Web aplikacija za recenziju i ocjenjivanje video sadržaja

Hećimović, Stjepan

Undergraduate thesis / Završni rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Algebra University College / Visoko učilište Algebra**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:225:148861

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-20**

ALGEBRA VISOKO UČILIŠTE

Repository / Repozitorij:

Algebra Univerity College - Repository of Algebra Univerity College

zir.nsk.hr

dabar DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

VISOKO UČILIŠTE ALGEBRA

ZAVRŠNI RAD

Web aplikacija za recenziju i ocjenjivanje video sadržaja

Stjepan Hećimović

Zagreb, Lipanj 2019 .

Student vlastoručno potpisuje Završni rad na prvoj stranici ispred Predgovora s datumom i oznakom mjesta završetka rada te naznakom:

„Pod punom odgovornošću pismeno potvrđujem da je ovo moj autorski rad čiji niti jedan dio nije nastao kopiranjem ili plagiranjem tuđeg sadržaja. Prilikom izrade rada koristio sam tuđe materijale navedene u popisu literature, ali nisam kopirao niti jedan njihov dio, osim citata za koje sam naveo autora i izvor, te ih jasno označio znakovima navodnika. U slučaju da se u bilo kojem trenutku dokaže suprotno, spreman sam snositi sve posljedice uključivo i poništenje javne isprave stečene dijelom i na temelju ovoga rada“.

U Zagrebu, 21.9.2019.

Predgovor

Želim se zahvaliti profesoru i mentoru Ivanu Božajiću na suradnji u izradi završnog rada. Mentor mi je uvelike pomogao pri razumijevanju glavnih značajki dinamičke web aplikacije i što bi ona trebala sadržavati. Također, želim se zahvaliti i Visokom učilištu Algebra koje me je kroz praktičan i teoretski rad naučilo svim glavnim konceptima i pristupima kodiranju bez kojih ne bih bio u mogućnosti izrealizirati svoju ideju u sklopu završnog rada.

Prilikom uvezivanja rada, Umjesto ove stranice ne zaboravite umetnuti original potvrde o prihvaćanju teme završnog rada kojeg ste preuzeli u studentskoj referadi

Sažetak

Tema završnog rada je izrada web aplikacija za recenziju i ocjenjivanje video sadržaja koja je napravljena uz pomoć znanja stječenih tijekom školovanja na Visokom učilištu Algebra. Web aplikacija za recenziju i ocjenjivanje video sadržaja ima tri vrste korisnika: dizajnere (*engl. designers*), posjetitelje (*engl. visitors*) i profesore (*eng. professors*). Dizajneri učitavaju video uratke na web stranicu, posjetitelji imaju mogućnost gledanja i komentiranja dok profesori koriste CMS (*engl. Content System Managment*) sustav koji je implementiran u sklopu web aplikacije. CMS sustavi se koriste za manipuliranje velikim brojem informacija u velikim sustavima. Svaki video uradak imaju svoju temu i područje iz kojeg dolazi. Pomoću teme i naslova video uratka dizajneri odabiru kategoriju pod koju žele staviti svoj video uradak u web aplikaciji. Profesori kao super korisnici pregledavaju video uratke učitane od strane dizajnera, komentiraju, ocjenjuju te izvršavaju cijeloukupnu recenziju koristeći CMS.

The topic of the final thesis is the development of web application for reviewing and evaluating video content that is made with the help of knowledge gained during my education at Algebra College. The Web video review and rating app has three types of users: designers, visitors, and professors. Designers upload videos to the site, visitors have the opportunity to watch and comment while professors use the CMS (Content System Management) system implemented in the web application. CMS systems are used to manipulate a great deal of information in large systems. Each video work has its own theme and the area it comes from. Using the theme and title of the video, designers choose the category in which they want to place their video work in a web application. Professors as super users watch videos uploaded by designers, comment on, rate and conduct overall reviews using CMS.

Sadržaj

1. Uvod ... 1

1.1. Uvod u web aplikacije i prednosti nad klasičnim aplikacijama ... 2

1.2. Cilj projekta i završnog rada ... 4

1.3. Namjena i osnova projekta ... 5

1.4. Pregled programskih jezika i alata koji se koriste u izradi ... 6

2. Vrsta korisnika i njihovi odnosi ... 14

2.1. Posjetitelji ... 16

2.2. Dizajner ... 18

2.2.1. Profil ... 19

2.2.2. Stavljanje video sadržaja i njegov prikaz ... 20

2.2.3. Recenzija video sadržaja ... 22

2.2.4. Recenzija i ocjenjivanje ... 27

3. Web aplikacija ... 28

3.1. Organizacija projekta i raspored datoteka ... 29

3.2. Bitni programski elementi ... 32

3.3. Sesija ... 33

3.4. Repozitoriji i njegove funkcionalnosti ... 37

3.5. Baza Podataka ... 42

Zaključak ... 47

Popis kratica ... 48

Popis slika ... 49

Literatura ... 51

1. Uvod

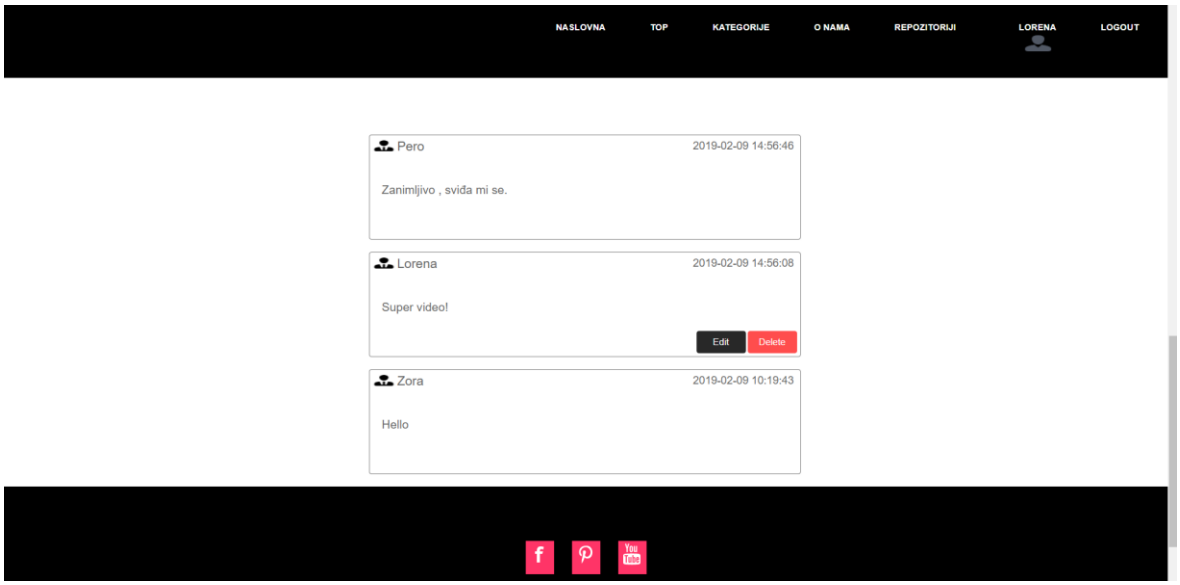
Web aplikacija za recenziju i ocjenjivanje video sadržaja je dinamična web aplikacija kojom se pristupa putem interneta. Dinamična web aplikacija je vrsta stranice na kojoj se obavlja jedna ili više funkcionalnih radnji. Prosječna dinamična web stranica koja je funkcionalna za korisnike, mora sadržavati strukturu baze podataka, skripte koje pozivaju podatke iz baza i funkcionalnost web sučelja.

Projekt je izrađen u pet programskih jezika i tehnologija koji pokrivaju svih pet navedenih područja, a to su : HTML (*engl. Hyper Text Markup Language*), CSS (*engl. Cascading Style Sheeats*), PHP (*engl. Hypertext Preprocessor*), SQL (*engl. Structured Query Language*) i JQUERY. U sklopu projekta najprije je definirana logika, ciljevi i svrha web projekta. Dizajneri učitavaju video uratke na stranicu, profesori ih ocjenju, dok obični korisnici komentiraju video uratke zajedno sa dizajnerima. Zbog lakšeg razumijavanja, u početnom dijelu završnog rada skicirana je logiku odnosa između korisnika. Komponenta koju je trebalo definirati na početku jest baza. Na osnovu toga definirana je vrsta podataka, skiciran je odnos podataka i prioritet. Također, definirane su glavne značajke same baze i njezini sporedni segmenti, kako bi se u konačnici dobila šira slika i izgled. Skicirana je cijeloukupna struktura baze podataka sa tablicama i ključevima te vrstom informacija i elemenata od kojih bi se stranica trebala sastojati. Također, definirani su cijeloukupni putevi korisnika od trenutka kad pristupe web apliakciji pa do mjesta gdje stavljaju određeni sadržaj na web aplikaciju. Bitan segment je i sustav za upravljanje sadržajem (*engl. Content System Managment*) kojem je svrha korištenja podataka od strane korisnika i manipulaciju sa njima. Dizajn i izgled stranice napravljen je na kraju procesa skiciranja. Napravljena je i komunikacija između tri vrste korisnika kako bi aplikacija bila prezentirana korisniku kao dinamička web aplikacija. Također, bilo je potrebno napraviti funkcionalnosti kao što su registracija i spremanje podataka u bazu, stvaranje mogućnosti interakcije između korisnika i mogućnost stavljanja sadržaja na web aplikaciju.

Profesionalna motivacija za izradu projekta je prepoznati problem u nedostatku takvih rješenja te želja da se kroz vlastiti projekt pokuša prezentirati novi način ocjenjivanja dinamičkog web sadržaja.

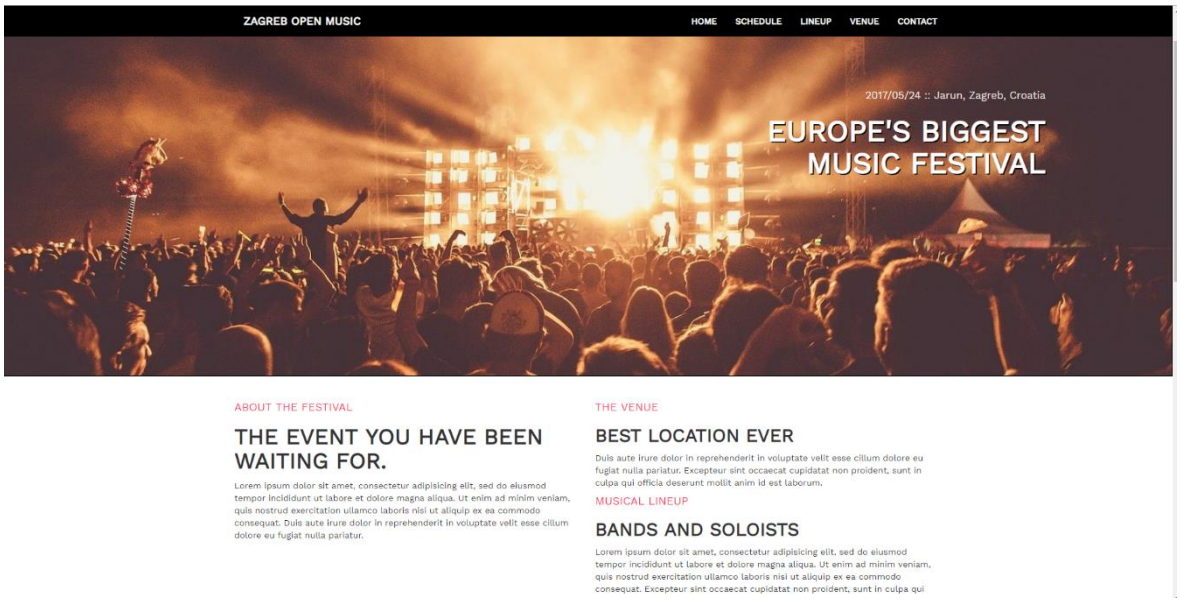
1.1. Uvod u web aplikacije i prednosti nad klasičnim aplikacijama

Web aplikacije su programska rješenja kojima se može pristupiti preko interneta ili intraneta. Web aplikacije mogu biti otvorenog ili zatvorenog tipa ovisno o njihovoj svrsi i namjeni. Web aplikacije se koriste na velikim web projektima, te moraju sadržavati razne dinamičke privilegije. Ako navedena aplikacija sadržava ažuriranje sadržaja, bazu podataka u kojoj se nalaze korisnici i informacije o sadržajima, mogućnost komentiranje sadržaja od strane korisnika, stavljanje sadržaja na platfromu aplikacije ili sl. onda je riječ o web aplikaciji Slika (1.1). Ponekad postoje nejasnoće pri spomenu web aplikacije, stvarajući dvojbu je li to web stranica ili nešto sasvim različito od web stranice. Zapravo, web aplikacija je dinamička web stranica, dinamičkog sadržaja kojoj se može pristupiti samo preko web preglednika. Prednosti web aplikacija nad klasičnim web aplikacijama su u dinamici i funkcionalnosti. Klasična aplikacija ili statična web stranica je aplikacija koja nema nikakvu dinamiku niti skup funkcionalnih elemenata. Klasična aplikacija služi kao pano na kojem su prezentirane određene stvari, koje su statične i sa kojima se ne može manipulirati niti ih koristiti. One prezentiraju napisane informacije koje korisniku služe za informiranje.



Slika 1.1 Primjer dinamičke web aplikacije

Dinamična web aplikacija ili u ovom slučaju moj projekt sadržava proces ulogiravanja i registriranja. Rezultat tog procesa je vidljiv u gornjem desnom kutu (**Pogreška! Izvor reference nije pronađen.**.1) gdje je vidljivo trenutno ime korisnika koji je ulogiran. Klikom na ime korisnika dobivaju se i vidljive informacije o korisniku. Dinamična stranica pruža puno više, s njom je moguće izvršiti niz operacija između korisnika i sadržaja. Korisnici na dinamičkim web aplikacijama su u mogućnosti brisati i uređivati svoje komentare, dok na statičnim to nije moguće. Upravo zbog navedenih činjenica, prednosti dinamike i funkcionalosti sadržaja, dinamične web stranice imaju veliku prednost na statičnim web stranicama. Na (Slika 1.2). je vidljivo da statična web aplikacija nema mogućnosti prikazivanja korisnika koji trenutno koristi aplikaciju, nema dodatne informacije o sadržaju, ne sadrži bazu podataka niti veću kompleksnost informacija. Sve što statična stranica ima jest statičan tekst koji korisniku pomaže da se informira, u ovom slučaju o glazbenom festivalu koji nadolazi. Klasične aplikacije kao što je ova služe samo za informiranje, oglašavanje i marketing. Razlika između web aplikacija i klasičnih aplikacija je ogromna, a najvažnija je da dinamične web aplikacije u današnjem svijetu pridonose svojim prednostima. Kompleksniji procesi plaćanja te obavljanja transakcija upravo se odvijaju preko dinamičnih web aplikacija.



Slika 1.2. Prikaz statične web aplikacije

1.2. Cilj projekta i završnog rada

Cilj projekta je bio da svoju ideju prezentiram kroz dinamičku web stranicu sa jednostavnim CMS sustavom koji nekim terminima naziv i repozitorij. Za postizanje cilja bilo je potrebno proći kroz razne faze izgradnje samog projekta. Od skiciranja logike web aplikacije, baze podataka, odnose korisnika i vrstu sadržaja do samog programiranja zamišljene ideje. Nastojao sam da moj projekt ima početak sredinu i kraj i da svaki proces izvršavanja radnje na web aplikaciji bude jednostavan i praktičan s obzirom na video sadržaj koji sam zamislio da bude na web stranici. Sekundarni cilj je bio odnos između triju vrsta korisnika i njihova definicija.

Razlika odnosa je bila u prijavljivanju i definiranju korisnika prilikom pristupanja na samu stranicu, što je napravljeno kroz proces odabira gdje bi se korisnici ovisno o svrsi i cilju na stranici separirali. Taj odnos je trebalo utemeljiti na programskom kodu kao i na jednostavnosti odnosa. Drugi odnos koji je bio značajniji i koji se sastoji od glavnog cilja izrade ovog projekta, jest odnos između dizajnera i profesora (super korisnika) . Stvaranje jednostavnog pristupa u dio web aplikacije odgovoran za učitavanje u kojem su dizajneri učitavali svoje uradke, definirali im kategorije i slali ih na server, dok su profesori te iste video uratke dobivali u CMS ili repozitoriju gdje su ih ocjenjivali i slali na objavu. Uspostavljanje kompatibilnosti između sustava kojeg koriste super korisnici kako bi manipulirali sadržajem dizajnera bio je jedan od glavnih i bitnijih ciljeva projekta i završnog rada. To je zapravo bilo središte problema , koji mi je predstavljao izazov.

1.3. Namjena i osnova projekta

Glavna namjena ovog projekta je stvaranje domene korisnika i stavljanje video sadržaja koji će biti ocjenjen. Korisnici su podijeljeni u tri kategorije. Prva vrsta korisnika su posjetitelji, čija ciljana skupina nije striktna i ograničena. Pretpostavljeno je da u u domenu spadaju korisnici bilo koje dobne skupine, pošto su oni definirani kao korisnici koji će samo gledati i komentirati video uratke stavljene od strane dizajnera. Kod druge vrste korisnika koja se zovu dizajneri, njihova ciljana skupina su osobe starije od 18 godina teprvenstveno mladi ljudi. Glavna namjena i osnova ovog projekta je mogućnost da studenti prezentiraju svoje kreativne uratke, zamisli i ideje kroz video sadržaj, postavljajući ga na web aplikaciju. Svaki taj video dobiva vrstu povratne informacije, ocjene i komentare što potiče studente na stvaranje i kreiranje novih videa. Kroz navedenu fazu natjecanja koja je i osnova ovog projekta, glavni su faktori super korisnici, koji su profesori sa raznih sveučilišta, veleučilišta ili visokih škola, koji bi svojom reputacijom, iskustvom i znanjem pridonosili još većem značaju ocjenjivanja, što bi cijelom procesu dalo dodatnu originalnost i ozbiljnost. Video sadržaji se učitavaju ovisno o njihovoj vrsti, pod kategorije koje su ponuđene putem web aplikacije. Lista kategorija pokriva uglavnom sva područja koja studenti studiraju, a to su:

- Fizika i Matematika
- Psihologija
- Biologija
- Kemija
- Informatika i IT
- Dramska umjetnost
- Marketinška istraživanja
- Politika
- Ostalo

Svaka od ovih kategorija je navedena prilikom procesa stavljanja video uratka, u koju namjenu i vrstu spadaju kreativni video uradci koje studenti žele uploadati. Također jedan od bitnih faktora namjene ovog projekta je da svaku kategoriju pokriva jedan profesor, dok svaki video ocjenjuju svi profesori, odnosno svih 9 profesora. U konačnici dobivena je aritmetička sredina same ocjene kao i područja koja su sudjelovala u donošenju recenzije.

1.4. Pregled programskih jezika i alata koji se koriste u izradi

Programski jezici koji su se koristili u sklopu web aplikacije su jezici koji su u IT-u (*engl. Information Technology*) standard za izradu web aplikacija. Kako bi naša aplikacija bila dinamična, zadovoljavala uvjete korisnicima koji žele biti dio naše web aplikacije i pravilno manipulirala informacijama, ona mora sadržavati primarne programske i skriptne jezike kako bi to mogla i ostvariti. Programi koji su se koristili u izradi ovog projekta su:

- HTML
- CSS
- JS, Jquery
- PHP
- phpMyAdmin ili RDBMS

Svaki od navedenih programskih jezika ima svoju svrhu u izradi web aplikacije. HTML (*engl. Hyper Text Markup Language*) je programski jezik koji služi za prezentiranje sadržaja koji se nalazi na web stranici te je temeljan kod svakog internetskog sučelja. On je usko vezan sa CSS-om (*engl. Cascading Style Sheets*). CSS služi za manipuliranje html elemenata i dizajniranje istih elemenata. JS (*engl. Javascript*) je dodatni skriptni jezik koji služi za stvaranje funkcionalnosti web aplikacije. Uz JS dolazi i Jquery programski jezik koji je ništa drugo nego biblioteka JS-a. Dva programa su jako slična, samo što se preko jquery uzimaju gotove funkcije napisane u javascriptu. Sljedeći skriptni jezik koji je jako bitan za dinamičnost web aplikacije jest PHP (*engl, Hypertext Preprocessor*). PHP služi za funkcionalnost, dinamičnost i manipulaciju podatcima iz baze podataka. Bitan je za izgradnju web aplikacije i bez njega stranica ne može biti visoko funkcionalna u korist korisnika. Posljednji program koji je korišten u izradi ove web aplikacije je database managment system (RDBMS) mySQL. MySQL program koji kreira i upravlja bazom podataka, tablicama unutar njih i informacijama pomoću SQL naredbi. Unutar tog programa spremamo sve potrebne informacije korisnika, sadržaja i sl. RDBMS je bitan jer je potreban u izgradnji strukture korisnika i logici cijeloukupnog procesa web aplikacije od njezinog početka izrade pa do samog kraja. U svrhu moje web aplikacije sam koristio phpMyadmin koji je je po definiciji open source i koji u sebi sadrži mySQL ali je administriran php kodom.

Prezentacijski jezik za izradu web aplikacije (HTML) jest jezik koji u sebi sadrži temeljni kod unutar svakog preglednika (*engl. browser*). Svaki HTML dokument sastoji se od osnovnih građevnih blokova - HTML elemenata. Svaki,, HTML element sastoji se od para HTML oznaka (engl. *tag*). Također, svaki element može imati i atribute kojim se definiraju svojstva tog elementa. Na samom početku HTML dokumenta preporučljivo je postaviti <!DOCTYPE> element, kojim se označava DTD (engl. *Document Type Declaration*), čime se definira točna inačica standarda . Nakon <!DOCTYPE> elementa, dolazi nam element <html> koji označava početak html, sve ispod njega su ostali elementi koji počinju sudjelovati u izgradnji. Nakon <html> elementa uglavnom nam dolazi <head> element, on predstavlja zaglavlje HTML dokumenta u kojemu se najčešće specificiraju jezične značajke HTML dokumenta kao i sam naslov (engl. *title*) stranice. Pomoću određenih HTML elemenata unutar zaglavlja dodaju se i stilska obilježja stranice, bila ona direktno ugrađena (engl. *embedded*) ili dodana kao referenca na vanjsku CSS datoteku. Često se unutar zaglavlja još definiraju i skripte kreirane u JavaScript jeziku. U <body> elementu kreira se sadržaj HTML dokumenta, odnosno, stranice koju on reprezentira. Svaka HTML oznaka (koja u paru kreira HTML element) počinje znakom < (manje od), a završava znakom > (više od). Zatvarajuća HTML oznaka kreira se na isti način kao i otvarajuća, ali se prije završnog znaka > dodaje i kosa crta / (engl. *slash*). Prikaz jednostavnog html dokumenta (Slika 1.3). Najnovija inačica HTML-a je verzija HTML 5.

```
1   <!DOCTYPE html>
2   <html>
3     <head>
4       <title>Naziv stranice</title>
5     </head>
6     <body>
7       <p>Ovdje se unosi sam sadržaj stranice.</p>
8     </body>
9   </html>
10  </pre>
```

Slika 1.3 Prikaz jednostavnog HTML dokumenta

CSS (*engl, Cascading Style Sheets*) je stilski jezik, koji se koristi za dizajn i manipulaciju prezentacije dokumenta napisanog pomoću HTML jezika.

Kako se web razvijao, prvotno su u HTML ubacivani elementi za definiciju prezentacije (npr. tag <font>), ali je dovoljno brzo uočena potreba za stilskim jezikom koji će HTML osloboditi potrebe prikazivanja sadržaja (što je prvenstvena namjena HTML-a) i njegovog oblikovanja (čemu danas služi CSS). Drugim riječima, stil definira kako prikazati HTML elemente. CSS-om se uređuje sam izgled i raspored stranice. CSS se može direktno povezati na html, ali se može i pisati u posebnom stil (*engl, style*) zaglavlju koji se nalazi unutar zaglavlja html koji počinje sa elementom <style> i završava elementom </style>. U praksi se najćešće koristi način da se css referencira u zaglavlju html-a, tako da se stilski kod piše u zasebnoj css datoteci. Jedna bitna značajka kod CSS jest njegov selektor. Pomoću selektora CSS točno odabire element na html elementu sa kojim želi napraviti vez kako bi bio u mogućnost manipulirati s njim i primjenjivati stilski jezik. Dva glavna selektora koja se pridodaju html elementima kako bi ih CSS mogao prepoznati su:

- Id
- klasa

ID (*engl, identity document*) obuhvaća jedinstveni element, klasa (*engl, class*) može obuhvaćati više od jednog elementa. Id se označava oznakom # ispred oznake, klasa se označava točkom. To su dva od bitnih selektora koji se koriste za označavanje html elemenata. Također je moguće selektirati html primarne (engl, default) elemente kao što su p, h, li, ul, body, footer. Postoji i identifikator koji se naziva lebdjeti (*engl, hover*) pokreće radnju na elementom samo ako korisnik drži pokazivač na njemu. Na (Slika 1.4). Vidljivi su selektori koji su napisani u css formatu.

```
61  #container{
62
63    position:relative;
64    margin:0 auto;
65    width:400px;
66    height:800px;
67    margin-top:15%;
68
69  }
70
71  h3 {
72
73    font-size: 3em;
74  }
75
76
77  .designer:hover, .designer:active
78  {
79
80    color: #0C5;
81    border-color: #0C5;
82    background: #FFF;
83  }
```

Slika 1.4 Prikaz Selektora

Java script je skriptni programski jezik koji služi za stvaranje web aplikacije interaktivnijom i lakšom za korištenje. Java script je u većini slučajeva u sinergiji sa html kodom. Java script je zapravo skriptni jezik koje pruža dodatne funkcionalnost html-u, funkcionalnosti koje on nije u stanju izvesti sam. Java script također sadržava selektore kojima locira html elemente i na osnovu toga kao i CSS nastavlja s radnjama manipulacije. Razlika između Javascripta i Jquery je u tome što je Jquery biblioteka gotovih funkcija napisanih detaljnim java script kodom. Sadržavaju puno sličnih elemenata ali njihova jednostavnost je puno različitija. Java script je znatno složeniji nego jquery ali i teži za korištenje. Jquery nam nudi niz gotovih funkcija za jednostavno implementiranje i manipuliranje html elemenata. Prikaz java script koda koji je zadužen za prilagođavanje grafikona u trenutku kada se pruži vrijednost unutar html elementa. Na liniji 89. je prikazan tipičan primjer izjednačavanja obične varijable koja dobiva vrijednost selektiranog html elementa, odnosno klase koji je označen točkom unutar zagrade. Klasa označava status grafikona (Slika 1.5). Ova linija koda zapravo sa if uvjetima dodaje ostale html klase.

```
87          $(function(){
88          var $ppc = $('.progress-pie-chart'),
89              percent = parseInt($ppc.data('percent')),
90              deg = 360*percent/100;
91          if (percent > 50) {
92              $ppc.addClass('gt-50');
93          }
94          $('.ppc-progress-fill').css('transform','rotate('+ deg +'deg)');
95          $('.ppc-percents span').html(percent+'%');
96          });
```

Slika 1.5. Prikaz java script koda za prilagođavanje html grafikona.

(Slika 1.6). Prikazuje jednostavni jquery kod za prilagođavanje navigacijskog sučelja s obzirom na njegov položaj na web aplikaciji. Na liniji 105. je vidljiva već gotova funkcija scrollTop() [1]koja u sebi sadržava gotov java script kod koji kaže :“ako je pozicija odabrane html klase unutar if[2] uvjeta već od 1. Odnosno ako se stranica počne klizati (*engl, scroll*) prema dolje što znači da je vrijednost veća od 1, dodaj mi css stil pozadinu crnu.“ Zapravo na toj slici je vidljiva interakcija jquery, html i css zajedno. Također tu je još jedna gotova funkcija addClass()[3], koja izvršava dodavanje klase definirane u css-u. Klasa može biti

[1] Jquery funkcija
[2] Uvijet koji je definira ako se nešto dogodi
[3] Jquery funkcija za dodavanje klase

definirana ne samo striktno u html, u ovakvom slučaju ona može biti definirana kao i css element. Prikaz jquery koda koji služi za pozicioniranje navigacijskog sučelja ovisno o poziciji (Slika 1.6).

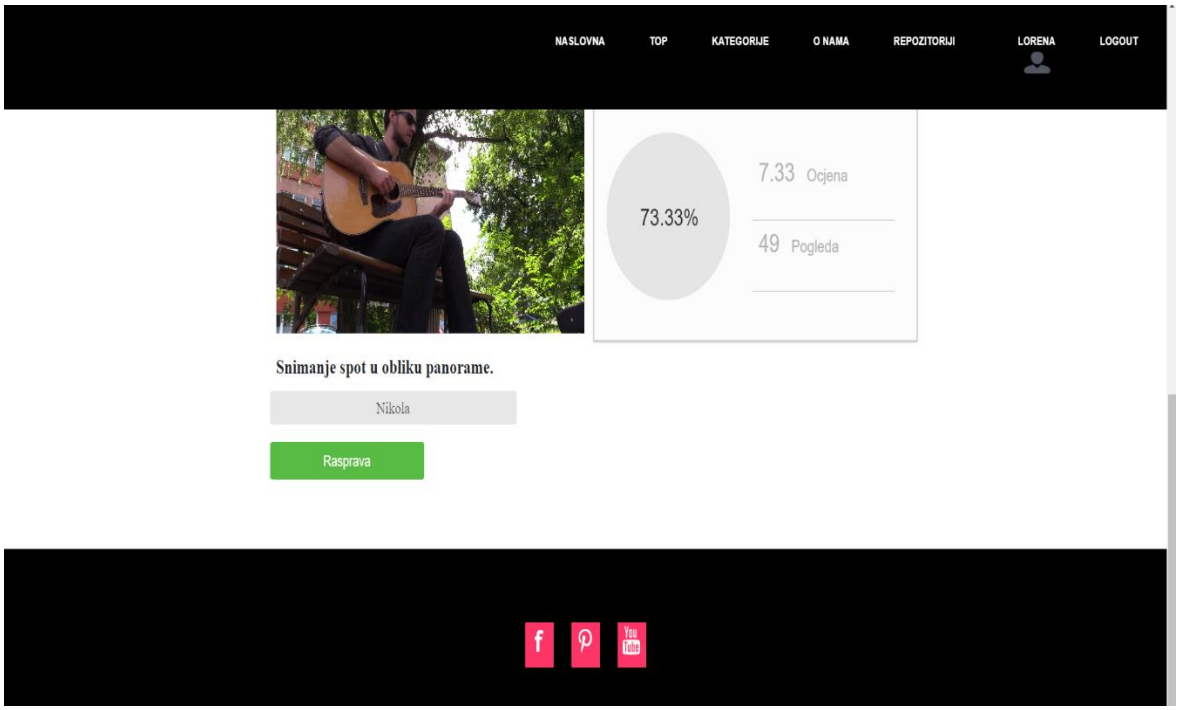
```
103                 $(window).scroll(function(){
104
105                     if($(this).scrollTop() > 1 ){
106
107                         $('nav').addClass("sticky").css("background-color", "black");
108                     }
109                     else if($(this).scrollTop() < 1 ){
110
111                         $('nav').addClass("sticky").css("background-color", "transparent");
112                     }
113                 });
114
115
116             $(document).ready(function(){
117
118                 $("#prikazi").click(function(){
119
120
121                     $("#profesori").slideToggle("slow", "swing");
122
123                 });
124
125
126             });
```

Slika 1.6 Prikaz jquery koda.

Rezultat napisanog jquery koda vidljiv je na (Slika 1.7).

Klasa koja se dodaje se naziva ljepljiva (*engl, sticky*) vrsta klase koja u sebi sadrži css svojstva, kao što su pozadina kontejnera koji će se pojaviti oko navigacije sučelja u trenutku kada se sama navigacija spusti ispod zadanog uvjeta definiranog na liniji 105. U ovom slučaju to svojstvo je pozadinska boja crne boje. Naravno mi možemo promijeniti svojstva klase tako da se pozicioniramo na tu klasu unutar CSS fajla ili jednostavno na HTML-u otvorimo <style> [4]naredbu i unutar nje bez da moramo ići na definirani referencirajući css fajl upišemo naredbe. Također je moguće uz pravilan napisani css, napisati svojstva i unutar funkcije, odnosno dodati funkciju AddClass() kao što je navedeno prethodno. Jquery kao biblioteka koja je napisana u javascriptu je dosta bitan dio moje web aplikacije jer sam pomoću njega uspio iz realizirati ovakvu vrstu funkcionalnosti na mojoj stranici, što je čini još više funkcionalnom i prijateljski nastrojenom prema mojim korisnicima.

[4] Nardba stil unutar koje pišemo stilskim jezikom



Slika 1.7 Prikaz navigacijskog sučelja prilagođen jquery kodom s obzirom na poziciju.

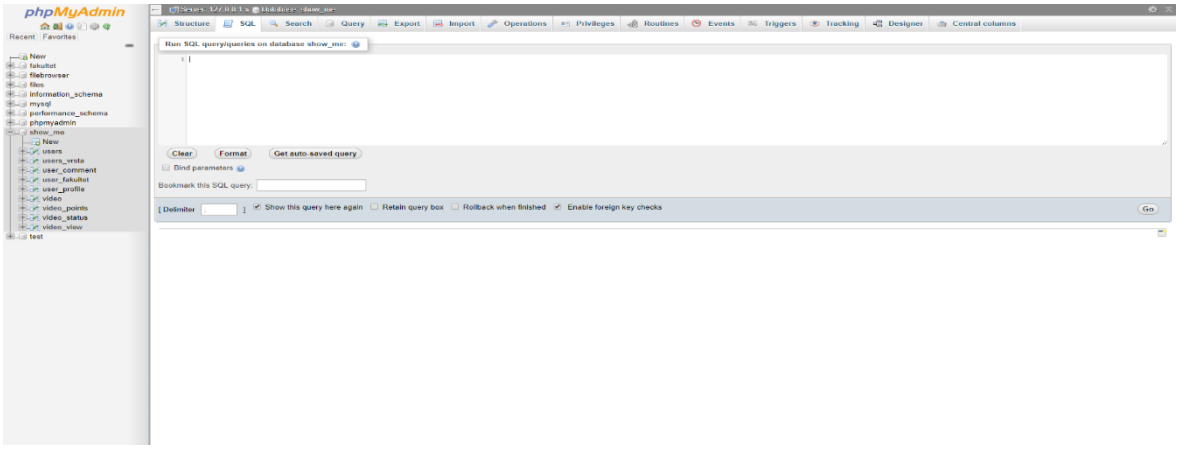
PHP (lukavo rekurzivna skraćenica za PHP: Hypertext Preprocessor) je popularan open source skriptni jezik namijenjen za web development. Vrlo ga je lako umetnuti u HTML i tom simbiozom dobiti dinamički kreirane web stranice. Razlikuje se od klijentskih skriptnih jezika poput JavaScripta jer se izvršava na poslužitelju. Rezultat izvršavanja je HTML kod koji se šalje browseru [5]kojeg razumije bez potrebe za bilo kakvim nadogradnjama. PHP je vrlo pogodan za početnike u svijetu programiranja, ali istovremeno ima mnogo toga za ponuditi i iskusnim programerima. PHP je zamišljen kao server-side skriptni jezik, tako da ga možemo koristiti za prikupljanje podataka iz obrazaca, dinamičko generiranje sadržaja stranice ili za slanje i primanje kolačića [6](*engl, cookie*). Korištenjem PHP-a, niste ograničeni samo na generiranje HTML-a. Moguće je u letu generirati slike, PDF i Flash datoteke. Napravljene datoteke nije nužno odmah prikazati, već ih možete spremiti na server i koristiti kasnije. Jedna od najjačih i najznačajnijih mogućnosti PHP-a je njegova podrška za različite baze podataka. Izuzetno je jednostavno nadograditi web aplikaciju korištenjem specifičnih ekstenzija poput mysqli, apstraktnog PDO (PHP Data Objects) sloja ili spajanjem na bilo koju bazu koja podržava ODBC (Open Database Connection) standard. U prethodnom tekstu je objašnjeno sve što bi se trebali znati o php-u. Njegova svrha je jednostavna . Kao skriptni program jedan je od bitnih faktora za izgradnju dinamičke web stranice. Na (Slika 1.8). Linija 8. Prikazana je implementacija php unutar html i korištenje html elemenata. Na slici je vidljiv jako bitan element php a to je echo. Echo element zapravo prikazivanje html elementa ili ostalih elemenata kroz web sučelje vidljivo korisniku.

```
1   <!DOCTYPE html>
2   <html>
3     <head>
4       <title>Naziv stranice</title>
5     </head>
6     <body>
7       <p>Ovdje se unosi sam sadržaj stranice.</p>
8      <?php echo '<p> Hello world </p>' ?>
9     </body>
10  </html>
```

Slika 1.8 Prikaz php koda unutar html-a

[5] Aplikacija koja omogućuje pristup internetu
[6] Mali dio podataka poslan i pohranjen na korisnikovo računalo.

Program phpMyAdmin je otvorenog izvora (*engl,open source*), koristi se kao administracijski program za korištenje mySql baze podataka. Program je napisan u php-u i od značajne je uloge za manipulaciju informacijama. Unutar phpmyAdmina imamo opcije za stvaranje baze podataka i njezinih tablica. Sami definiramo i određujemo od čega će se naša tablica sastojati koliki broj redaka i stupaca, kakvih vrsta informacija i koliko velikih. To je program koji se nalazi na poslužitelju i koji je u stalnom kontaktu sa php-om i php skriptama. Bazu kreiramo u phpMyAdminu dok konekciju obavljam php kodom iz skripte prema bazi. Program sadrži skoro pa sve značajke mySQL u njemu možemo izrađivati puno tablica i struktura jer nije ograničen, jako je bitan pri izgradnji dinamičke web aplikacije, iz razloga jer sadržava sve potrebne informacije koje bi se s phpom prikazivale na ciljanim skriptama. (Slika 1.9). Prikazuje sučelje phpMyAdmin. Unutar phpMyAdmina je dopušteno koristiti sql naredbe kako bi lakše dobili uvid u naše informacije, sortiranjem i manipulacijom njih sami.



Slika 1.9 Prikaz phpMyAdmin sučelja

Proces implementacije phpMyAdmina se odvija instalacijom na platformi na kojoj se odvija izgradnja projekta ili web aplikacija. Za izradu projekta je korišten XAMPP[7] softverski paket koji se implementira na Apache[8] web server i koji unutar svog paketa sadrži phpMyadmin koji se u trenutku podizanja Apache web servera može uključit naknadno. Svaka platforma funkcionira na sličnom principu, myPhp admin se još uvijek i danas pretežito koristi za komunikaciju s bazom i bitan je faktor kod svake dinamičke web aplikaciji.

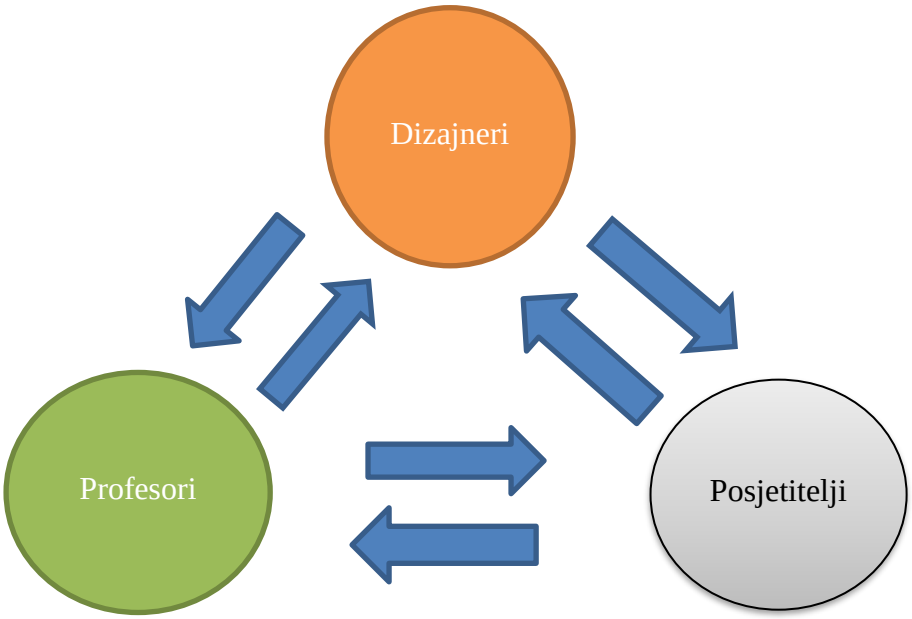
[7] Softverski paket koji sadrži phpMyAdmin
[8] Web server open-source koji je služio za hostanje server lokalno

2. Vrsta korisnika i njihovi odnosi

Vrste korisnika na mojoj stranici je jedan od bitnijih faktora, tako da ću u ovom poglavlju pisati isključivo o svakoj pojedinoj vrsti korisnika za sebe, prezentirati njihovo sučelje ulogiravanja, mogućnosti koje oni imaju prilikom korištenja web aplikacije, razlike između njih i njihove odnose. Tri vrste korisnika posjećuju stranicu:

- Posjetitelji
- Dizajneri
- Super korisnici ili profesori

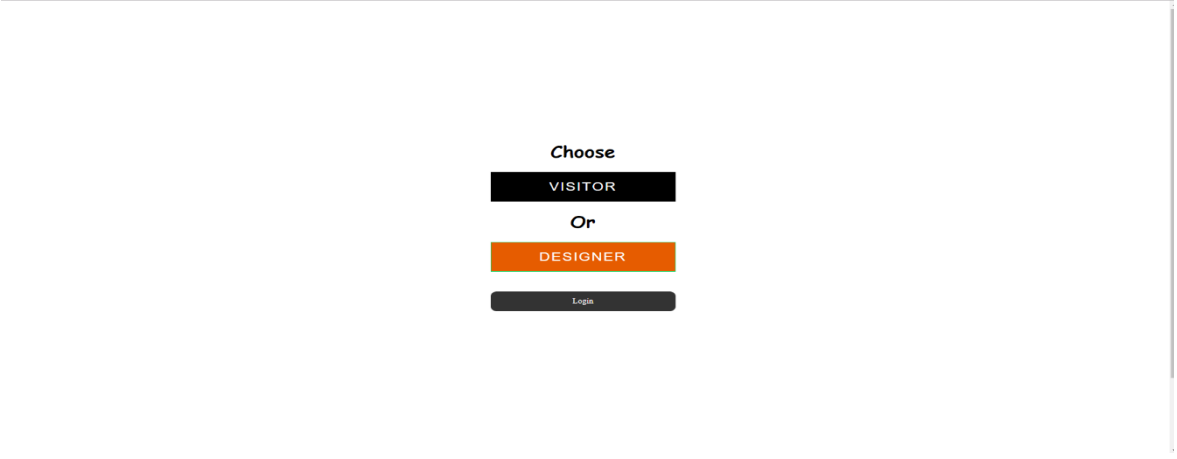
Odnos koji bi trebao funkcionirati između korisnika se definira kao međusobni odnos između svake grupe korisnika (Slika 2.1).



Slika 2.1 Prikaz odnos korisnika

Dizajneri dobivaju povratne informacije od posjetitelja koji komentiraju video nakon što on bude ocjenjen, tako da se tu stvara prvi odnos dizajnera i posjetitelja. Drugi odnos se odnosi na dizajnere i profesore. Dizajneri dobivaju povratne informacije kao što su ocjena i komentari od strane profesora za njihov video uradak u sklopu recenzije, dok ti isti profesori gledaju video, čitaju opis i nadgledaju dodatne komentare dizajnera nakon što video bude ocjenjen, također dizajner i profesori imaju nekakve zajedničke stvari kao što su poznavanje

procesa studiranja i implementiranje studentskog znanja u projekte kao što su u ovom slučaju inovativni video sadržaji, čime se stvara još jedan bolji odnos i poznavanje. Treći odnos je između profesora i posjetitelja, na kojem veliki dio povratnih informaciju dobivaju profesori na njihove ocjene i komentare video uratka, tako da nisu jedini dizajneri koji su recenzirani nego i profesori, što dovodi do puno bolje stabilnosti i odnosa između korisnika. Posjetiteljima kao zasebnoj domeni je omogućeno gledanje profila dizajnera i profesora. Profesori kao skupina super korisnika imaju mogućnost gledanja dizajnerskih profila i kontaktiranje dizajnera čiji su video ocijenili. To je omogućeno preko kontakt linka koji se nalazi na profilima dizajnera, prilikom čega se profesorima otvori kontakt obrazac za slanje email-a na email definiran od strane dizajnera. Također je moguć odnos kontaktiranja između posjetitelja i dizajnera. Dizajneri su također u mogućnosti kontaktirati profesor ne isti način, kada posjete njihov profil, pomoću tipke koja glasi „pošaljite email“. Na web aplikaciju nije moguće pristupiti direktno, nego je potrebno definirati koja vrsta korisnika želimo biti prije nego što pristupimo stranici. Nakon što se odredi koja vrsta korisnika želimo biti, prolazimo kroz proces registracija. Dva su procesa registracije i nisu isti jer postoje razlike u zahtjevima informacija za registraciju koje se zapravo definiraju po razlici između posjetitelja i dizajnera. Web aplikacija od nas traži izbor između dvije vrste korisnika a to su posjetitelj (*engl. visitor*) i dizajner *(engl. designer) (*Slika 2.2). U trenutku kada se odabere vrsta korisnika, korisnika se šalje na registracijski obrazac. Ako je korisnik već registriran na stranici onda je proslijeđen na login formu, pritiskom na login[9].

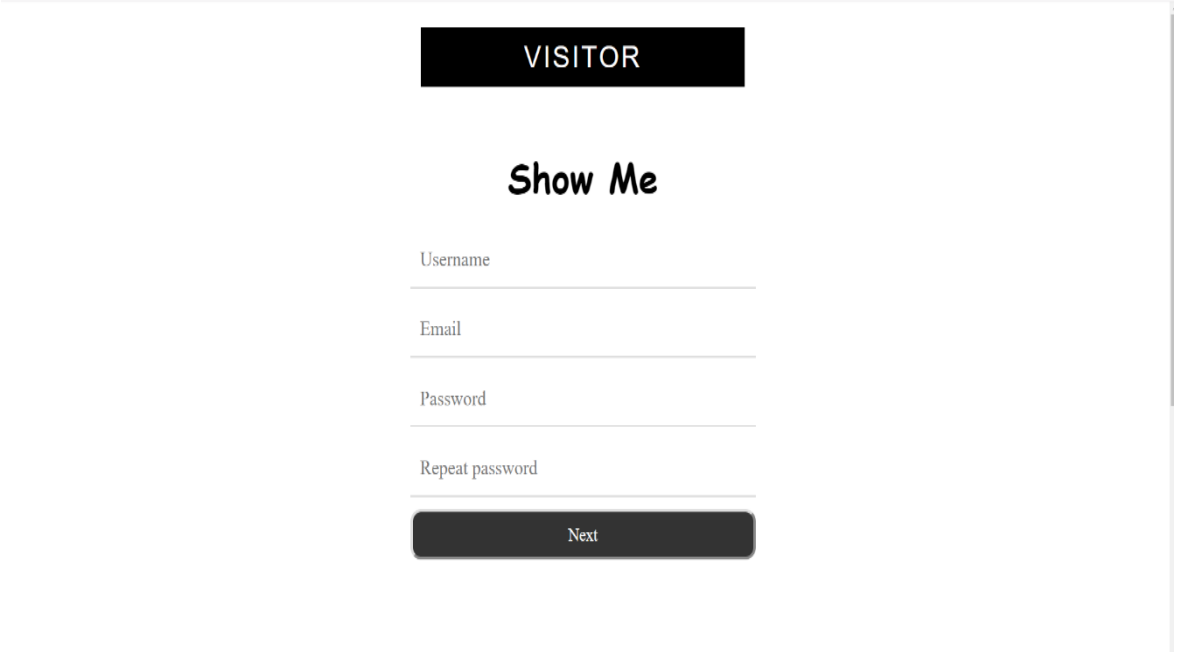


Slika 2.2 Prikaz sučelja za odabir korisnika.

[9] Gumb koji označava odlazak na pristupnu formu za ulazak u web aplikaciju

2.1. Posjetitelji

Posjetitelji su korisnici koji dolaze na web aplikaciju u cilju gledanja i komentiranja video uradaka. Njihovo web sučelje i navigacijsko sučelje se razlikuje od ostalih korisnika. Prilikom registriranja na stranicu, posjetiteljev registracijska forma je manja od dizajnerske, sadržavajući upola manje traženih informacija. Na posjetiteljevoj formi se traže podatci kao što su: korisničko ime, email, šifra i potvrđivanje šifre korisnika. Posjetitelji nisu kompleksni, njihova forma je jednostavan. Posjetiteljeva forma za registraciju je vidljiva na (Slika 2.3).



Slika 2.3 Prikaz Posjetiteljeve registracijske forme.

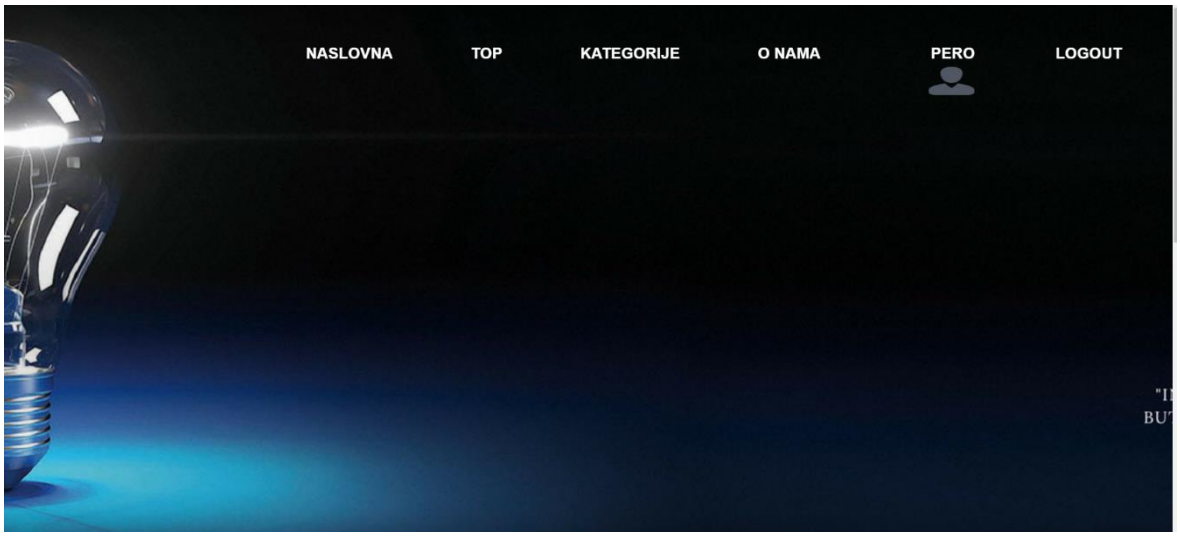
Razlog zašto je registracijska forma posjetitelja skromna jest zbog toga što posjetitelj ne treba nikakve dodatne informacije osim imena, šifre i elektroničke pošte (*engl, electronic mail)*. To su zapravo osnovne informacije koje se potrebne za kontaktiranje dizajnera i komentiranja videa pod njegovim korisničkim imenom. Druga bitna razlika i identifikacija ovog korisnika jest njegovo navigacijsko sučelje.

Navigacijsko sučelje posjetitelja je isto tako skromno kao i registracijska forma. Posjetitelju je ponuđeno par osnovni stavki u navigacijskom sučelju u trenutku kada se on ulogira na stranicu pod svojim korisničkim imenom.

Navedene opcije su:

- Naslovna
- Top
- Kategorije
- O nama
- Korisničko ime
- Logout

Prikaz navigacijskog sučelja korisnika je vidljiv na (Slika 2.4). Posjetitelj je tu da posjeti stranicu, pogleda aktualne i ocjenjene video uratke, komentira, u najboljem slučaju kontaktira dizajnera i to je sve što on može napraviti. Posjetiteljeva namjena kao korisnika jest da bude publika i glas za video uratke i ocjenjivanje profesora. To je skupina korisnika bez kojeg projekt nebi bio moguć u smislu odnosa između korisnika i njihove stabilnosti.

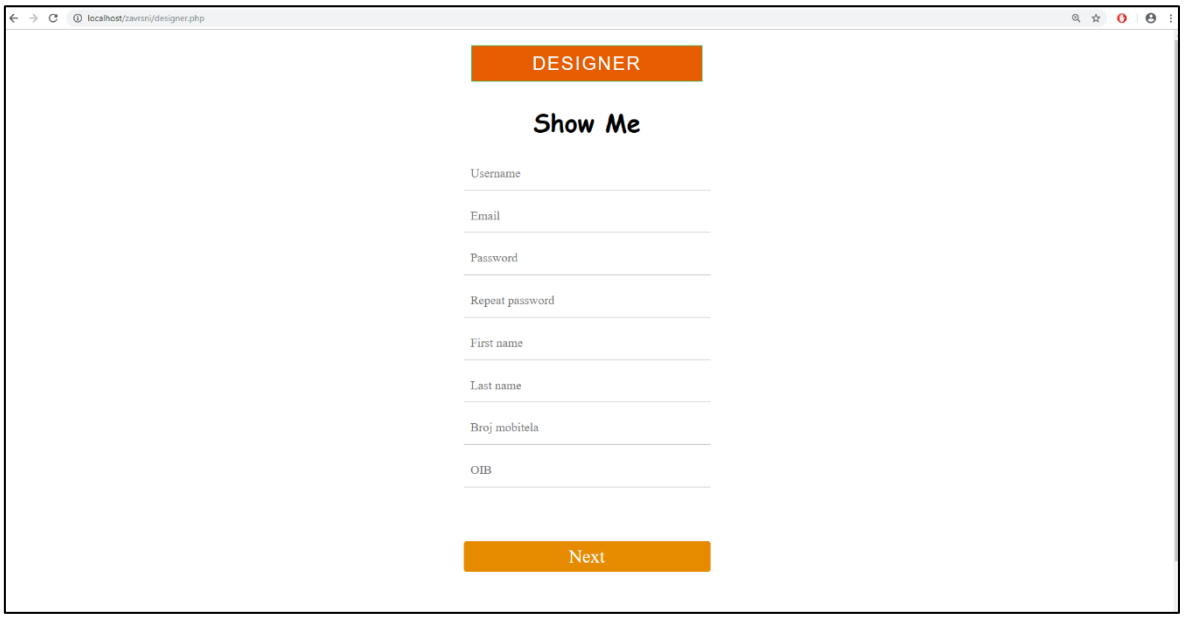


Slika 2.4 Prikaz navigacijsko sučelja posjetitelja.

Kod komentiranja video uratka posjetitelji su u mogućnosti vidjeti ostale komentare drugih posjetitelja ali nisu u mogućnosti biti u kontaktu s njima. Posjetitelji na svom navigacijskom sučelju također imaju karticu o nama, koja ih vodi na kontakt obrazac gdje mogu kontaktirati administratore web aplikacije i upitati ih za razna pitanja koja im nisu jasna u sklopu web stranice kao i prigovori, prijava kvarova vidljivih na stranicama i sl. Navigacijsko sučelje je kroz ovaj projekt segmentirano i prilagođeno ekvivalentno kompleksnosti vrste usera koji je u ovom slučaju gledatelj ili obični user.

2.2. Dizajner

Dizajneri kao vrsta korisnika su motiv izgradnje projekta u sklopu završnog rada. Oni su korisnici koji se registriraju na web aplikaciju s namjerom da prezentiraju svoje radove bili oni individualni ili grupni. Kao vrsta korisnika i ciljana skupina, primarna ideja je da su to osobe starije od 18 godina odnosno studenti sa raznih sveučilišta, visokih škola, veleučilišta ili sl. Nije strogo određeno da dizajneri moraju biti studenti iako je stranica namijenjena njima, to mogu biti mlade osobe koje ne pohađaju nikakvo visoko obrazovanje, već se bave kreativnim radom u slobodno vrijeme. Prilikom registriranja kroz dizajnersko sučelje, dizajneri su obavezni unijeti puno više informacija o sebi za razliku od posjetitelja. Dizajneri moraju postaviti korisničko ime, email (*engl, electronic mail)*, šifru, ime, prezime, broj mobitela i oib. Svi ti podatci su potrebni kako bi se dizajnera moglo brzo i jednostavno kontaktirati. Ime, prezime i broj mobitela su informacije koje su potrebne profesorima u slučaju dodatnog kontaktiranja u sklopu dobre kritike prema video uratku, broj mobitela kao vrsta informacija je vidljiva samo profesorima , dok je oib i broj mobitela informacija koja je kriteriji za ulogiravanje na stranicu kao dizajner, kao i dodatne informacije za kontaktiranje. Ako se dizajneri žele registrirati kao grupa studenata, ime i prezime je također potrebno za ispuniti, jer je to informacija o sobi na koju se podiže dizajnerski račun u slučaju ako se radi o grupi studenata.



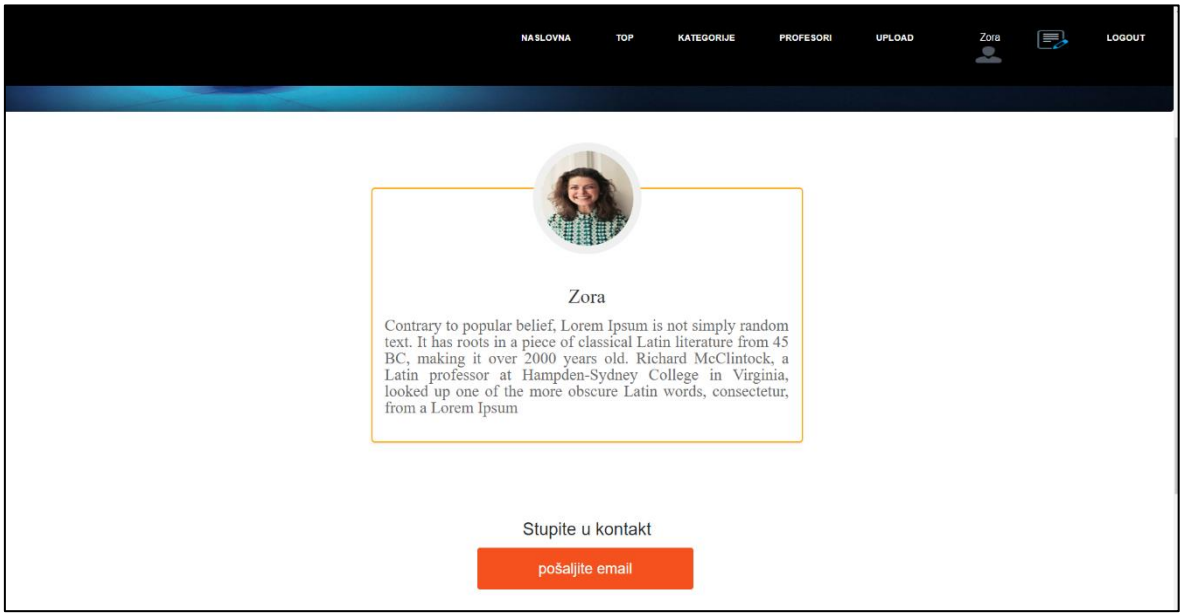
Slika 2.5 Prikaz dizajnerskog registracijskog sučelja.

2.2.1. Profil

Dizajneri kao što je napisano na prethodnim stranicama, imaju svoj profil koji je glavni izvor dodatnih informacija. Na njihovom profilu se nalaze sve potrebne informacije, koje su potrebne gledateljima da se više informiraju o autoru samog video uratka. Informacije koje se nalaze na profilu su:

- Korisničko ime
- Slika korisnika
- Opis

Dizajneri su također u mogućnosti uređivati svoj profil, gdje mogu mijenjati svoj osobni opis profila i sliku. Dizajnerski profil i njegovo web sučelje je vidljivo na slika 2.6.



Slika 2.6 Prikaz profila dizajnera

Kao što je vidljivo na (Slika 2.6). Iznad profila dizajnera je dizajnersko navigacijsko sučelje slično posjetiteljskom ali sa dodatnom karticom upload i ikonicom uredi. Sa karticom upload dizajneri otvaraju posebno web sučelje na kojem uploadaju svoje video uratke. Ikonica uredi se na istoj slici nalazi između ikone usera i linka logout. Ta navigacija kartica dizajnera vodi na uređivanje profila, dok ikonica user prikazuje originalni prikaz dizajnerskog profila, kako ga imaju mogućnost vidjeti ostali korisnici, kao što je prikazano na (Slika 2.6). Na poglavlju vrsta korisnika i njihovi odnosi, spomenuta je činjenica da se dizajneri mogu kontaktirati. Na slici je također vidljiv gumb „pošaljite email“.

2.2.2. Stavljanje video sadržaja i njegov prikaz

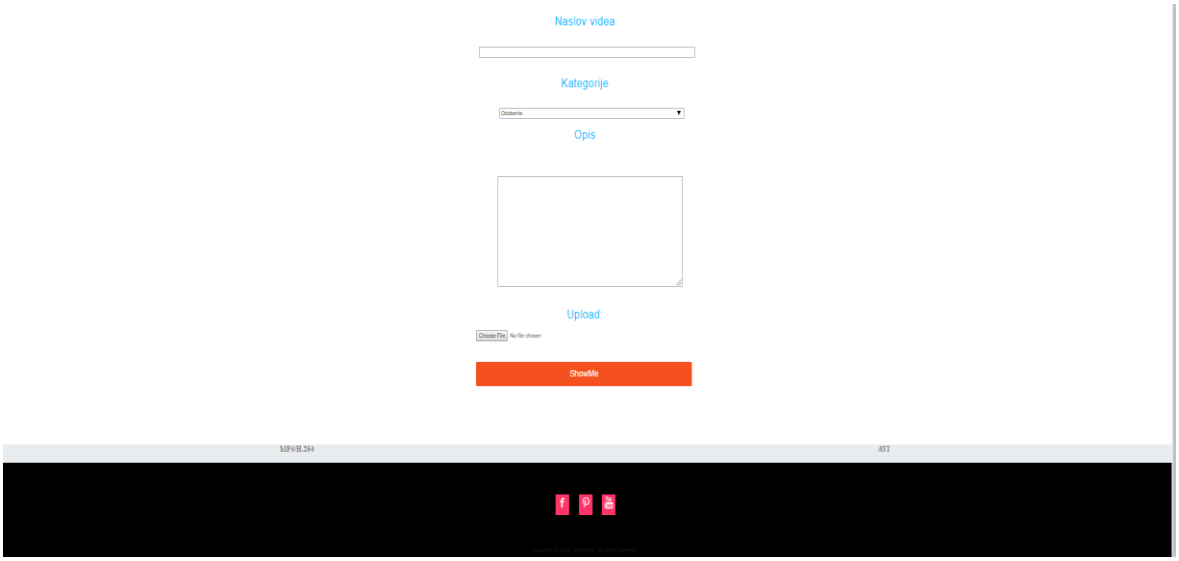
Upload je najbitniji proces i akcija koja je dozvoljena dizajnerima, ona kao takva čini i glavninu i značenje mog projekta. To je posebno web sučelje na koje dizajneri dolaze klikom na karticu upload koja je vidljiva na (Slika 2.5) u navigacijskom sučelju. Prilikom dolaska na sučelje za upload video uratka, (Slika 2.6). dizajnerima je ponuđen obrazac koji u sklopu upload videa moraju ispuniti. U ponuđenom obrascu se nalaze različiti uvjeti za upload video uratka.

- Naslov video
- Kategorije
- Opis
- Postavljanje videa

Prvi i najbitniji dio za ispunjavanje forme je naslov videa gdje dizajneri stavljaju naslov video uratka koji će biti vidljiv na glavnoj naslovnoj strani ispod videa. Sljedeća stavka je kategorija u kojoj dizajneri definiraju kojem području njihov video uradak pripada, zatim slijedi video opis u kojem detaljno opisuju o čemu se točno radi u njihovom video uradku i što ih je točo inspiriralo da naprave takav video i postave ga na web stranicu. Opis je vidljiv ispod videa tijekom gledanja i komentiranja videa . Također ispod upload obrazca je vidiljva lista video formata koje web aplikacija podržava, to su:

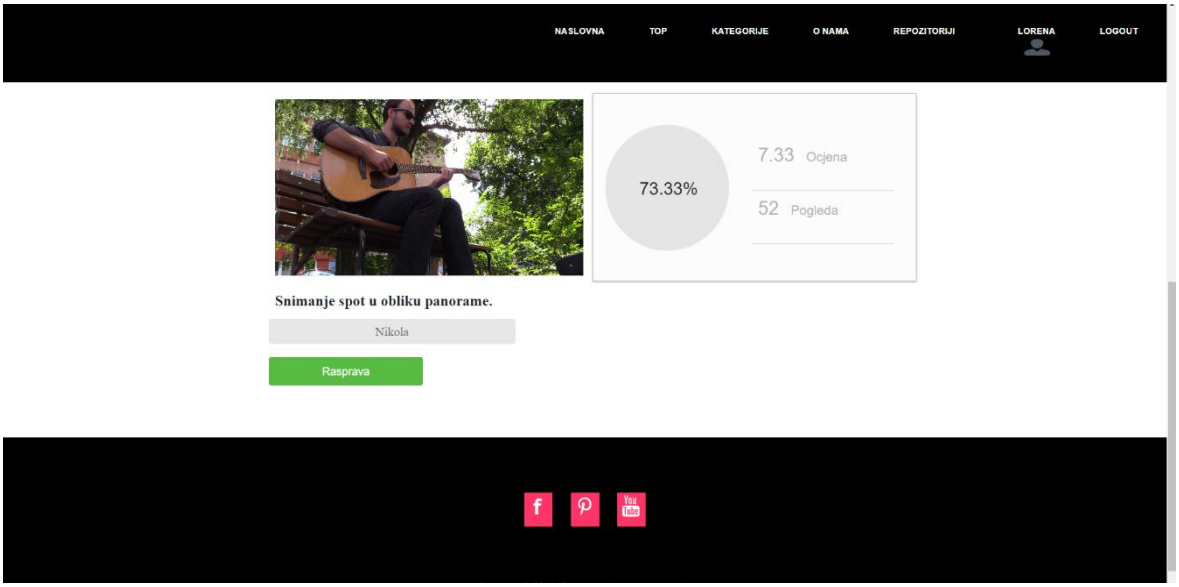
- AVI (*engl, Audio Video Interleaved*)
- MP4 (*engl, Advanced Video Coding*)

AVI datoteke pružaju podršku za višestruki streaming zvuka i videa, što je zapravo normalan standard u digitalnom zapisu video sadržaja, također tu je i MP4 format koji je znatno kvalitetniji i ima puno bolju jasnoću slike i zvuka, ali je i veći za razliku od AVI. Funkcionalnost uploada funkcionira na način da kada dizajner ispuni formu sa informacijama koje su potrebne za upload videa, video sadržaj odlazi na web server. Kada će video biti spreman za ocjenjivane i kada će on biti na web serveru ovisi o brzini interneta, veličini sadržaja i kvaliteti sadržaja. Kad se video postavi na web server automatski odlazi na repozitorij web aplikacije u kojem se izlistava kao ne ocjenjen nakon što je uploadan. Kada super korisnici otvore web sučelje na kojem se nalazi repozitorij, video je izlistan i vidljiv kao ne ocjenjen.



Slika 2.6. Prikaz dizajnerskog video upload sučelja

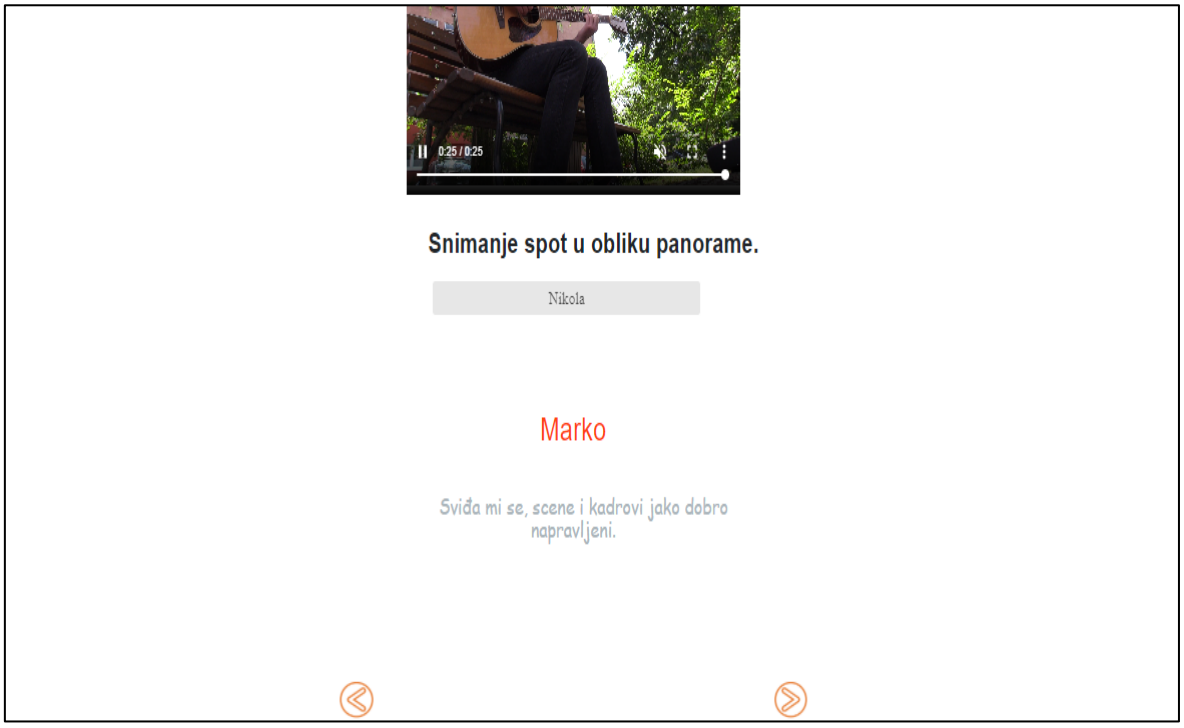
Nakon što je video uredno poslan na web server i ocjenjen, rezultati ocjenivanja se prikazuju na glavnoj naslovnoj stranici web apliakcije, koja je dostupa svim vrstama korisnika. To je prvi prikaz videa i njegova prva prezentacija. Na (Slika 2.7) vidljiv je video koji je trenunto ocjenjen I nalazi na naslovnoj strain. Također, vidljiv je naziv videa i ime korisnika koji je učitao video. Na desnoj strani videa odnosno na grafu je vidljiv status videa odnosno prosječna ocjena dobivena artimetičkom sredinom svih 8 profesora koji su ocjenili video. Ispod ocjene je vidljiv broj pregleda videa od strane svih korisnika. Na grafu je vidljiva artimetička ocjena prezentirana u postotku.



Slika 2.7 Prikaz video sadržaja na naslovnoj strani

2.2.3. Recenzija video sadržaja

Na prethodnoj (Slika 2.7) vidljiv je gumb rasprava koji se nalazi ispod naziva autora video uratka, Klikom na gumb korisnici otvaraju novo web sučelje na web stranici te otvaraju drugi prikaz video sadržaja ispod kojeg se nalaze opis videa, komentari profesora, video sadržaj i komentari posjetitelja . Otvaranjem dijela web aplikacije koji je prikazan na (Slika 2.7) dobiva se web sučelje u kojem je prikazan video, naslov videa i naziv autora. Ispod video uratka nalaze se komentari ostalih korisnika. Na tom sučelju je prezentiran dizajnerov rad kroz više faza ocjenjivanja. Korisnik kao posjetitelj ili dizajner ima uvid u cijeloukupni proces ocjenjivanja i komentiranja video uratka. Ovaj dio web aplikacije je zapravo namjenjenjen recenziji video uratka kroz koju je moguće dobiti kompletan prikaz kritika video uratka. Korisnici su u mogućnosti komentirati, uređivati i brisati svoje komentare kako je prethodno pokazano na (Slika 2.9). Vrijeme kada su postavljeni komentari vidljivo je u gornjem desnom kutu a soritranje komentara definirano je vremenski kada je komentar postavljen. Iznad komentara se nalazi prostor u kojem se nalaze komentari super korisnika. Za prikazivanje ostalih komentara profesora potrebno je koristiti desnu i lijevu strelicu, prilikom čega će se ostali komentari super korisnika pridolaziti ovisno koju strelicu stisnemo. Komentiranje je jedan od bitnih faktora jer pruža povratne informacije i kritike dizajnera. Korisnici kojima je omogućeno komentiranje su posjetitelji, dizajner i super korisnici (profesori).

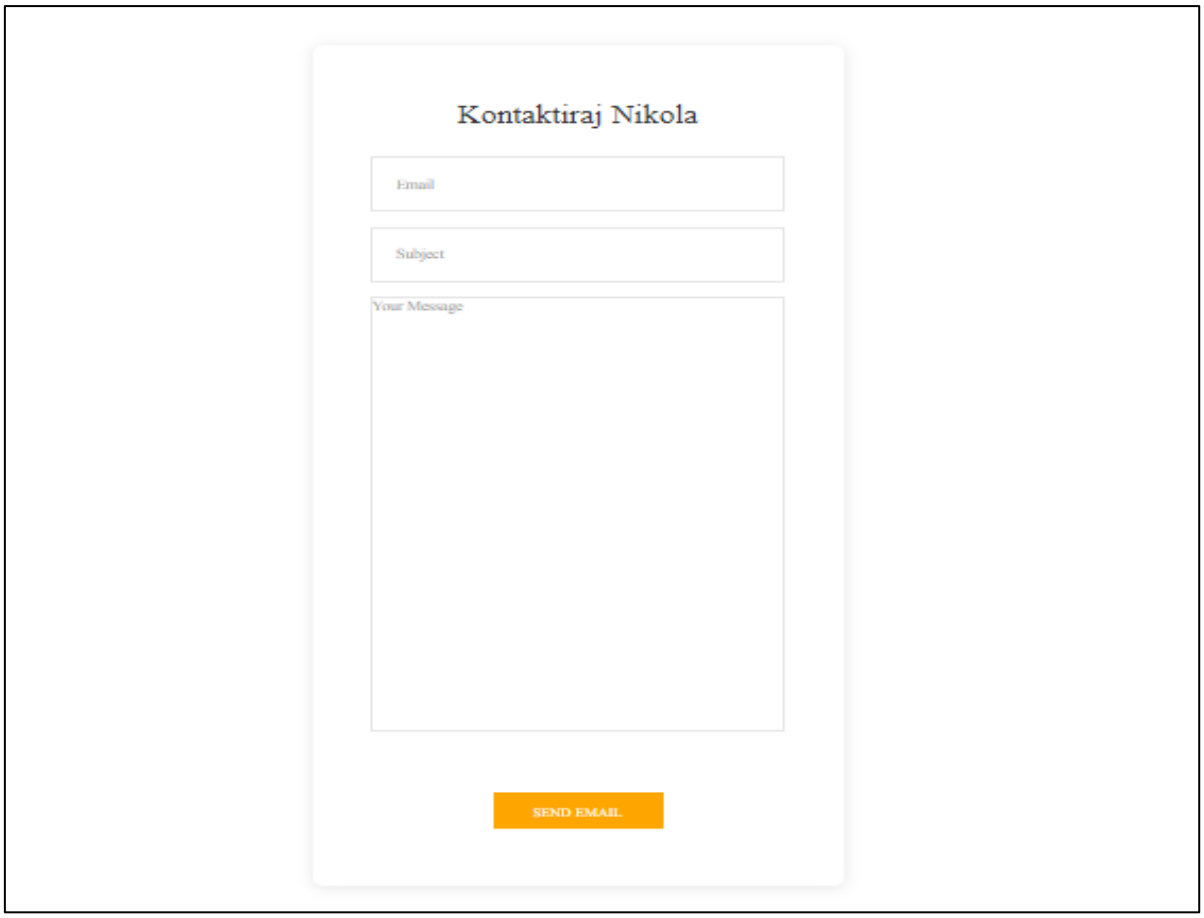


Slika 2.8 Prikaz video sadržaja i komentara profesora

Slika 2.9. Prikaz liste komentara ispod video uratka

Na (Slika 2.9) prikazani su komentari koji se nalaze ispod video uratka. Komentar korisnika Lorene ima opcije uredi (*engl, edit*) i obriši (*engl,delete*) komentar. Korisniku Loreni su prikazane te informacije zao što je ona trenutno ulogirana na web aplikaciju i ona ima mogućnost brisanja komentara. Dio o sesijama i identifikaciji korisnika će biti definirano u idućim poglavljima. Komentari kao što je vidljivo su izlistani prema datumu objave kako većina komentara na današnjim poznatim web aplikacijama funkcionira. Klikom na ime autora video uratka koji je vidljiv na (Slika 2.8) korisnika se preusmjerava na web sučelje za kontaktiranje navedenog autora video sadržaja (Slika 2.11). Kontakt obrazac je isti za kontaktiranje profesora i dizajnera. Na svakom kontakt obrazcu osim na onome za kontaktiranje same web stranice je vidljivo korisničko ime korisnika kojeg želimo kontaktirati. Algoritam izlistavanja komentara se odvija na način da se komentari i korisnici izlistavaju iz baze. U trenutku kada korisnik napiše komentar ispod navedenog video sadržaja podatci njegovog ID-a kao i podatci samog komentara odlaze u bazu pod posebnu tablicu koja se zove komentari. U trenutku kada je proces spremanja komentara završen i kada se podatci spreme u bazu, dolazi sljedeći proces koji u vrlo kratkom periodu te iste podatke koji su spremljeni u bazi izlistava na web aplikaciju i prezentira korisnicima. Svaki video je identificiran ID brojem korisnika i ID-om samog komentara. Kada se odrađuju operacije uređivanja i brisanja, šalje se sql zahtjev sa dvije SQL komande. Alter[10] za uređivanje i delete za brisanje. Takve operacije se obavljaju identificirajući se s ID-em komentara.

[10] Uredi podatak koji je spremljen u bazi



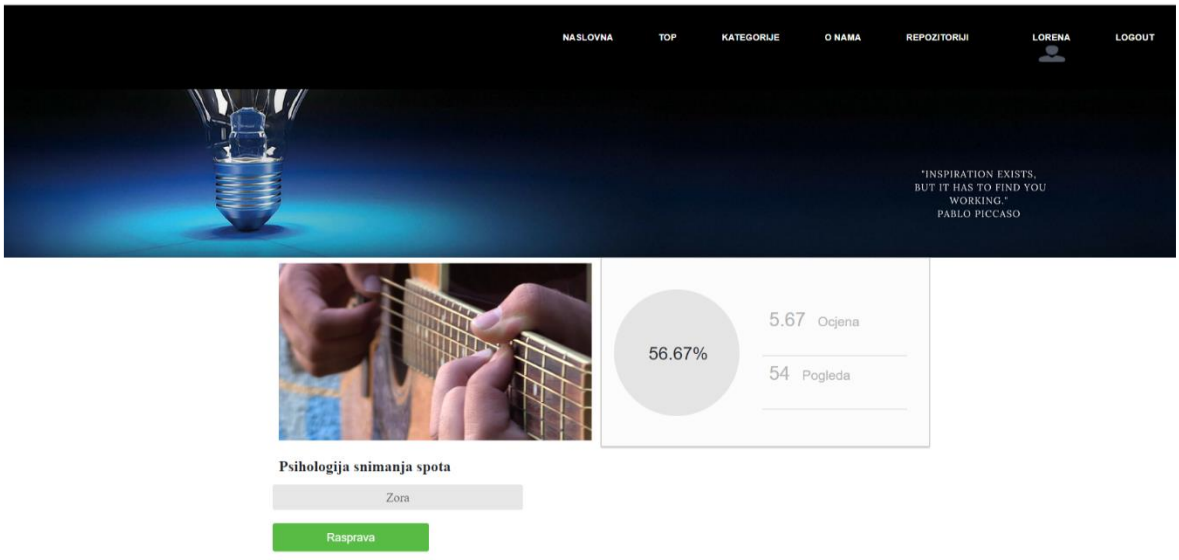
Slika 2.10 Kontakt obrazac za kontaktiranje autora video sadržaja

Kada korisnik pošalje elektroničku poštu drugom korisniku, njegova poruka ide putem SMTP protokola (*engl, Simple mail transfer protocol*). To je protocol koji je namjenjen za prijenos elektroničkih poruka, poruka sa servera odlazi na dotični mail registiranog autora video sadržaja, te svoju poruku dobiva na svoj email koji je se nalazi na jednoj od poznatih email platformi kao što su Hotmail, Gmail ili nešto drugo. Kako bi znali kome treba poslati email, odnosno kako bi aplikacija znala kome treba poslati email, potrebno je poznavati. Hotmail korisnika kojeg želimo kontaktirati. Hotmail informacije su dobivene na početku registriranja dotičnog korisnika, u trenutku kada se on registrirao i pružao informacije vezane za mail. Php kripta povratno prezentira takvu informaciju, kroz kontakt obrazac dok u sebi ima liniju koda koja odrađuje kontaktiranje korisnika kroz gotovu implementiranu funkciju u php-u email()[11].

[11] Email funkcija za kontaktiranje putem SMTP protokola

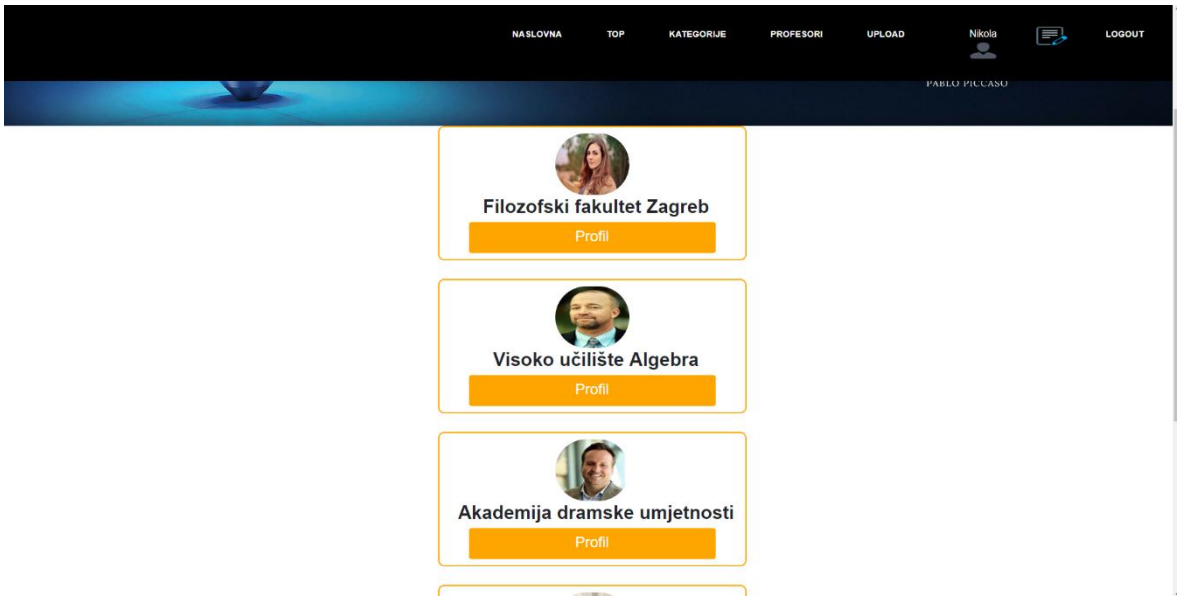
Super korisnici ili profesori su treća vrsta korisnika web aplikacije. Oni su posebno područje korisnika zato što se oni ne definiraju kao obični korisnici, nego kao korisnici koji mogu manipulirati video sadržajem stavljenim od strane dizajnera. Primarna ideja je bila da profesori koji bi u stvarnom smislu bili profesori raznih fakultetskih obrazovanja, zapravo doprinosili svojim statusom, iskustvom i znanjem. Oni bi sa svojim navedenim karakteristikama pridonosili boljem ocjenjivanju video sadržaja na temelju znanja koje su stekli tokom svog studiranja i daljnjeg poslovanja kasnije. Dizajner i profesori dijele poznavanje studentskog procesa učenja, razmatranja i dobivanja ideja i kako se te ideje implementiraju u uratke kroz znanje stečeno na fakultetu . Samim time se mogu naglasiti identifikacije između te dvije skupine korisnika koje se pojavljuju na web aplikaciji kao i njihov odnos koji je prikazan i puno stabilniji i razumljiviji između dvije strane. Također je naglašena lista profesora koja je vidljiva i dostupna svima, na toj listi korisnici se mogu dodatno informirati o profesorima. Na profilima profesora bi se najćešće nalazile informacije kao što su fakultet i obrazovanje, kojih su zanimanja, njihov životopis ili link na LinkedIn te ostali zanimljivi detalji iz životopisa koji bi odavali dozu orginalnosti i stručnosti prema dizajnerima i ostalim korisnicima koji su dio web aplikacije. Profesori ili super korisnici po glavnoj i primarnoj ideji bi bili 8 profesora iz 8 stručnih područja. Na (Slika 2.11) vidljiv je prikaz navigacijskog sučelja profesora.

.



Slika 2.11 Prikaz navigacijskog sučelja super korisnika

Na navigacijskom sučelju dizajnera (Slika 2.12) vidljiva je kartica profesori. Klikom na tu karticu Dizajner dobiva uvid u listu, čiji su članovi profesori „žiriji“ . Kada dizajner dobije uvid u komentare profesora, on može provjeriti dodatne informacije o profesorima klikom na karticu profesori. Rezultat toga je vidljiv na (Slika 2.12). Cijeloukupna lista svih profesora zaduženih za recenziranje video sadržaja, klikom na profil, korisnik odlazi na profil profesora na kojem istovremeno može kontaktirati profesora, prilikom čega odlazi na sučelje za kontakt, koje je identično kao na (Slika 2.10).

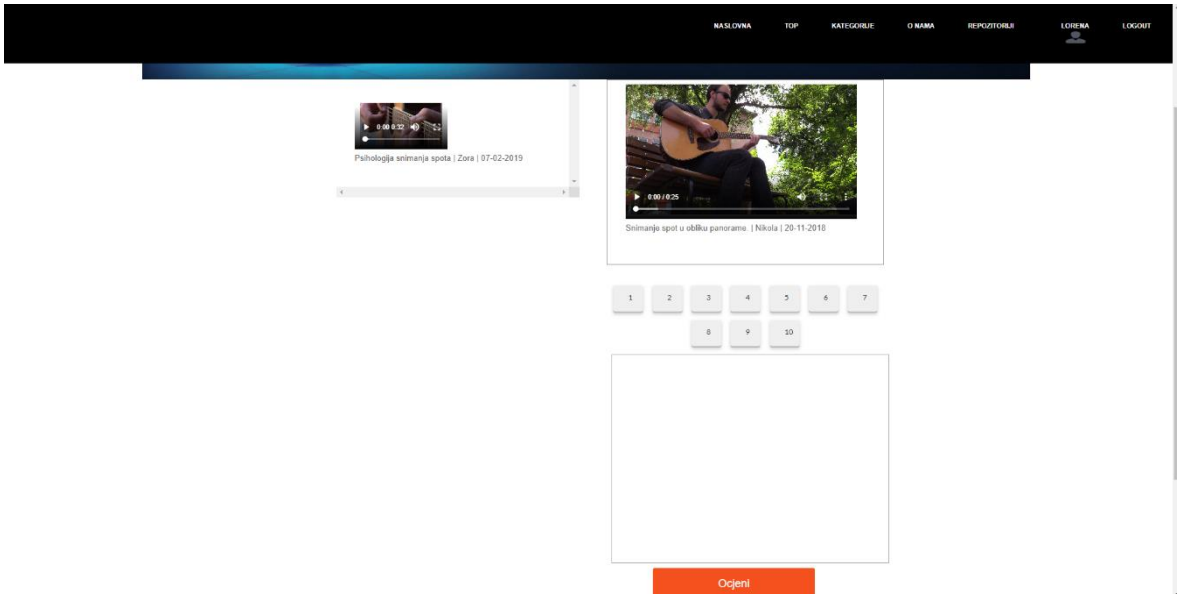


Slika 2.12 Prikaz liste profesora

Uloga ovog dijela sučelja web aplikacije jest da dizajneri kada dobiju komentare na svoj prvi video sadržaj, imaju mogućnost nakon toga vidjeti ime profesora koji je pružio odgovarajuću recenziju. Za ostale informacije moraju otići na listu profesora, potražiti odgovarajućeg usera i vidjeti dodatne informacije o njemu ili ga kontaktirati u slučaju nejasnoća ili sl. Ideja je na takav način kroz projekt približiti dizajnere sa ljudima koji ocjenjuju njihov sadržaj kako bi na taj način dizajneri dobili povratne informacije ljudi koji stručno mogu recenzirati njihov sadržaj. Tu dolazi do glavnog cilja i namjene ovog projekta da dizajner dobije stručno mišljenje te na osnovu njega pokuša optimizirati svoje ciljeve i ideje kroz svoj sljedeći projekt.

2.2.4. Recenzija i ocjenjivanje

Svaki od profesora ima pristup sučelju koji se naziva repozitoriji. Do sučelja profesori dolaze pomoću kartice koja je postavljena na navigacijskom dijelu. Repozitoriji je zamišljen kao nekakva vrsta CMS-a (*engl. Content Managment System*) ili sustava za upravljanje sadržajem. Kako mu i sam naziv glasi, to je sučelje ili dio web aplikacije na kojem profesori ocjenjuju video uratke stavljene od strane dizajnera (Slika 2.13). Na slici je vidljiv sustav za upravljanje sadržajem u sklopu web aplikacije. Lista sa lijeve strane prikazuje ne ocjenjene video sadržaje, koji su uplodani od strane dizajnera. Klikom na jedan od tih videa, video se prebacuje u prostor sa desne strane, prilikom čega je profesor spreman pregledati video. Nakon što je pregledao video, profesor se odlučuje za ocjenu od 1 do 10, te upisuje komentar i dojam unutar sekcije za komentare.Nakon toga, profesor klikne na gumb ocjeni nakon čega video sadržaj dobiva svoju ocjenu i status da je ocjenjen od strane profesora. Cijeli taj proces nakon klika na gumb "ocjeni" se događa jako brzo i transparentno, jer se video već odavno nalazi na web aplikaciji samo ga je potrebno ocjeniti, da se ostale informacije identificiraju i poslože u bazi podataka. Svaki komentar koji profesor upiše se prikazuje na komentar slajderu koji je vidljiv na (Slika 2.13).



Slika 2.13 Prikaz repozitoriji sučelja

3. Web aplikacija

Web aplikacija po svom sastavu kao aplikacija mora imati uvjet, a to je da bude dinamična. Web aplikacija u sklopu projekta ima sve karakteristike web aplikacije a to su:

- HTML I CSS
- MySQL BAZa PODATAKA
- SKRIPTNI JEZIK PHP
- STRUKTURU BAZE PODATAKA
- I FUNKCIONALNOSTI JAVA SCRIPTA I JQUERY-a

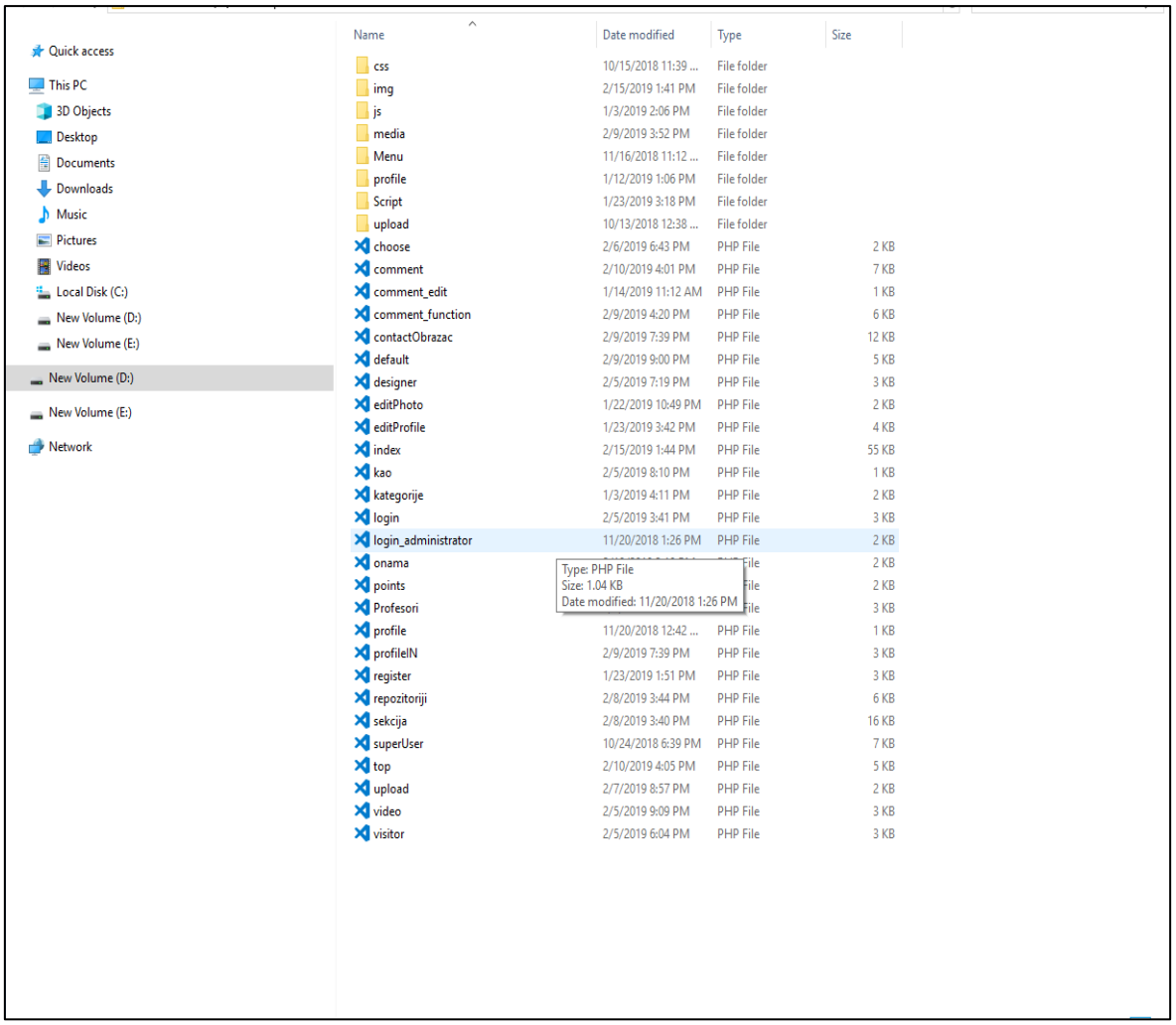
Primarno što je napravljeno u sklopu web aplikacije jest definirati strukturu informacija kroz bazu podataka, skicirati cijeloukupnu skicu web aplikacije kao stranice koja će kroz razna sučelja biti prezentirana korisnicima, definirati hijarahiju korisnika, njihovu povezanost unutar baze podataka i prezentiranje informacija iz baze podataka na web stranicu pomoću skriptnog jezika PHP. Web aplikacija sadrži:

- 38 php skripti unutar kojih se nalazi html
- 12 foldera
- 15 CSS fajlova
- 14 javascript fajlova
- 40 fajlova slika
- 9 tablica unutar baze podataka

Sve navedeno je sastavni dio web aplikacije i svaki od tih fajlova su više- manje međusobno povezani. PHP skripte sadrže unutar sebe ne samo php kod nego i html, css i jquery. Skripte služe za funkcionalne radnje na stranici što stranicu i čini web aplikacijom. Svaka od skripti ima povezanost između sebe. Svi ti fajlovi od kojih se sastoji web aplikacije se nalaze smješteni na web serveru. Glavni fajl i središte preko kojeg dolaze sve druge skripte po zahtjevu korisnika se naziva index.php skripta. Ona je smještena u primarni folder u kojem se nalazi i većina ostalih skripta. Web aplikacija ima skripte zadužene za login sučelje, za provjeru ulogiravanja, registraciju, postavljanje video sadržaja, komentiranje, promjenu navigacijskog sučelja ovisno o vrsti korisnika, brisanje i uređivanje komentara , ocjenivanje i komentiranje videa od strane super korisnika.

3.1. Organizacija projekta i raspored datoteka

Raspored datoteka u sklopu moje web aplikacije je napravljen vrstom logičkog redosljeda. Prvo što je trebalo napraviti jest smjestiti glavne php skripte koje se nalaze u glavnom folderu. Nakon toga potrebno je definirati primarnu php skriptu preko koje će se ostale skripte prikazivati. Takva skripta u IT području se po nekakvoj logici i standardu naziva index. Preko te skripte se uključuju sve ostale skripte koje se nalaze na istom mjestu ili u folderu. Uključivanje skripta se uključuje s obzirom na zahtjev korisnika, odnosno na zahtjev klika na jednu od kartica koje se nalaze na navigacijskom sučelju. Css fajlovi koji služe za raspored elemenata i njihovog dizajn se stavljaju u poseban folder, koji se naziva css. Na index skirpti postoji linija koda koja poziva css fajlove. Java script fajlovi također imaju svoje folder koji se naziv "JS" u kojem se nalaze java script fajlovi koji se na isti način pozivaju linijom koda unutar index skripte. Također postoji i media folder u kojem se nalaze slike i video sadržaji. Slike se postavljaju na ostale skripte ovisno o kojim se skritpama radi. Slike za razliku od css i javascripta se povezuju sa ostalim i pozivaju se iz ostalih skripta, jer nisu ovisne o index skripti. Kada index skripta poziva neku drugu skriptu ona također poziva i skriptu unutar koje se nalaze individualni pozivi slika i njihovog sadržaja koji se nalazi u folderu media. Index skripta je zapravo „srce" web aplikacije u kojoj se sadržavaju svi ostali pozivi na druge vrste programskih jezika koji pomagaju u realizaciji funkcionalnost web aplikacije da bi ona postala dinamična u raznim područijima. Na (Slika 3.1). je prikazan raspored datoteka u sklopu projekta. Kao što je vidjivo većina fajlova se sastoji od php formata, također su vidljivi i ostali folderi koje nisam naveo. Folder po nazivom "Script" sadrži ostale skripte. Jedna od tih skripti koja se nalazi tamo jest skripta checklogin o kojoj će se kasnije raspravljati. Vidljiv je folder „"Menu" u kojem se nalaze zasebne skripte koje sadrže navigacijsko web sučelje za svakog tipa korisnika posebno, posjetitelj, dizajner i super korisnik. Takve skripte kao i sve ostale koje su vidljive na slika (Slika 3.1). Pozivajući preko index stranice Upload direktoriji predstavlja folder u kojem se nalaze video sadržaji postavljeni od strane korisnika. U trenutku kada korisnik pošalje video sadržaj taj sadržaj odlazi u dotični folder. "Img" folder je folder u kojem se nalaze slike i dizajn koji se sastoji od web aplikacije. U profile folderu se nalaze dodatne skripte i dodatni kodovi koji služe za responzivnost i dinamiku profila korisnika.



Slika 3.1 Prikaz rasporeda datoteka

Skripte koje je potrebno automatski povezati s index skriptom, koje su uvjet da bi se ostale funkcije odrađivale na index skripti je potrebno automatski povezati kao što je vidljivo na (Slika 3.2). One postaju dio index skripte i nije ih potrebno regulirati i dodavati glavnim kodom za pridurživanje skripti koji je vidljiv na (Slika 3.2). Ovaj dio koda pridodaje povezanost na bazu i pridruživanjem cheklogin skripte koja je jako bitna i koja provjerava korisničke podatke korisnika i regulira njegovo logiranje.

```
1
2   <?php
3
4
5   include("script/db_connection_show_me.php");
6   include("script/checklogin.php");
7
8
9
```

Slika 3.2 Prikaz php koda koji obavlja automatsko pridodavanje skripti

Najvažniji mehanizam php koda koji je implementiran na index skirpti je prikazan na (Slika 3.3). Taj kod zapravo organizira i pridružuje skripte ovisno o njihovom zahtjevu. Skripte se pozivaju kodom prikazanim na (Slika 3.2). Skripte se samo prodružuju u trenutku kada korisnik na web aplikaciji pošalje zahtjev, one nisu su dio index skripte cijelo vrijeme kao što su skripte prikazane na slika (3.2). One su dio index skripte kada ih se pozove i dok se ne zatraži drugi zahtjev. U liniji 19. je kroz if uvjet definirana globalna varijabla page koja ako je postavljena ili ako je nejednaka odnosno ne postavljena, u liniji 21. je iz tog razloga napisano da se postavi. Drugim riječima postavili smo primarni uvjet da stvorimo globalnu varijablu page unutar obične varijable. U liniji 24. else uvjet odrađuje sljedeće: ako nema zahtjeva za određenom skriptom onda neka zahtjev bude primarna skripta (*engl, default*) i neka prikaže primarnu skriptu koja prikazuje naslovnu stranicu web aplikacije uključujući video sadržaj i statistiku. Nakon što je sve to odrađeno i nakon što su stavljeni početni uvjeti za pridruživanje skripti sa zahtjevom dolazimo na liniju 27. u kojoj se definira varijana file u kojoj se varijabla page lektorira pridodavanjem nastavka .php, kako bi varijabla mogla u sebi imati infromaciju da ispravno pozove skriptu nastavkom php jer je to primarni uvjet da se skripta pravilno pridruži. Ako je sve dobro napravljeno i ako je skripta koja se dodaje preko file varijable pravilna od linije 29 . pa do linije 36 će se skripta pravilno prikazati korisniku s obzirom na njegov zahtjev, odnosno php kod će ga pravilno prikazati na liniji 27.

```
18  <?php
19      if(isset($_GET["page"]) && $_GET["page"] !=""){
20
21          $page = $_GET["page"];
22
23      }
24      else {
25          $page = "default";
26      }
27      $file = trim($page).".php";
28
29          if(file_exists($file)){
30
31              $include2page = $file;
32          }
33          else {
34
35                  $include2page = "default.php";
36          }
37
38
39  ?>
```

Slika 3.3 Prikaz php koda koji služi za pridruživanje skripti

3.2. Bitni programski elementi

Programski elementi koji su jedni od bitnijih dolaze iz svih 5 područja skriptnih jezika koji su navedeni na prethodnim stranicama. Jedan od bitnijih programski elemenata je vidljiv i na prijašnjim slikama a to je query[12] . Query ili $query je varijabla koja je dio php skriptnog koda i koja u sebi sadržava funckiju koja funkcionira na način da pošalje zahtjev prema bazi. On je bitna značajka prilikom prezentiranja video sadržaja od strane korisnika. Varijabla query nam je potrebna kako bi poslali zahtjev bazi da nam izlista informacije vezane za njegov video sadržaj. Sve to radimo pomoću query koji u sebi sadrži sql naredbe, te naredbe kroz query idu u bazu podataka zahvaljujući php sql funkcijama. Funkcije koje su najviše korištene za vrijeme izrade web aplikacije i koje su bitan programski element su funkcije php skriptiranog jezika a one su

- mysqli_query
- mysqli_fetch_array
- $_SESSION
- HTML

Mysqli_query je funkcija koja služi za slanje query varijable u bazu podataka, dok mysqli_fetch_array funckija služi za izlistavanje redaka i stupaca iz baze podataka. Sljedeći bitni programski element jest sesija. Ona se označavala oznakom $_SESSION [13]globalna varijalba koja se identificirala s podatcima korisnika. Pomoću nje je definirano koji je trenutno korisnik ulogiran na web aplikaciju. Također, takva globalna varijabla omogućila je izlistavanje svih podataka istog korisnika koji su bili bitni za prezentaciju. Pomoću $_SESSION varijable je definiran profil i napravljen način na koji korisnik upravlja profilom. <HTML> kao sljedeći bitni programski element pomoću kojeg je prezentiran sadržaj kroz web sučelje korisnicima. On je glavni i temeljni dio svake web aplikaije neovisno o kojem se prednjem (*engl, frontend*) programskom jeziku radi.

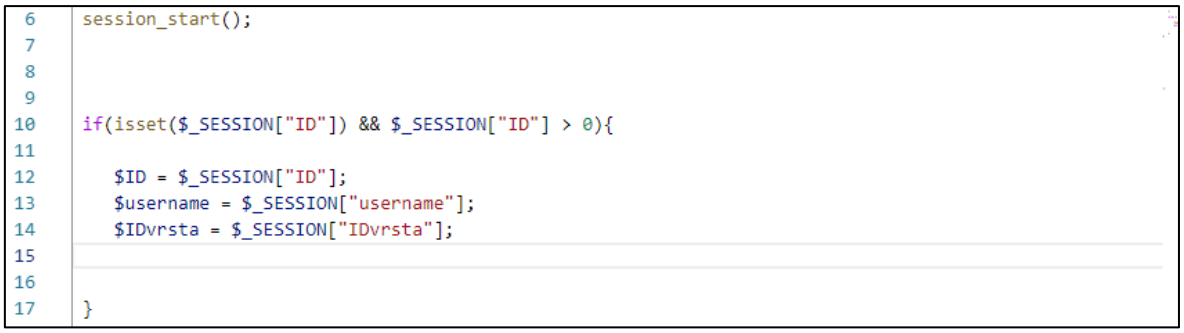
[12] Upit koji šaljemo u bazu sa definiranim zahtjevom
[13] Globalna php varijabla takozvana sesija.

3.3. Sesija

Sesija je jedan od značajnih dijelova php programiranja. Sesija je općenito prostor ili domena koja regulira protok informacija na osnovu globalne varijable $_session. Ona služi za definiranje i razlikovanje korisnika koji se registriraju i ulogiravaju na web aplikaciju. Sesija je vrsta regulatora, koji se izjednačava sa definiranim informacijama iz baze podataka. Naprimjer, kada korisnik upiše svoje korisničke podatke u registar obrazac, nakon što se počme ulogiravati sa istim informacijama na stranicu, globalna varijabla sesija se izjednačava sa varijablama koje sadrže primarne informacije o korisniku. U trenutku kada se ta dva tipa informacija izjednače, korisničke informacije strujaju kroz cijelovitu web aplikaciju izjednačene unutar sesije kako bi se provjerio korisnik i definirao kroz globalnu varijablu $_session. To je jedan od bitnih fatkora, jer se na osnovu tog obavljaju bitne funkcije unutar web aplikacije kao što su:

- Prikaz korisnika u navigacijskom sučelju
- Idetifikacija korisnikovih komentara
- Identifikacija video sadržaja po njegovom autoru.
- Identifikacija korisnika kroz sučelje sesije

Za vrijeme izrade završnog rada, sesija je definirana u skripti checklogin u kojoj se nalazio cijelovit php kod sa globalnim varijablama sesije, koje su provjeravale identifikaciju korisničkog imena, i šifre prilikom procesa ulogiravanja korisnika. Treći parametar koj je bitan kod sesije jest ID. Prilikom izrade checklogin skripte definiran je ID kroz globalnu varijablu $_session kako bi se na osnovu tog session ID-a varijabla izjednačila sa ID-om varijable zasebnom za sebe. Prvo i najbitnije jest da ako želimo koristiti sesiju, moramo definirati njezin početak rada.Taj početak ide na vrh glavne skripte index.php, zbog toga jer je to skripta na kojoj se pravi prva provjera informacija korisnika i zbog toga jer je index skripta srce web aplikacije na kojoj pridolaze u ovom slučaju sve ostale skripte ovisno o zahtjevu korisnika. Prikaz početka rada sesija i njenog uključivanja je prikazan na linji 6. Slika 29. Nakon toga u linji 10 prikazan je if uvjet koji kaže: „ Ako je varijabla $_SESSION ID definirana ili ako je veća od 0 onda mi izjednači običnu varijablu ID sa globalnom varijablom sesije ID“. Sesija identificira, provjerava i izjednačava ID sesije i ID obične varijable korisnika vidljiv na (Slika 3.4) linji 12.



Slika 3.4 Prikaz početka rada sesije i njezinog definiranja

Na liniji 12. i liniji 13. vidljivo je isto, prilikom čega se i izjednačavu varijabla korisničkog imena (*engl,username*) i vrste korisnika. Sada su te obične informacije jednake onome što će se poslje pohraniti u varijablu sesije. Sada kada su te informacije izjednačene običnim varijablama, možemo krenuti u sljedeći korak u kojem ćemo varijablama sesije dati bitne informacije. Na Slika 28 prikazan je kod koji odrađuje izjednačavanje informacija sa drugim tipom varijabli. To su informacije korisnika koje se nalaze u bazi podataka, sljedeće prazne varijable $username_form i $password_form sada dobivaju prave informacije spremljene u $_POST varijabli, koja služi za pohranjivanje informacija u sklopu login forme. Tu spadaju korisničko ime i šifra. U liniji 20. je vidljiv if uvjet koji govori :“ ako je kliknut gumb login, onda mi novonastale varijable izjednači sa vrijednostima koje sadrži post varijabla. Sada kada username_form i password_form imaju vrijednosti pravih informacija radimo sljedeći korak koji je vidljiv na liniji 30. Šaljemo upit i selektiramo sve informacije koje imaju korisničko ime slično onome koje je došlo sa login forme, na taj način dobivamo informacije korisnika koji se ulogirava. Upit šaljemo kroz mysqli_Query funkciju u sql na liniji 32 te zatim na liniji 34. Pomoću funkcije mysqli_fecth_array definiramo stupce i retke uvjetom sličnog korisničkom imena sa login forme i dolazimo do ključnog trenutka. „Ako je šifra korisnika iz baze podataka koji se želi ulogirati jednaka šifri sa login forme, onda mi tu istu šifru iz baze podataka izjednači sa ID-om iz baze, korisničko ime sesije i vrstu izjednači mi korisničkim imenom i vrstom iz baze“. Odgovarajućim kodom smo zapravo pohranili u sesiju informacije korisnika pristigle iz login forme i povjerene kroz upit i šifru korisnika. Sada naša sesija koja je pristupna na bilo kojoj skripti zahvaljujući kodu na (Slika 3.4).sadrži u sebi informacije korisnika, unikatna informacijama korisnika koji je ulogiran i nemoguće je za promjeniti. Sada web aplikacija identificira korisnika koji se prijavi na stranicu i na osnovu toga daje sve potrebne funkcionalnost bazirane na njegovim informacijama. Ako su podatci pogrešni, odnosno ako korisnik upiše krive podatke u login, checklogin provjera će

biti negativna i kod će odmah prebaciti na else uvjet na liniji 52. i vratit korisnika ponovno na login sučelje.

```
20  if(isset($_POST["btn_login"])){
21
22
23
24      $username = $_POST["username"];
25      $password = $_POST["password"];
26
27      $username_form = trim($username);
28      $password= trim($password);
29
30      $query = "SELECT ID, username, password, IDvrsta  from users where username like '$username'";
31
32      $result = mysqli_query($con, $query);
33
34      $user = mysqli_fetch_array($result);
35
36      if($user["password"] == $password){
37
38
39          $_SESSION["ID"] = $user["ID"];
40          $_SESSION["username"] = $user["username"];
41          $_SESSION["IDvrsta"] = $user["IDvrsta"];
42
43      }
44
45      else {
46
47          header("Location: ../login.php");
48          exit;
49
50
51      }
52  }
```

Slika 3.5 Prikaz sesije i njezine provjere

Svi koraci koji su objašnjeni na prethodnim stranicama su zapravo početno konfiguriranje sesije na web aplikaciji. Prava namjena i svrha sesije je prikazana u kasnijim radovima na aplikaciji. Posebno bitno je za naglasiti da pomoću informacija koje smo spremili u sesiju možemo lako i jednostavno manipulirati sadržajem iz baze podataka u trenutku kada se korisnik ulogira i lakše ostvarivati ideje u sklopu projekta. Sesija na nekin način metaforčki gledano je policajac svake web aplikacije, bez kojeg web aplikacija ne može funkcionirati ako želi biti dinamična. Kako bi prikazivali sadržaj koji je povezan sa korisnikom, upravu je to svrha sesije i njezine globalne varijable $_SESSION. Na (Slika 3.5) tipičan primjer upotrebe varijable koja je izjednačena sa Sesijom. Varijablu $Idvrsta smo izjednačili sa varijablom sesije, varijabla sesije pri izjednačavanju je sadržavala informacije iz baze podataka. Tako smo instancirali taj sadržaj u običnu varijabu $Idvrsta prikazanu (Slika 3.6), Linija 14. Pošto je checklogin skripta automatski pridodana na index skriptu to znači da je ona jednim dijelom index skripte tako da nije potrebno ponovno definirati varijablu. U liniji 2417. je vidljiv upit u kojem tražimo korisnika kojemu je vrsta identična vrsti smještenoj u sesiji, odnosno korisnika koji je trenutno ulogiran. U liniji 2421. Vdljiva je varijabla user

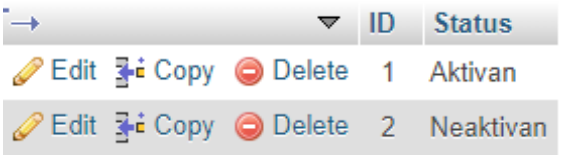
koja u sebi sadrži vrstu korisnika identičnoj vrsti koja se nalazi u sesiji, odnosno korisniku koji se trenutno ulogirao. If uvjet kaže „ ako je vrsta definirana po vrsti korisnika koji je ulogiran jednaka 3 onda mi uključi skriptu koja sadrži navigacijsko sučelje namjenjeno i napravljeno za super korisnika. Ako je vrsta ulogirano korisnika 2 to znači da će se pridodati navigacijska skripta za sučelje dizajnera, ako je 1 pridodat će se navigacijska skripta sučelja za posjetitelja. Ako u nekom s slučaju se nemože identificirati broj vrste, nemoj dozvoliti da korisnik pristupi bilo kojem sadržaju“. Većina ostalih funkcionalnosti na web aplikaciji funkcioniraju na ovakvom principu, jer je cijelo vrijeme omogućeno orijentiranje pomoću sesije kao objekta i vrijednosti koje su instancirane u njoj.

```
2410
2411                $query= "SELECT IDvrsta FROM users where IDvrsta like '$IDvrsta'";
2412                $result = mysqli_query($con, $query);
2413                $user = mysqli_fetch_array($result);
2414
2415                if($user["IDvrsta"] == '3'){
2416
2417                    $include_profesor = include("menu_profesor.php");
2418
2419
2420                }
2421                elseif ($user["IDvrsta"] == '2'){
2422
2423                    $include_designer = include("Menu/menu_designer.php");
2424
2425                }
2426
2427                elseif($user["IDvrsta"] == '1'){
2428
2429                    $include_visitor = include("Menu/menu_visitor.php");
2430                }
2431
2432                else {
2433
2434                    header("Location:../login.php")
2435                }
2436
```

Slika 3.6 Prikaz dodavanja sučelja navigacije uvjetom sesije

3.4. Repozitoriji i njegove funkcionalnosti

Repozitoriji se temelji na jednostavnom principu postavljanja video uratka na web server od strane korisnika i primanje tog istog uratka u repozitoriji. Repozitoriji u sklopu mog završnog rada je međuprostor između dizajnera i profesora. Korisnik kada postavi sadržaj na web stranicu, taj isti sadržaj se pojavljuje super korisnicima na njihovom web sučelju . Takav proces funkcionira na sljedeci nacin. Korisnici postave video sadržaj na server, koji sa svojim informacijama kao što su: naziv, kategorija, opis, format i vrsta tijekom postavljanja slanjem, odlaze u bazu podataka i takve informacije zajedno sa video uradkom pohranjuju unutar baze . Prilikom postavljanja videa od strane korisnika, skripta koja je napisana se automatski pohranjuje vrstu definiranu brojem 1. Što znači da je video neaktivan, kada je video neaktivan on nije prikazan na naslovnoj strani jer to znači da taj isti video nije ocjenjen. Upravo zbog tog uvjeta, video sadržaj se nalazi u repozitoriju, koji je vidljiv super korisnicima upravo zbog njegovog uvjeta 1. U bazi podataka je definirana tablica koja se naziva video_status i koja se sastoji od dva stupca. Prvi stupac prikazuje ID status, drugi stupac je stupac koji se naziva status. Video_status tablica je definirana po ID statusu. Ako je Idstatus 1 to znači da je video aktivan , ako je definiran IDstatus 2 to znači da je video neaktivan. Pomoću tablice se na taj način preko skripte s uvjetima ako je video Idstatusa jedan video se postavlja kao aktivan i prikazuje se na naslovnici, ako je video 2 statusa 2, video se neće prikazati na naslovnici. Da bi takva jednostavna logika i uvjeti funkcionirali, tablica status mora biti povezana s još jednom tablicom koja se naziva video. Upravo tablica video u sebi sadrži strani ključ koji se naziva statusID i koji je povezan s tablicom video_status. Ako je u video tablici statusID jedan to znači da je aktivan kako je definirano u tablici video_status, ako u statusID u tablici video piše dva onda je video neaktivan. Prikaz tablice video_status (Slika 3.6). Na (Slika 3.7) vidljiva je tablica video u kojoj se jasno vidi statusID i njegova poveynica sa tablicom video_status.



Slika 3.7 Prikaz tablice status

ID	statusID	1	title	format	filename	description	size	path	time	userID	Kategorija
4		2	Snimanje spot u obliku panorame.	MP4	borna.MP4	Lorem Ipsum is simply dummy text of the printing a...	52225386	../media	2018-11-20 14:18:31	6	Umjetnost
32		2	Psihologija snimanja spota	MP4	gitara.MP4	Psihologija snimanja spota	93583470	../media	2019-02-07 16:50:41	16	Psihologija

elected: Edit Copy Delete Export

Slika 3.8 prikaz tablice video

Na (Slika 3.8) prikazan je video sadržaj u trenutku kada su video sadržaji postavljeni od strane dizajnera. Oni su neaktivni i takvi su definirani na početku. Na skripti koja se naziva upload se nalazi dio koda koji automatski, koji ima proces postavljanja video sadržaja u tablicu video. Prilikom tog postavljanja i spremanja video sadržaja u tablicu video, kroz php skriptu je definirano da u upitu status bude definiran kao dva što je vidljivo na slika 31. Na (Slika 3.9) cijeloukupan kod prikazuje slanje upita prilikom odrađivanja procesa postavljanja video sadržaja na web server. Varijabla upit (*engl, query*) sadrži upit koji govori: „ Stavi u tablicu video, sadržaj koji je definirana po stupcima, vrijednosti (*engl, values*). Varijable koje se nalaze u vrijednostima su varijable definirane vrijednostima koje dizajner sam postavi na sučelju upload koji je objašnjena na prethodinm stranicama. U tim vrijednostima se nalazi i definiran status dva, koji nam govori da je video nekativan sam od sebe prilikom postavljanja jer nije ocjenjen. Kada su definirane sve vrijednosti i kada je definiran status sadržaja, upit query koji je postavljen u varijabli rezultat (*engl, result*) kroz funkciju mysqli_query odlazi u bazu podataka linija koda je vidljiva na liniji 35.

```
28
29          if(move_uploaded_file($tmp_name, $new_name)){
30
31            $query =" INSERT INTO  video (ID, statusID, title,  format, filename, description,  size, path, time, userID, Kategorija)
32              VALUES(null ,'2', '$title',  '$ext',  '$name',  '$description', '$size', '$path', NOW(), '$ID', '$kategorija' )";
33
34
35              $result = mysqli_query($con, $query);
36
```

Slika 3.9 Prikaz spremanje video sadržaja kroz upit u bazu podataka

(Slika 3.9) prikazuje dio koda koji se nalazi u skripti upload i dio koda koji nije dio repozitorija, ali pomaže dijelu koda repozitorija da pomoću status 2 koji definira neaktivnost, na pojednostavljen način prikaže video sadržaje postavljene od strane korisnika. Na (Slika

3.10) prikazan je cijeloukupan kod koji prezentira video sadržaj stavljen od strane dizajnera u sučelje koje se na repozitoriju nalazi na lijevoj strani kao što je prikazano na (Slika 2.14). Prikaz koda koji počinje u linija 14. (Slika 3.10) funkcionra na način uvjeta status koji je definiran na (Slika 3.7).

```
6   echo '
7       <div class="container" >
8       <div class="row">
9         <div class="col-xs-12 col-sm-12 col-md-12 col-lg-6 text-left">
10
11          <div class="container-prvi">
12           ';
13
14          $query = "SELECT video.ID, video.filename, video.title, video.time,  video.description,
15          video.Kategorija, users.username from video, users
16            where video.userID = users.ID and statusID = '2' ";
17
18        $result = mysqli_query($con, $query);
19        while($row = mysqli_fetch_array($result)){
20
21          $username = $row["username"];
22          $title = $row["title"];
23          $kategorija = $row["Kategorija"];
24          $filename = $row["filename"];
25          $ID = $row["ID"];
26          $description = $row["description"];
27          $time = $row["time"];
28
29          $newDate = date('d-m-Y', strtotime($time));
```

Slika 3.10 Prikaz php koda koji izlistava video sadržaje s obzirom na njihov status

Na (Slika 3.10). prikazan je kod koji izlistava video sadržaj postavljen od strane dizajnera. U liniji 14. prikazan je upit koji selektira sve informacije iz tablice video, uključujući i video status, koji se automatski postavlja uz video sadržaj koji su postavili korisnici, pomoću toga se sortiraju video sadržaji statusa dva i ispisuju u repozitoriji. Unutar istog upita prikazan je i upit u tablicu users koja sadrži informacije o autoru kao što su korisničko ime. Upit vraća informacije iz obje tablice zahvaljujući poveznici između te dvije tablice koji je ostvaren stranim ključem userID i users ID-om. To je vidljivo na liniji 15. U liniji 19. kod pomoć while funkcije i $row varijable ispisuje stupce i redke iz tablice video iz baze podataka. Svi ti stupci su definirani varijablom $row koja se separira dijeli i definira u posebne varijable sa značenjem kao naprimjer u liniji 21. Varijabla $username sa $row varijablom koja prikazuje stupce za korisničkim imenom. $row varijabla korz while funkciju zapravo prikazuje cijelovitu tablicu baze podataka kroz web apliakciju dok god u njoj ima sadržaja. Ona sadrži svaki dio tablice, a upravo while funkcija u liniji 19. Koja sadrži pod funkciju mysqli_fetch_array, dohvaća sve te redove i stupce definirane u varijabli $row. Sa dodatnim varijablama izjednačavamo vrijednosti koje nama trebaju. Unutar while funkcije se nalaze

html elementi i div sekcije u kojima se nalaze nazivi video sadržaja, imena autora i vrijeme kad su video sadržaji uplodani. Sve te informacije se izlistavaj unutar while petlje. Na (Slika 3.11). Prikazane su varijable smještene u sekcije koje sačinjava html kod, to se na kraju prezentira super korisniku.

```
36  echo'
37
38              <div id="video-ocjena">
39              <video width="420px" height="300px" class="video_neocjenjen" controls>
40                <source src="media/'.$filename.'" type="video/mp4">
41                <input type="hidden" name="ID" value="'.$ID.'">
42                <input type="hidden" name="username" value="'.$username.'">
43                <input type="hidden" name="Kategorija" value="'.$kategorija.'">
44
45                  <source src="media/borna.ogg" type="video/ogg">
46
47                  </video>
48
49
50
51                  <p id="mali-opis"> '.$title.' | '.$username.' | '.$newDate.'  </p>
52
53                </div>
54
55            ';
```

Slika (3.11) Prikaz html i php koda koji prezentira sadržaj na repozitoriju

Bitno je za naglasiti da je samo jedna sekcija html koda, while petlja ce ispisati html sekcija onoliko koliko video sadržaja ima u bazi. Ovo je jedna od korištenih primjera za izvlačenje informacije iz baze podataka i to je jedna od svrha php-a. Div koji ima svoj id „video-ocjena“ je sekcija video uratka, pomoću te oznake id (html taga koji nije povezan sa bazom) se manipulira videom da se prebaci na desnu stranu repozitorija za ocjenjivanje. Prebacivanje navedene sekcije je napravljeno pomoću jednostavnog jquery koda (Slika 3.12). Linija koda 146 označava kad se klikne na html sekciju oznake video-ocjena, starta se funkcija koja u sebi ima definiranu liniju koda 150. da dijete, odnosno element koji se nalazi odmah ispod sekcije video-ocjena bude stavljen u sekciju spremnik, odnosno desnu stranu repozitorja gdje je navedeno da klikom na video on tu dolazi. Također mu je definiran css stil za širinu i visinu u trenutku kad se prebaci. Sljedeca linija koda 157 označava da se tekst koji objašnjava da se video treba klikom prebaciti, mali opis ispod videe označen id „mali-opis“ također se u sklopu videa pomiče u spremnik sekciju.

```
137
138         $(document).ready(function(){
139
140             $("#video-ocjena").click(function(){
141
142
143
144              $("#video-ocjena").children(".video_neocjenjen").appendTo("#spremnik").css({
145
146                  width:"500px",
147                  height:"300px"
148
149               });
150
151               $(".ovdje").css("display", "none");
152             $("#mali-opis").appendTo("#spremnik").css("font-size", "1.1em");
153
154
155
156             });
157     });
```

Slika 3.12 Prikaz jquery koda za prebacivanje videa

3.5. Baza Podataka

Baza podataka je skupina informacija, koja u sebi sadrži strukturu informacija koje su međusobno povezane. U IT-u baza podataka je jedan od ključnih faktora bez kojeg niti jedna dinamična aplikacija nebi mogla biti dinamična. Baza u sebi sadrži veliki broj tablica koje se tvore od redaka i stupaca. Svaka tablica ima informacije namjenjene ideji i cilju samog projekta. Dva glavna pojma koja su ključna što se tiče baze podataka u računalnom području su:

- Primarni ključ (*engl, primary key*)
- Sekundarni ključ (*engl, foreign key*)

Program koji prezentira bazu podataka u IT području se naziva mySQL. Sql je program otvorenog koda koji je napravljen kako bi se lakše i pratikčnije upotrebom sql koda gradile tablice u bazi podataka i povezivale s obzirom na njezine ključeve. U sklopu obrane završnog rada, jedna od bitnijih značajki je bila baza podataka. Prva i temeljna skica koja je bila na početku ovog projekta je bila upravo skica baze podataka. Pomoću te skice raspoređena je i sagrađena strukturu informacija koje bi sadržavala moja web aplikacija. Svaka baza podataka koja se radi na temelju nekakve web aplikacije sa budućim korisnicima treba sadržavati tablicu u kojoj će pohranjivati podatke svojih korisnika. U takve tablice u većini slučajeva se pohranjuju podatci kao što su:

- Korisničko ime
- Elektronička pošta
- Šifra korisnika
- Ime
- Prezime

Ovisno o zahtjevima aplikacije i o strukturi baze podataka, ali u većini slučajeva ovo su glavni podatci koji bi se trebali smještati u tablicu korisnika. Tablica korisnika je uvjek povezana s nekom drugom tablicom i na osnovu tablice korisnika bi se trebale izvršavati većina glavnih funkcionalnosti na stranici kao što su sesije web aplikacije koje su objašnjenjene na stranici 38. Uz korisničku tablicu u većini slučajeva potreban je i strani ključ koji će povezati i stvoriti glavne funkcionalnosti s ostalim tablicama tipa u mom slučaju video na temelju čega ćemo dobiti poveznicu između korisnika koji su uplodali video, identificirane na temelju stranog ključa koji je smješten u tablicu video i koji apelira na id korisnika koji je taj isti video stavio na web aplikaciju. Strani ključ tablice korisnika se

definira kao korisnikID, strani ključ se proizvoljno može definirati na boli koji način, ali u programerskom žargonu je kodeks ili nekakva vrsta primjerenog pisanja da se strani ključ piše na način da se ID(*engl, identification*) stavlja a kraju. Da remiziram, korisnikID je strani ključ koji ide u drugu tablicu, koja sadrži druge podatke, dok korisnikID služi za poveznicu sa tablicom korsnik. Na (Slika 4.1) prikazana je tablica korisnik u sklopu završnog rada. Ona se može uzeti za primjer općenitog definiranja korisnika u većini sličnih projekata.

ID	username	email	ime	prez	password	mobitel	oib	IDvrsta
6	Nikola	nikola@hotmail.com	Nikola	NikiÄ‡,	travian1223	2147483647	2147483647	2
8	Marko	marko.antunovic@hotmail.com	Marko	Antunovic	antunovic123	974435662	1241412424	3
9	Lorena	lorena.zarkovic@hotmail.com	Lorena	Zarkovic	zarkovic123	985673242	931424145	3
16	Zora	zora@hotmail.com	Zora	hecimovic,	travian123	2147483647	2132131	2

Slika.4.1 Prikaz tablice korisnik

Kao što je vidljivo na Slika (4.1) poszoji stupac ID, čijem su brojem označeni korisnici. Također tu je vidljiv Username, email ime i prezime. Šifra također sadrži tekstualni oblik, mobitel i oib. ID vrsta predstavlja stupac koji definira koje vrste je korisnik, kao što je navedeno u prethodnim poglavljima. Je li on dizajner, obični korisnik ili profesor. U ovom slučaju na (Slika 4.1) postoji korisnik Nikola po vrsti i registraciji dizajner dok je korisnik Lorena pod Idvrsta broja tri profesor. Na takav način se radi autmatska separacija korisnika, definiranje korisnika i razvrstavanje u vrstu korisnika koja je definirana u projektu. Ova tablica je bitna jer na startu radi filter korisnika koje bih kasnije mogao definirati video uratke koji se uplodaju i video uratke koji su ocjenjeni od korisnika profesori. Svi realcijski odnosi sa stranim ključem u drugim tablicama koji se povezuju na ovu stranicu idu preko ID-a. Ako je tipa strani ključ u tablici Video userID on se povezuje na ID tablice korisnik i takva relacijska poveznica se definira na osnovu identifikacijskog broja tablice korisnik. Tablica korisnik je u većini slučajeva povezana sa više tablica, na osnovu tablice korisnik, se definiraju drugi tipovi podataka i tablica kao što su mjesto, rođenje i sl. Bitno je za naglasiti da se ne stavlja puno podataka o korisniku u tablicu korisnik jer to može naškoditi daljnjem razvoju strukture tablice i dovesti do komplikacija funkcionalnosti baze podataka. Preporučava se da se naprave dodatne tablice i poveže ih se sa iznad navedenom tablicom.

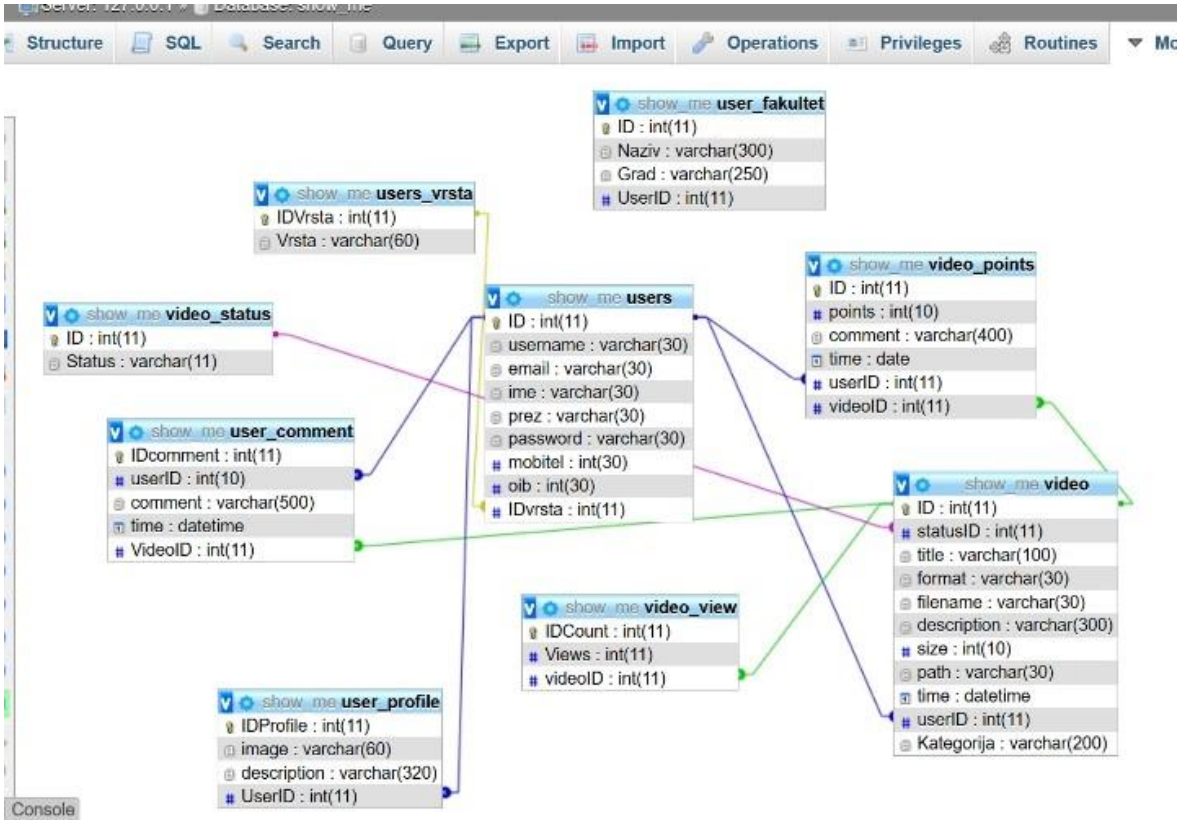
U sklopu mog završnog rada projekt je sadržavao 9 tablica. Glavna tablica je nazivala korisnici(*engl, users*) koju su se pohranjivali svi podatci trenutnih korisnika koji su

registrirani na stranicu koja je navedena u prethodnom poglavlju. Ostalih osam tablica su sljedeće:

- Users_vrsta
- Video_status
- User_comment
- Video_view
- Video_points
- Video
- User_fakultet
- User_profile

Korisnici i njihova vrsta tablica u kojoj je smještena vrsta korisnika. Video sa statusom (*engl,video status*) sadrži u kojem stanju se nalazi video. Korsnici komentari tablica (*engl,User_comment*) jest tablica koja sadrži u sebe komentare na video uratke i informacije o korisnicima koji su komentirali. Video pogledi (*engl,vide_view*) je tablica koja sadrži u sebi video poglede određenog videa koji je definiran ID-em . Video bodovi (*engl, video points*) je tablica koja sadrži bodove za svaki video i ID. Video tablica sadrži razne informacije o videima, kao što su vrijeme stavljanja videa na web aplikaciju, identifikacijski broj korisnika koji ga je stavio, vrijeme stavljanja, veličina videa, putanja videa, opis, format, naslov i kategorija. Sve na tablici video je označeno kojem korisniku odnosno dizajneru video pripada. Korisnik fakultet (*engl, user*) je tablica koja sadrži fakultete dizajnera. Korisnik profil (*engl, user profile*) je tablica koja sadrži profil korisnika, opis sliku i primarni i strani ključ korisnika. Na (Slika 4.2) vidljiva je tablica korisnik koja je dosad spominuta u prethodnom poglavlju. Tablica korsnik je povezana s tablicom video. Povezani su primanim ključem ID-om tablice korisnik i stranim ključom userID tablice video. To je relacijska poveznica u kojoj ID tipa korisnika 6. Nikola je ispisan isti broj šest u stupcu userID u tablici Video. Na takav način povezanostću primarnog i stranog ključa dobiva se uvid da je Nikola pod userID brojem šest ima sve ostale podatke video sadržaja .Na taj način osiguravajući da video pripada korisniku Nikola, (Slika 4.3) tablica korisnici povezana je sa tablicom points, na sličan način kao i sa tablicom video. Relacija između tablice bodovi i tablice korisnici se odvija preko ID-a tablice korisnika i userID. Na taj način je poznato čiji video je ocjenjen s koliko bodova, drugim riječima potznat je autor videa koji je dobio x bodova. Na tablici video je vidljiva povezanost između tablice video i tablice video pregledi. Povezani su

identifikacijskim brojem video na koji se tablica pregled referencira. Kad bi otišli u tablicu video pregledi vidjeli bi identifikacijski broj i broj pregleda videa. Sa videoID stranim ključem u tablici video pregledi bi znali prepoznati koji je to video a kasnije i čiji je autor videa. Tablice korisnik profil i korisnik koment su povezane s tablicom user na isti način na koji je povezana i tablica video. Referencirajući se stranim ključevima userID na ID tablice korisnik dobivam uvid koji korisnik pod kojim ID-om je komentirao video sadržaj i koji korisnik pod kojim ID-om ima napravljen profil. U tablici korisnik komentar imamo poveznicu VideoID odnosno strani ključ koji se referencira na ID tablice video kako bi dobili uvid koji komentar se odnosi na koji video definiran primarnim ključem odnosno idetifikancijskim brojem ID-em.



Slika 4.2 Prikaz strukture baze podataka

	ID	statusID 1	title	format	filename	description	size	path	time	userID	Kategorija
e	4	2	Snimanje spot u obliku panorame.	MP4	borna.MP4	Lorem Ipsum is simply dummy text of the printing a...	52225386	../media	2018-11-20 14:18:31	6	Umjetnost
e	32	2	Psihologija snimanja spota	MP4	gitara.MP4	Psihologija snimanja spota	93583470	../media	2019-02-07 16:50:41	16	Psihologija

Slika 4.3 Prikaz tablice Video

Zaključak

Tijekom izrade završnog rada stekao sam znanja u području dinamičkih web aplikacija, način na koji one funcioniraju te koje su stavke njihov neizostavni dio. Također, dobio sam uvid u odnos između skriptnog jezika PHP i baze podataka Mysql.

Korištenjem pet programskih jezika dosegnut je maksimum u stvaranju dinamike web aplikacije što je i bio cilj. Pomoću dinamike web aplikacije uspješno su povezana zamišljena tri tipa korisnika (korisnici, dizanjeri i profesori). Sve informacije o korisnicima i dizajnerovim video sadržajima su spremljene u bazu podataka koja na strukturiran i logičan način pridonosi lakšem segmentiranju podataka i prezentiranju na web stranici ili aplikaciji. Također, bitna odredba koja je bila od velike koristi i koja je neizostavni dio svake dinamičke aplikacije jest sesija pomoću koje je moguća identifikacija korisnika i omogućavanje statičnog statusa korištenja aplikacije. Postavljanje video sadržaja na web aplikaciju je praktičan pristup, osobito kada se pridodaju dodatni parametri i uvjeti kao što su podatci o samom autoru te vrsta i kvaliteta video sadržaja. Naravno, tu je i funkcionalnost samog sustava za upravljanje sadržajem čiji je cilj da na što bolji i jednostavniji način postigne dinamiku učitavanja video sadržaja od strane dizajnera na platformu web aplikacije. Svaka segmentacija video sadržaja mora sadržavati kategorizaciju kako bi se na što bolji i transparentniji način video sadržaji prezentirali korisniku. Podjela korisnika na tri vrste korisnika te gradiranje sadržaja se pokazalo dobrim odabirom stvarajući web aplikaciju unikatnijom što na kraju znatno pomaže u dobivanju stručnog i kvalitetnog mišljenja. Negativna strana takve ideje je što se stvara veća kompleksnost sustava i povećanje većeg broja informacija. Reduciranje kompleksnosti postignuto je stvaranjem strukture baze na pravilan i školski način kao i logikom odnosa između triju korisnika.

Izrada dinamičke web aplikacije zanimljiv je ali i zahtjevan projekt. Zadovoljan sam načinom na koji mi je dinamika omogućila realizaciju moje ideje, stvarajući web aplikaciju za recenziju i ocjenjivanje video sadržaja ozbiljnim projektom.

Popis kratica

CMS	*Content management system*	Sustav za upravljanje sadržajem
PHP	*Hypertext Preprocessor*	Skriptni predprocesni jezik
HTML	*HyperText Markup Language*	Prezentacijski jezik za izradu web stranica
CSS	*Cascading Style Sheets*	Stilski jezik
AVI	*Audio Video Interleaved*	Multimedijski video format
MP4	*MPEG-4*	Standard za zvuk
SMTP	*Simple Mail Transfer Protocol*	Protokol za prijenos električne pošte
ID	*Identity document*	Dokument iza identifikaciju
PDO	*Php data objects*	Php objekti

Popis slika

Slika 1.1 ..... **Pogreška! Knjižna oznaka nije definirana.**

Slika 1.2 ..... **Pogreška! Knjižna oznaka nije definirana.**

Slika 1.3 7

Slika 1.4 8

Slika 1.5 9

Slika 1.6 10

Slika 1.7 11

Slika 1.8 12

Slika 1.9 13

Slika 2.2 15

Slika 2.3 16

Slika 2.4 17

Slika 2.5 18

Slika 2.6 21

Slika 2.7 21

Slika 2.8 ..... **Pogreška! Knjižna oznaka nije definirana.**2

Slika 2.9 ..... **Pogreška! Knjižna oznaka nije definirana.**3

Slika 2.10 ..... **Pogreška! Knjižna oznaka nije definirana.**4

Slika 2.11 25

Slika 2.12 26

Slika 2.13 27

Slika 3.1 ..... **Pogreška! Knjižna oznaka nije definirana.**0

Slika 3.2 ..... **Pogreška! Knjižna oznaka nije definirana.**0

Slika 3.3 ..... **Pogreška! Knjižna oznaka nije definirana.**1

Slika 3.4 34
Slika 3.5 35
Slika 3.6 37
Slika 3.7 38
Slika 3.8 38
Slika 3.9 38
Slika 3.10 ..... **Pogreška! Knjižna oznaka nije definirana.**9
Slika 3.11 ..... **Pogreška! Knjižna oznaka nije definirana.**0
Slika 3.12 ..... **Pogreška! Knjižna oznaka nije definirana.**1
Slika 4.1 43
Slika 4.2 45
Slika 4.3 46

.

Literatura

[1] HTTPS://HR.WIKIPEDIA.ORG/WIKI/HTML 2.2.2019.

[2] https://hr.wikipedia.org/wiki/CSS 3.2.2019.

[3] http://www.netakademija.hr/sto-je-php/ 3.2.2019.

[4] https://en.wikipedia.org/wiki/PhpMyAdmin 5.2.2019.

[5] Php programiranje priručnik, Ivan Božajić

[6] https://www.php.net/

[7] https://www.w3schools.com/