekat . $_FILES['fileToUpload']['name'], ".png");
            $ekstenzijaFajla=pathinfo($_FILES['fileToUpload']['name'], PATHINFO_EXTENSION);
            $counter=0;
            $noviNaziv="";
            do {
                $counter++;
                $noviNaziv="".$osnovnoIme."(" . strval($counter) . ").".$ekstenzijaFajla;
            } while(file_exists(fajloviProjekat . strval($noviNaziv)));
            $_FILES['fileToUpload']['name']=$noviNaziv;
        }
        $image=move_uploaded_file($_FILES['fileToUpload']['tmp_name'], fajloviProjekat . $_FILES['fileToUpload']['name']);
    }
    if($image==1) {
        $sql="
        INSERT INTO users(username, password, country, avatar)
        VALUES
        ('$username','$password','$country', '". $connection->real_escape_string('img/users/' . $_FILES['fileToUpload']['name']) . "'
        ";
    }
}
```

Slika 5. Code snippet za kreiranje korisnika

Nakon što korisnik unese pravilne parametre u polja za registraciju, podaci se POST metodom preusmeravaju na stranicu confirm.php, gde se obrađuju; dohvaćaju se korisničko ime, šifra i država korisnika iz superglobalnog `_POST` asocijativnog niza, s tim da se lozinka hešira pri dohvaćanju, u kakvom će se obliku skladištiti u bazi. S obzirom da je slika opcionalna, prvo se proverava da li je uopšte aploudovana, ukoliko jeste, proverava se da li je ispravan format(.png, .jpeg ili .jpg), ukoliko jeste, prolazi se kroz još jednu proveru gde se proverava da li fajl, na onoj lokaciji kojoj ćemo novoaploudovani fajl pomeriti, postoji sa identičnim imenom, ukoliko postoji, aploudovani fajl se preimenuje u fajl(n).ekstenzija, gde će n biti neki prirodni broj počevši od 1 pa dok god se ne pogodi ime fajla koji ne postoji u folderu. Konačno ako fajl prođe kroz sve provere uspešno i uspešno se pomeri u željeni folder, u bazu podataka se upisuju ime korisnika, država, heširana lozinka i relativna putanja ka slici korisnika(ukoliko je slika aploudovana). Korisnik se zatim može ulogovati na svoj profil pristupnim parametrima kojima je zadao i pristupi profilu gde može promeniti lozinku, dodati novu sliku i dodati sumu novca na svoj nalog.

2.3 Stranica profil korisnika

Nakon što se korisnik uspešno uloguje, može svom profilu pristupiti klikom na ikonicu u headeru, gde može ažurirati svoje podatke:

You can update your data on this page

Username
Korisnik123

Account created at:
2020-05-19 21:46:53

Country of origin:
Serbia

Account balance:
150.50

Profile picture placeholder: Browse... No file selected.

Your new password:
[password input]
Confirm new password:
Confirm your new password...
Add currency to your account:
[currency dropdown]
Update your info
Disable account

Slika 6. Profil stranica

```
if(isset($_POST['newPassword1']) && $_POST['newPassword1'] != NULL && $_POST['newPassword1'] != '' && $_POST['newPassword1'] != ' '){
    $newPassword=password_hash($_POST['newPassword1'],PASSWORD_BCRYPT);
    $sqlUpdate="
    UPDATE users
    SET password='".$newPassword."'
    WHERE user_id=".$_SESSION['user_id'].
    ";
    $connection->query($sqlUpdate);
    $_SESSION['password']=$_POST['newPassword1'];
}

if(isset($_POST['uplata']))){
    $accBalance=abs(floatval($_POST['uplata']));
    $sqlUpdate="
    UPDATE users
    SET account_balance=account_balance+".(number_format($accBalance, 2, ".", ""))."
    WHERE user_id=".$_SESSION['user_id'].
    ";
    $connection->query($sqlUpdate);

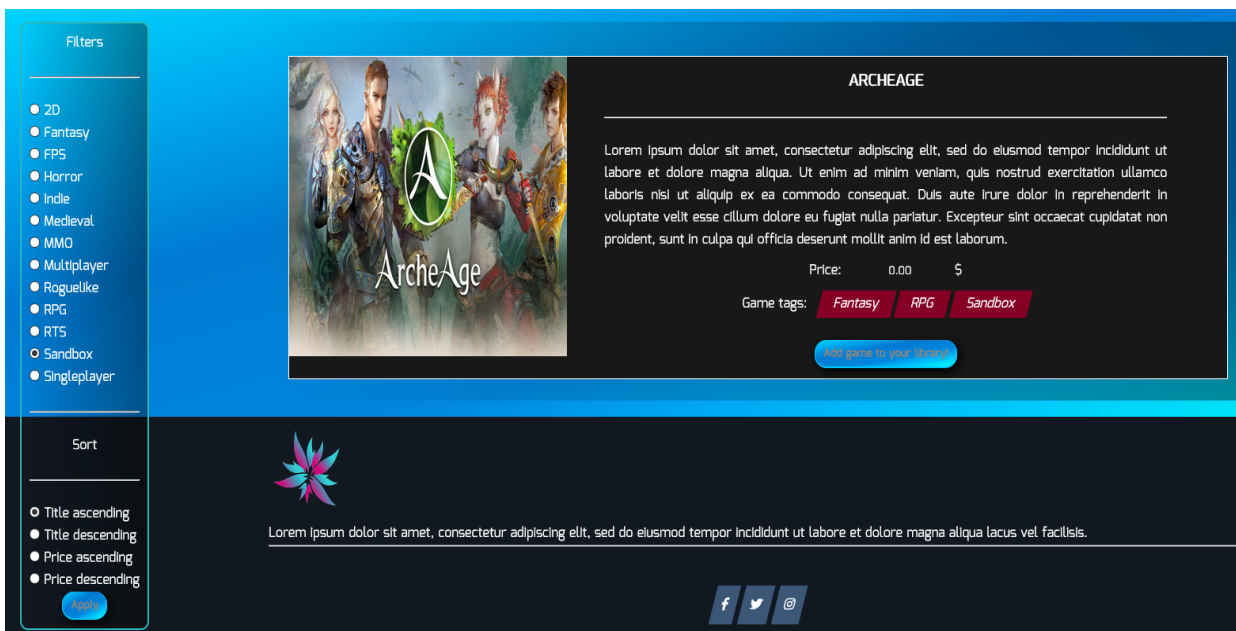
    $sql="
    SELECT account_balance
    FROM
    | users
    WHERE
    | users.user_id = '".$_SESSION['user_id']."'
    ";
    $resultat=$connection->query($sql)->fetch_assoc();
    $_SESSION['accBalance']=$resultat['account_balance'];
    $_SESSION['account_balance']=$resultat['account_balance'];
}
```

Slika 7. Logika iza ažuriranja podataka

Pre svega se vrši provera atributa, tj. da li su setovani, za lozinku potrebno je da se nova lozinka podudara u oba polja, tek nakon toga vrši se UPDATE SQL naredba nad bazom za dati ID korisnika koji je uspostavio sesiju. Takođe, ukoliko je setovana količina novca za dodavanje na nalog, potrebno ju je formatirati kao float vrednost i uzeti njenu absolutnu vrednost(u slučaju da je korisnik nekako uneo negativnu vrednost). Pre unosa se ovaj broj formatira number_format() metodom; zaokružuje na 2 decimalna mesta i da koristi tačku, umesto zareza kao separator za decimalni deo, ovo je potrebno radi formata polja account_balance koje je definisano u bazi podataka. Ukoliko je korisnik promenio sliku, ona se takođe ažurira, na isti način kako bi se to učinilo pri registraciji, samo što se ovog puta vrši UPDATE umesto INSERT. Na ovoj strani korisnik takođe može videti svoje podatke koji se iščitavaju iz .txt. fajla koji se generiše iznova svaki put kad se korisnik ponovo uloguje.

2.4 Stranica Store

Stranica Store će korisniku, ukoliko je ulogovan, prikazati spisak svih igara koje on nema u svojoj biblioteci igara(inicijalno 0). Takođe, ukoliko korisnik želi, može proizvode sortirati u opadajućem/rastućem redosledu po imenu, a takođe može ih i isfiltrirati prema određenom tagu, ukoliko sortiramo po naslov igara po alfabetskom redosledu i štikliramo „Sandbox“ kao tag, dobićemo jedan rezultat.



Slika 8. Prodavnica valute

Ukoliko korisnik ima dovoljno sredstava na svom „računu“, onda će klikom na „Add this game to your library,“ (u slučaju besplatnih igara) ili „Buy now,“ dodati tu igru u svoju biblioteku i tamo može videti statistiku vezanu za nju. Ukoliko korisnik pak nema dovoljno sredstava, izaći će mu obaveštenje o tome i biće mu ponuđena opcija da bude preusmeren na svoj profil i doda dovoljnu sumu novca.


```

$sql="
SELECT games.game_id, games.title, games.price, games.image, games.file, games.description
FROM games LEFT JOIN library USING(game_id) INNER JOIN games_tags ON games_tags.game_id = games.game_id INNER JOIN tags ON tags.tag_id = games_tags.tag_id
WHERE
(library.user_id IS NULL OR library.game_id NOT IN(SELECT library.game_id FROM
`users` INNER JOIN library ON `users`.user_id = library.user_id INNER JOIN games ON library.game_id = games.game_id
WHERE
library.user_id=".$SESSION['user_id']." )) ".((isset($_POST['filter'])) ? ("AND UPPER(tags.tag_name) = UPPER(\"".$_POST['filter']."\"))"."
GROUP BY
title
ORDER BY ".((isset($_POST['sortiraj'])) ? $_POST['sortiraj']:"library.game_id ASC")."";


```

Slika 9. SQL upit za dohvaćanje igara koje korisnik ne poseduje i sortiranje/filtriranje

SQL upit koji radi na stranici prodavnice funkcioniše pomoću više JOIN mehanizama, kao prva tabela uzima se tabela games(gde se nalaze sve igre koje su na aplikaciji) i vrši se left join sa tabelom biblioteka preko polja game_id, ovo omogućava da se igre, koje korisnik poseduje, ne pojavljuju na store stranici. Dodatno, za potrebe filtracije i sortiranja, vrše se INNER join-ovi sa tabelama games_tags i tags. U WHERE klauzuli, filtriraju se rezultati iz join-ova; uzimaju se samo oni u obzir gde je ID korisnika NULL(potreban uslov zbog realizacije LEFT JOIN-a u MySQL-u) ili oni rezultati kod kojih game_id ne pripada skupu vrednosti koje vrati podupit. Takođe, ukoliko je filter podešen na strani, vrši se i provera po imenu taga, dodatno, ukoliko je odabran način sortiranja, sortira se po određenom kriterijumu, u protivnom se po defaultu sortira prema game_id-u u rastućem redosledu.

2.5 Stranica „Library“

Na stranici lične biblioteke korisnika korisnik, ukoliko je ulogovan, može videti listu igara koje poseduje, kao i neku dodatnu statistiku; koliko je koju igrao, koliko je vremena prosečno proveo po igri, koja igra mu je najigranija, koja najmanje, koliko igara ima.



ARCHEAGE

Lorem Ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Game tags: Fantasy RPG Sandbox

You've played this game for 151.37 hours.

You have 4 games in your library

Average game time: 321.892500

The most played game: Lineage II , played for: 704.83 hours.

The least played game: ArcheAge , played for: 151.37 hours.

Slika 10. Rezultat obrade forme

```

$sql="
SELECT users.username, games.title, library.hours_played, games.image, games.file, games.description, library.game_id
FROM
`users` INNER JOIN library ON `users`.user_id = library.user_id INNER JOIN games ON library.game_id = games.game_id
WHERE
library.user_id=".$SESSION['user_id']."
ORDER BY
library.hours_played DESC;
";

$resultat=$connection->query($sql) or die("Dosló je do greske pri selektu");

while($red=$resultat->fetch_assoc()){
    echo "<div class='gameRow'>";
        echo "<div class='gameImage'><a href=".$red['file']. "></a></div>";

        echo "<div class='gameInfo'>";
            echo "<a href=".$red['file']. ">";
            echo "<h3 class='gameTitle'>".$red['title']. "</h3>";
            echo "</a>";

            echo "<br><br>";

            echo "<div class='gameDescription'>";
                echo "<p>".$red['description']. "</p>";
            echo "</div>";

            echo "<div class='gamePricing'>";
                echo "<div>";
                //echo "<div class='pricing'>Price: ".($red['price']==0.00 ? 'FREE TO PLAY':(strval($red['price'])).'€'). "</div>";
                $sqlTags="SELECT tag_name
FROM
games INNER JOIN games_tags ON games.game_id=games_tags.game_id INNER JOIN tags ON games_tags.tag_id=tags.tag_id
WHERE
games.game_id=".$red['game_id']. " ";
                $tagovi=$connection->query($sqlTags);

                echo "<br><div>Game tags: &nbsp;<";
                while($tag=$tagovi->fetch_assoc()){
                    echo "<span class='gameTag'>".$tag['tag_name']. "</span>";
                }
                echo "</div><br>";

                echo "</div>";
                echo "<br><p>You've played this game for ".$red['hours_played']. " hours.</p><br>";
            echo "</div>";

        echo "</div>";
    }
}

```

Slika 11. Kod za biblioteku korisnika

SQL naredba vrši projekciju i koristi kolone za korisničko ime, naslov, broj sati odigran, putanju ka slici i fajlu igre na lokalnoj mašini i opis igre za svaki zapis. Tabele users, library i games spajaju se INNER JOIN mehanizmom da bi se pronašao presek te 3 tabele, dok se u obzir uzimaju samo zapisi koji u tabeli library za user_id imaju vrednost koju ima trenutno ulogovani korisnik. Odnosno, dohvaćće samo igre koje se nalaze u biblioteci specifičnog korisnika. Takođe se za svaku igru izvršava upit koji dohvata njene tagove: povezuje tabele games, games_tags i tags INNER JOIN mehanizmom i selektuje rezultate samo koji imaju game_id tekuće igre koja se obrađuje u while petlji.

2.6 Brisanje naloga

Korisnik može u bilo kom trenutku otići na svoju profilnu stranicu i pretisnuti „Disable account,,. Ovo dugme vrši redirekciju na stranicu koja je zadužena za brisanje naloga, koja će, pri uspešnom brisanju korisnika, vršiti ponovnu redirekciju na stranicu za logovanje.

```
<?php
    $connection=new mysqli("localhost","Zamachi","asdwclhu","sii");
    session_start();

    $sql="
    DELETE FROM library
    WHERE user_id=".$_SESSION['user_id']."
    ";

    $connection->query($sql);

    $sql="
    DELETE FROM users
    WHERE user_id=".$_SESSION['user_id']."
    ";

    $connection->query($sql);

    $_SESSION=[];
    session_destroy();

    $connection->close();
    header("Location: index.php");
    exit();
?>
```

Slika 12. Brisanje korisnika

Pre svega se uspostavlja konekcija sa bazom, zatim se izvršava upit koji briše sve zapise vezane za datog korisnika iz tabele library(na osnovu njegovog korisničkog ID-a koji se čuva u superglobalnom nizu sesija). Nakon toga, zasigurnim brisanjem svih stranih ključeva koji pokazuju na primarni ključ u tabeli users(user_id), tek tada moguće je obrisati samog korisnika i njegove podatke iz tabele. Zatim se resetuje superglobalni niz sesija na početno stanje(prazan) i uništava se trenutna sesija sa korisnikom, nakon čega se vrši redirekcija na početnu stranu Web aplikacije.

Zaključak

Zahvaljujući troslojnoj arhitekturi aplikacije, omogućeno je napraviti konkretnu i funkcionalnu aplikaciju koja, iako nije odrađena i funkcionalna 100%, približno omogućuje simulaciju poslovanja jedne ovakve platforme, gde je korisniku omogućeno da kreira nalog, uloguje se, menja svoje parametre, uplaćuje sebi novac na „račun“ i koristi taj novac za kupovinu unutar prodavnice, nakon čega može otići u biblioteku igara koje poseduje i pogleda svoju statistiku. Naravno, uplaćivanje novca na račun korisnika bi se u realnosti odigralo preko nekog servisa banke, jer je reč o pravom novcu. Takođe, korisnik na trenutnom stanju aplikacije ne može da preuzme nikakvu video igru ili je igra na lokalnom računaru, za ovo su potrebne dodatne funkcionalnosti npr. ključ za skladištenje svih tih igara, dostignuća korisnika. Dodatno bi se aplikacija mogla ispolirati nekim sistemom rangiranja igara na osnovu ocena korisnika, mogućnosti da korisnici ostavljaju svoje komentare, sistem refundacije i slično.