

MyComicsList platforma

Olja Ivković 34/16
Visoka ICT škola
Web programiranje PHP 1

<https://mycomicslist.datapoint.rs/>

Beograd 2018

1.Uvod	3
1.1. Korišćenje tehnologije	3
1.2. Opis funkcionalnosti	3
1.3. Templates	4
1.4. Korišćene biblioteke	10
1.4.1. Select2 biblioteka	10
1.4.2. Chartist biblioteka	10
1.4.3. JQuery Knob biblioteka	10
2. Organizacija	10
2.1. Organizaciona šema	10
2.2. Šema baze podataka	11
2.3. Sitemap	11
2.4. Opis stranica sajta	14
2.4.1. Home page	14
2.4.2. Browse page	19
2.4.3. MyList page	21
2.4.4. Contact page	23
2.4.5. About me page	24
2.4.6. Admin panel – panel page	25
2.4.7. Admin panel – stats page	28
3. Kodovi	29
3.1. Root	29
3.1.1. index.php	29
3.2. Admin panel	31
3.2.1. panel.php	31
3.2.2. stats.php	37
3.3. Views	39
3.3.1. head.php	39
3.3.2. navigation.php	40
3.3.3. main.php	42
3.3.4. browse.php	46
3.4.5. my_list.php	47
3.4.6. about.php	50
3.4.7. contact.php	50
3.4.8. footer.php	52

3.4. Php logika	57
3.4.1. configuration.php	57
3.4.2. database.php	57
3.4.3. utilities.php	58
3.4.4. login.php	65
3.4.5. register.php	67
3.4.6. logout.php	69
3.4.7. contact.php	69
3.4.8. addNewComic.php	71
3.5. AJAX logika	76
3.5.1. comics.php	76
3.5.2. addToList.php	82
3.5.3. removeFromList.php	84
3.5.4. getComic.php	85
3.5.5. deleteComic.php	87
3.5.6. addVote.php	89
3.5.7. removeVote.php	91
3.5.8. comicStats.php	92
3.6. JavaScript	93
3.6.1. comics.js	93
3.6.2. mainslider.js	102
3.6.3. modal.js	107
3.6.4. utils.js	110
3.6.5. validations.js	111
3.7. CSS	120
3.7.1. utils.css	120
3.7.2. style.css	138

1.Uvod

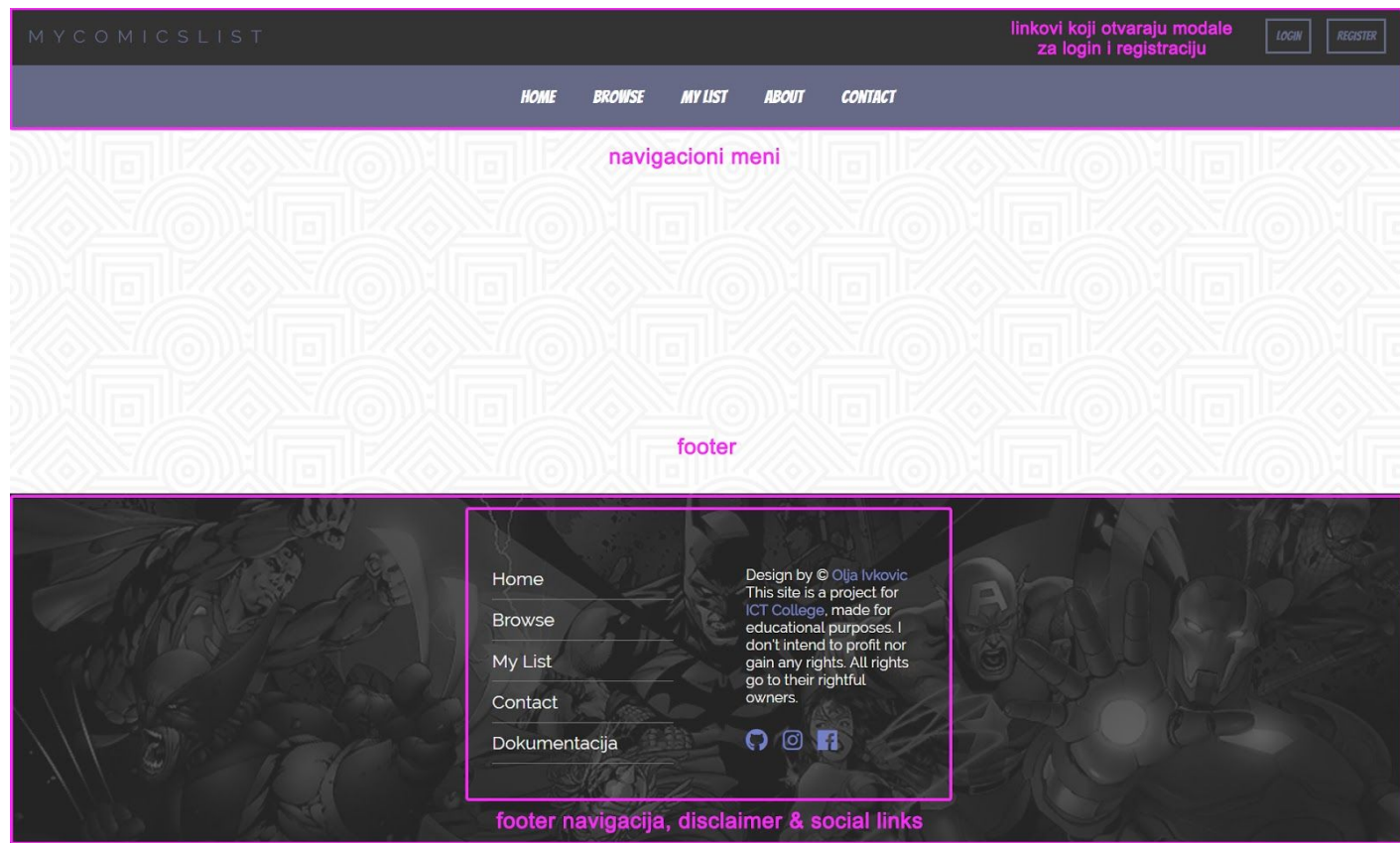
1.1. Korišćenje tehnologije

1. HTML 5
2. CSS 3
3. JavaScript
4. JQuery
5. PHP without frameworks
6. MySQL
7. PhpMyAdmin
8. AJAX
9. Apache
10. Git
11. Amazon Web Services
12. Visual Studio Code

1.2. Opis funkcionalnosti

1. Registracija i provera podataka kako sa klijentske, tako i sa serverske strane
2. Logovanje koje obezbedjuje korisnicima da se prijave na platformu
3. Dinamički generisan meni u header-u i footer-u
4. Dinamički učitani skoro svi podaci iz baze podataka
5. Admin panel preko kog je moguće dodavati, ažurirati i brisati stripove, kao i videti statistiku glasanja. Panelu pristupa samo administrator
6. Paginacija realizovana pomoću AJAX-a
7. Filtriranje stripova po žanru, poreklu, tipu i izdavaču, koje koristi paginaciju takodje
8. Dodavanje stripova u sopstvenu listu, kao i otklanjanje stripova iz iste. Mogućnost liste imaju samo prijavljeni korisnici
9. Anketa isključivo za prijavljene korisnike. Moguće je poništiti glas obzirom da jedan korisnik može samo jednom glasati
10. Kontakt forma koja validira podatke sa klijentske i serverske strane, i šalje administratoru poruku na mail
11. Responzivan slajder stripova na početnoj stranici
12. Full responzivan, pažljivo osmišljen i realizovan dizajn platforme

1.3. Templates



```
<header class='flex-col center'>

<nav class='log-reg flex-row nav center'>

    <h1>

        <a class='logo' href='/index.php'>MyComicsList</a>

    </h1>

    <ul>

        <?php

            $html = "";

            if(isset($_SESSION['user'])) {

                if($_SESSION['user']->id_role == 1) {
```

```
$html .= "
    <li><a href='index.php?page=panel' class='btn-style'>Admin
panel</a></li>

    ";
}

$html .= "

    <li>

        <a href='../php/logout.php' class='btn-style'>Logout</a>

    </li>

    ";
} else {

    $html .= "<li>

        <a href='#' class='btn-style open-modal-login'>Login</a>

    </li> <li>

        <a href='#' class='btn-style open-modal-register'>Register</a>

    </li>";

}

echo $html;

?>

</ul>

</nav>

<?php

$linkovi = selectMultipleRows($conn, "SELECT * FROM links"); ?>

<nav class='navigation nav center'>
```

```
<ul>

    <?php foreach($linkovi as $link): ?>

        <li>

            <a href='<?= $link->path ?>'><?= $link->name ?></a>

        </li>

    <?php endforeach ?>

</ul>

</nav>

</header>

<section class='list-adv flex-row center'> </section>

<footer>

    <div class="footer-picture flex-row center">

        <ul class="links">

            <?php

                $linkovi = selectMultipleRows($conn, "SELECT * FROM links");

                $trenutna = isset($_GET['page']) ? 'index.php?page=' .
$_GET['page'] : "index.php";

                foreach($linkovi as $link) {

                    if($trenutna != $link->path) {

                        echo "<li><a href='$link->path'>$link->name</a></li>";

                    }

                }

                echo "<li><a href='MyComicsList.pdf'>Dokumentacija</a></li>";
```

```
?>

</ul>

<div class="disclaimer">

  <p>Design by ©

  <strong>Olja Ivkovic</strong></p>

  <p>This site is a project for <strong>ICT College</strong>, made
for educational purposes. I don't intend to profit nor gain any rights.
All rights go to their rightful owners.</p>

  <a href="https://github.com/" class="fab fa-github"></a>

  <a href="https://www.instagram.com/?hl=sr" class="fab
fa-instagram"></a>

  <a href="https://www.facebook.com/" class="fab fa-facebook"></a>

</div>

</div>

<!-- LOGIN MODAL -->

<div class='modal'>

  <div class='card'>

    <div class='card__face card__face--front flex-col center'>

      <h1>Login</h1>

      <span class='modal-exit'>&times;</span>

      <form action='php/login.php' name='login-forma' method='POST'>

        <div class='relative'>

          <input type='text' name='email' placeholder='Email' />
```



```

        <span class='fas fa-at fa-span fs-login absolute'></span>

    </div>

    <p class='form-error errEmail'><?= error_for("email",
"greske"); ?></p>

    <div class='relative'>

        <input type='password' name='password' placeholder='Password'
/>

        <span class='fas fa-key fa-span fs-login absolute'></span>

    </div>

    <p class='form-error errPass'><?= error_for("password",
"greske"); ?></p>

    <button id='login' name='login'
class='btn-style'>Login</button>

    <button id='register-btn' class='btn-style'>Register</button>

</form>

</div>

<!-- REGISTER MODAL -->

<div class='card__face card__face--back flex-col center'>

    <h1>Register</h1>

    <span class='modal-exit'>&times;</span>

    <form action='php/register.php' name='reg-forma' method='POST' >

        <div class='relative'>

            <input type='text' name='reg-email' placeholder='Email' />

```

```
<span class='fas fa-at fa-span fs-reg absolute'></span>

</div>

<p class='form-error errEmail'><?= error_for("reg_email",
"greske") ;?></p>

<div class='relative'>

    <input type='password' name='reg-password'
placeholder='Password' />

    <span class='fas fa-key fa-span fs-reg absolute'></span>

</div>

<p class='form-error errPass'><?= error_for("reg_password",
"greske") ;?></p>

<div class='relative'>

    <input type='password' name='reg-confirm'
placeholder='Confirm password' />

    <span class='fas fa-unlock fa-span fs-reg absolute'></span>

</div>

<p class='form-error errConfirm'><?= error_for("reg_confirm",
"greske") ;?></p>

<button id='login-btn' name='login'
class='btn-style'>Login</button>

<button id='register' class='btn-style'
name='register'>Register</button>

</form>

</div>
```

```
</div>  
  
</div>  
  
</footer>
```

1.4. Korišćene biblioteke

1.4.1. Select2 biblioteka

<https://select2.org/>

1.4.2. Chartist biblioteka

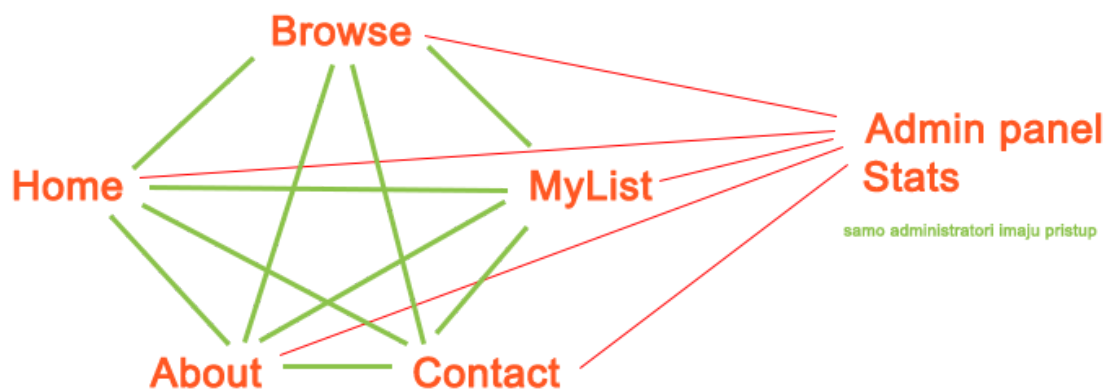
<https://gionkunz.github.io/chartist-js/>

1.4.3. JQuery Knob biblioteka

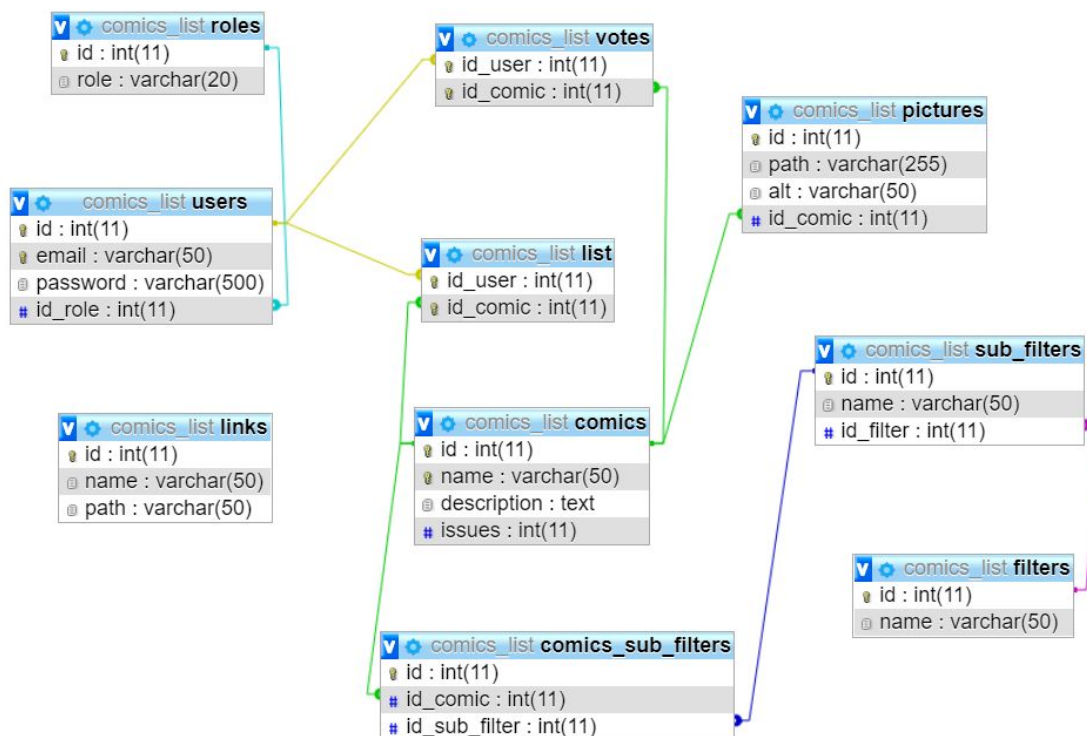
<https://github.com/aterrien/jquery-knob>

2. Organizacija

2.1. Organizaciona šema



2.2. Šema baze podataka



2.3. Sitemap

```

<?xml version="1.0" encoding="UTF-8"?>

<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">

  <url>

    <loc>https://mycomicslist.datapoint.rs/index.php</loc>

    <lastmod>2018-06-19</lastmod>

    <changefreq>weekly</changefreq>

    <priority>1</priority>

  </url>

  <url>
  
```

```
<loc>https://mycomicslist.datapoint.rs/index.php?page=browse</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>daily</changefreq>

<priority>0.8</priority>

</url>

<url>

<loc>https://mycomicslist.datapoint.rs/index.php?page=my_list</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>monthly</changefreq>

<priority>0.7</priority>

</url>

<url>

<loc>https://mycomicslist.datapoint.rs/index.php?page=about</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>yearly</changefreq>

<priority>0.4</priority>

</url>

<url>

<loc>https://mycomicslist.datapoint.rs/index.php?page=contact</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>yearly</changefreq>

<priority>0.6</priority>

</url>

<url>
```

```
<loc>https://mycomicslist.datapoint.rs/index.php?page=panel</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>weekly</changefreq>

<priority>0.9</priority>

</url>

<url>

<loc>https://mycomicslist.datapoint.rs/index.php?page=stats</loc>

<lastmod>2018-06-19</lastmod>

<changefreq>monthly</changefreq>

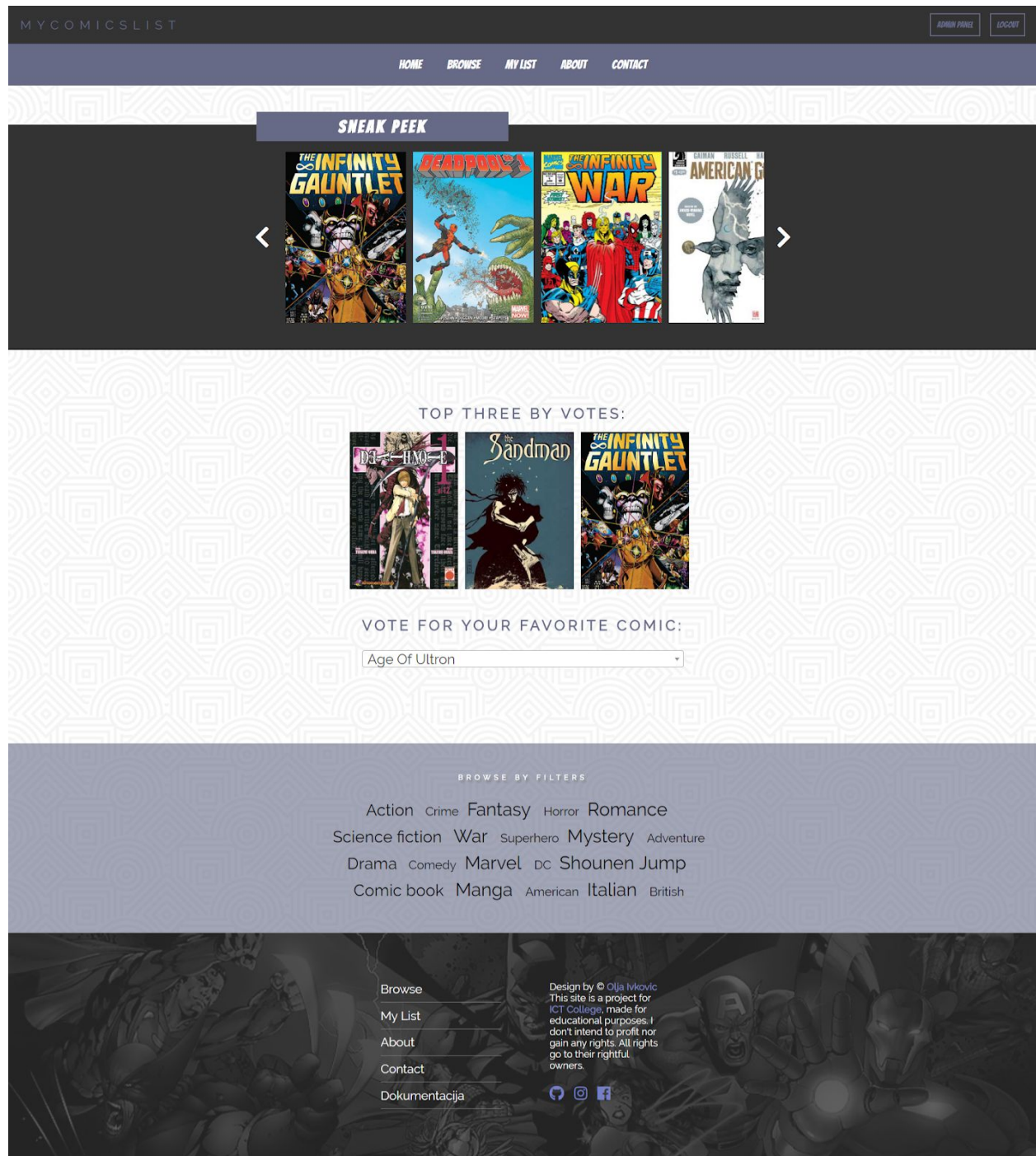
<priority>0.8</priority>

</url>

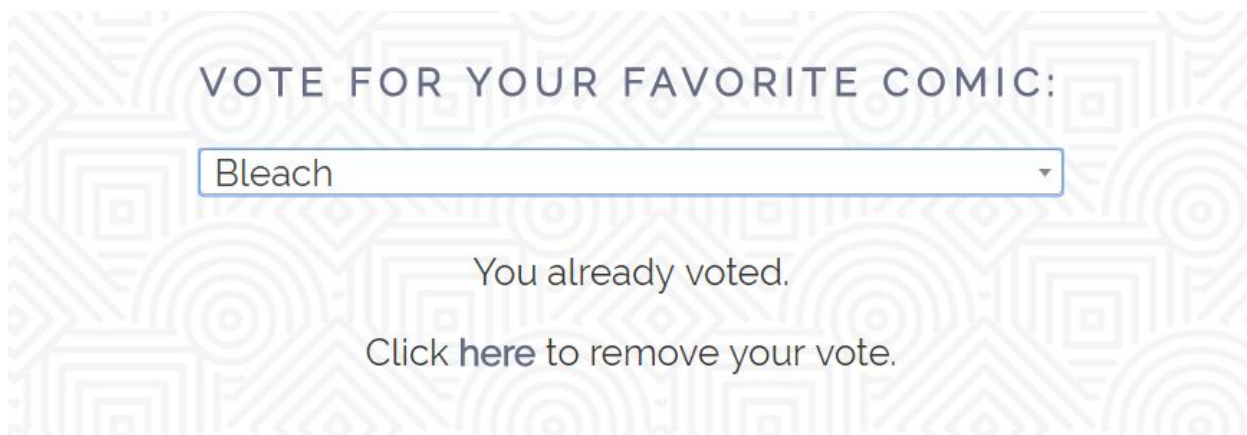
</urlset>
```

2.4. Opis stranica sajta

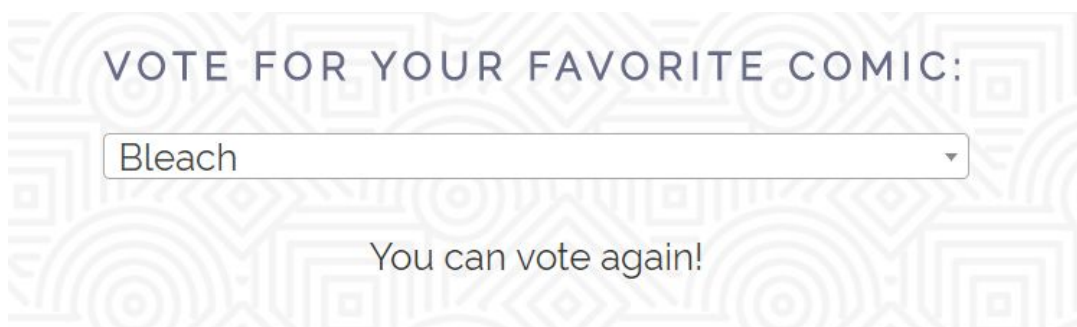
2.4.1. Home page



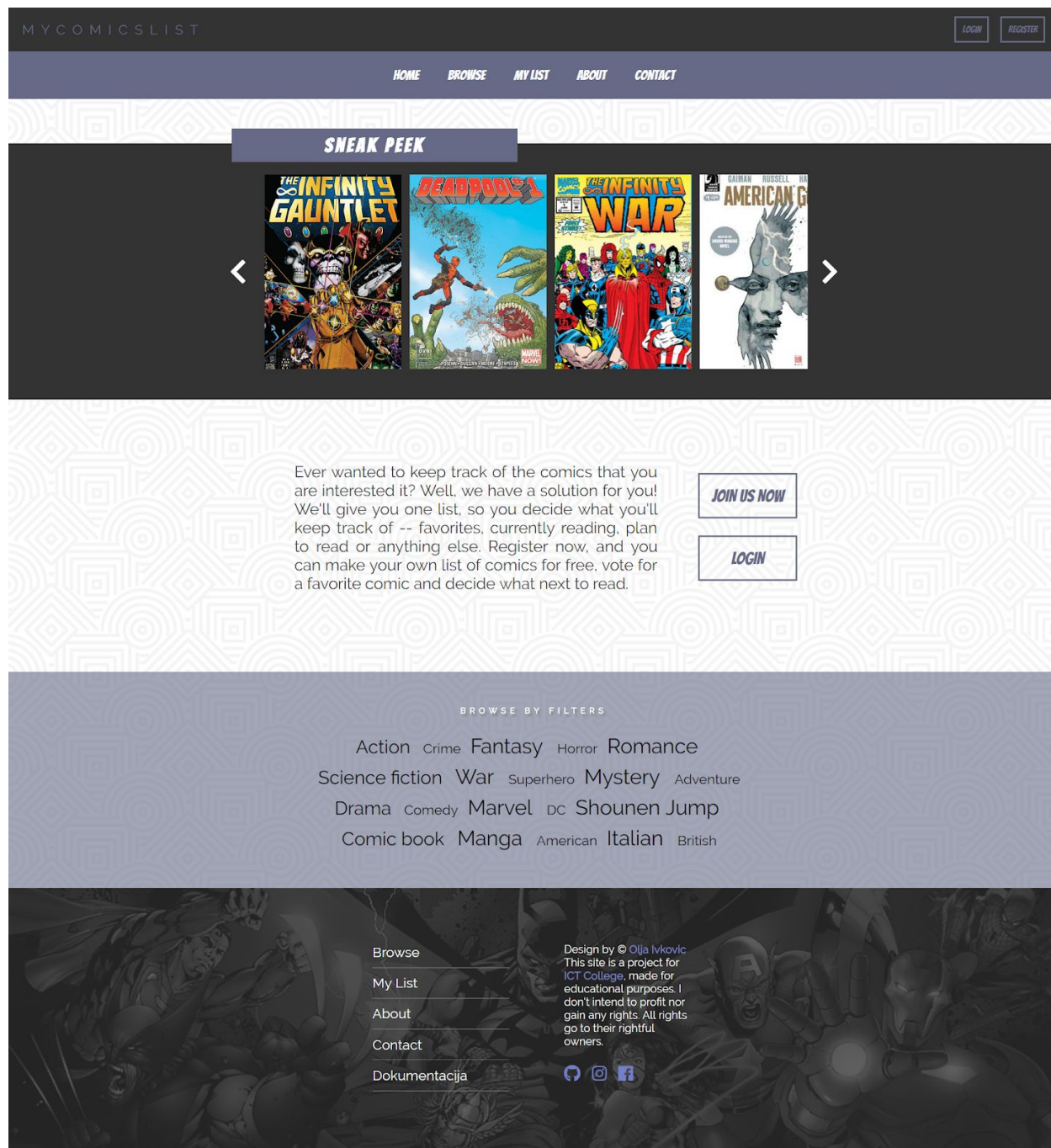
Kada je korisnik ulogovan, ima pristup svim stranicama sajta, sem admin panelu. Početnu stranicu sačinjava responzivan slajder koji prikazuje 10 stripova sortiranih po najmanjem broju poglavlja (issue-a). Ispod njega, ulogovan korisnik može videti top 3 stripa sortirana po najvećem broju glasova. Hover-ovanjem slike, može se videti ime stripa i koliko glasova ima. Ispod toga, korisnik vidi selectbox stilizovan pomoću biblioteke **select2**. Korisnik može samo jednom da glasa, tako da kada pokuša naredni put da glasa, pojavi mu se:



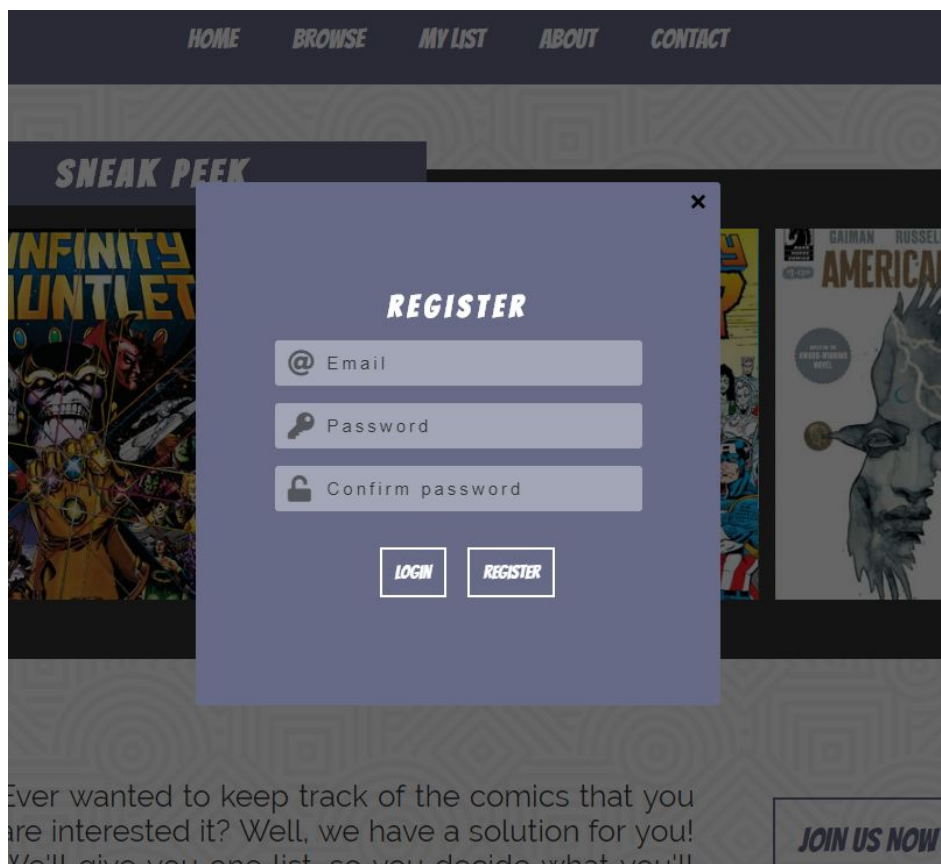
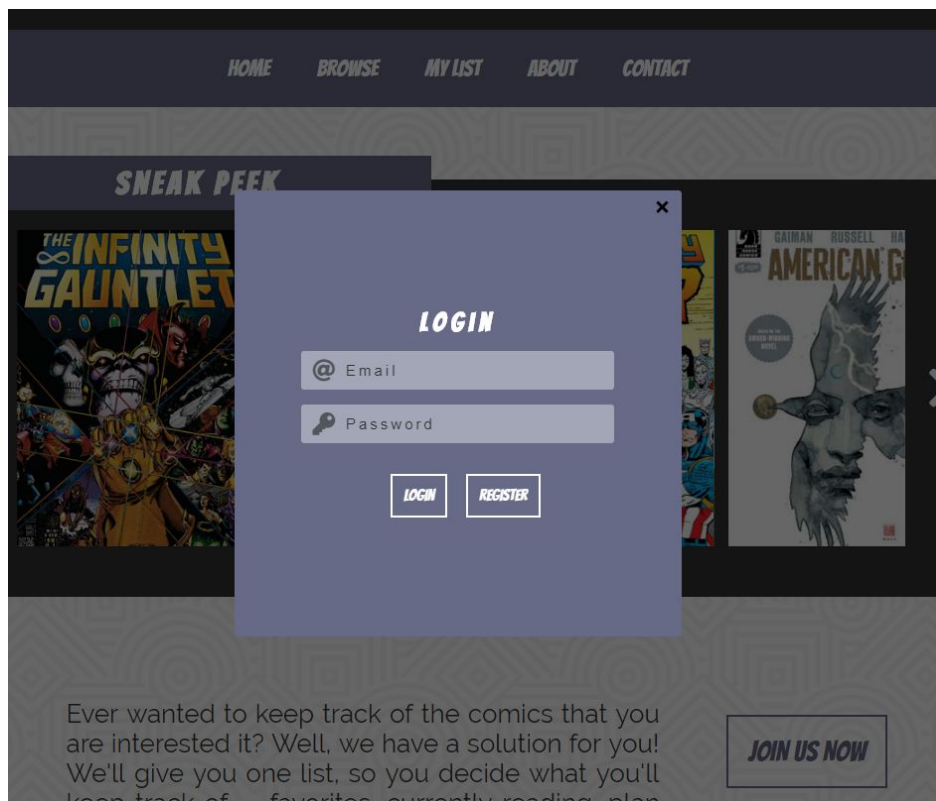
Klikom na "here" korisnik može da poništi svoj glas, pa nakon toga može ponovo da glasa. Statistika glasova se menja bez potrebe za refresh-ovanjem stranice.



Neulogovan korisnik nema pravo glasanja i ovako mu izgleda home stranica:



Klikom na login ili register (ili join us now), otvara se modal koji pokazuje datu formu. Iz login modala klikom na register rotira se modal i korisnik vidi formu za registrovanje. Isti slučaj sa rotacijom je i na ovoj formi prisutan. Kada se korisnik uloguje, vidi link za Logout koji ga briše iz sesije.



Podaci prosledjeni formom validiraju se i sa klijentske i sa serverske strane regularnim izrazima. Greške se ispisuju i prilikom klijentske validacije, i prilikom serverske, u slučaju da je korisnik isključio javascript u browseru. Prilikom registracije, ako su podaci ispravno uneti, korisnik se upisuje u bazu i odmah upisuje u sesiju (login-uje).

U footeru se nalaze linkovi svih stranica sajta, osim stranice na kojoj se korisnik trenutno nalazi.

Izabirom jednog od filtera sa home stranice, korisnik se prebacuje na Browse stranicu, prikazujući mu stripove tog filtera, uz smooth scrolling.

2.4.2. Browse page

Izgled za neulogovanog korisnika:

MYCOMICSLIST

HOME

BROWSE

MY LIST

ABOUT

CONTACT

LOGIN

REGISTER


FILTERS

Genre

Publisher

Type

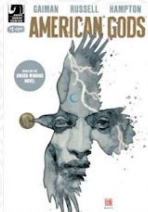
Origin



Attack On Titan

Hundreds of years ago, horrifying creatures which resembled humans appeared. These mindless, towering giants, called "titans," proved to be an existential threat, as they preyed on whatever humans they could find in order to satisfy a seemingly unending appetite. Unable to effectively combat the titans, mankind was forced to barricade themselves within large walls surrounding what many saw as their last hope for a more secure future.

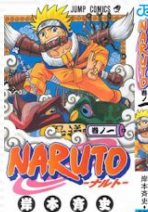
Issues/Chapters: 126



American Gods

This supernatural American road trip fantasy tells the story of a war between the ancient and modern gods. Shadow Moon gets out of jail only to discover his wife is dead. Defeated, broke, and uncertain where to go from here, he meets the mysterious Mr. Wednesday, who employs him to serve as his bodyguard—thrusting Shadow into a deadly world where ghosts of the past come back from the dead, and a new war is imminent.


Issues/Chapters: 9



Naruto

Whenever Naruto Uzumaki proclaims that he will someday become the Hokage—a title bestowed upon the best ninja in the Village Hidden in the Leaves—no one takes him seriously. Since birth, Naruto has been shunned and ridiculed by his fellow villagers. But their contempt isn't because Naruto is loud-mouthed, mischievous, or because of his ineptitude in the ninja arts, but because he's a demon.


Issues/Chapters: 700



Infinity Gauntlet

It's the Avengers, the New Warriors, the X-Men and more against the omnipotent Eternal, Thanos! The Mad Titan has become the most powerful being in the universe, and enslavement or destruction may be the only choices he gives!! The successive Starlin sagas that shook space and time start here!


Issues/Chapters: 6



Death Note

A shinigami, a god of death, can kill any person—provided they see their victim's face and write their victim's name in a notebook called a Death Note. One day, Ryuk, bored by the shinigami lifestyle and interested in seeing how a human would use a Death Note, drops one into the human realm. High school student and prodigy Light Yagami stumbles upon the Death Note and—since he deprecates the state of the world—takes the death note.

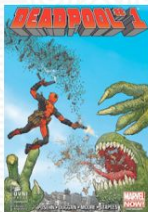
Issues/Chapters: 97



Sandman

The Sandman is the universally lauded masterpiece following Morpheus, Lord of the Dreaming—a vast hallucinatory landscape housing all the dreams of any and everyone who's ever existed. Regardless of cultures or historical eras, all dreamers visit Morpheus' realm. Upon his escape from an embarrassing captivity at the hands of a mere mortal, Morpheus finds himself at a crossroads, forced to deal with the unknown.


Issues/Chapters: 10



Deadpool Vol 1

Dead former United States president, from George Washington to Gerald Ford, have been resurrected—and that's bad. The Marvel heroes can't be the ones to stop them, someone is needed with the reputation, skills and plausible deniability to take out these com-monsters in chief. Deadpool time is NOW! Be here as Deadpool de-un-deadifies ex-presidents left and right, matching wits with T-Rex, Nick Nixon, Britton.

Issues/Chapters: 6



Batman The Dark Knight

The Dark Knight struggles against a deadly – yet strangely familiar – foe in this phenomenal debut issue from superstar writer/artist David Finch (BRIGHTEST DAY, ACTION COMICS)! As a mysterious figure slinks through the halls of Arkham Asylum, Batman must fight his way through a gauntlet of psychos, and Bruce Wayne faces the unexpected legal ramifications of Batman incorporated!

Issues/Chapters: 9

1

2

Home

My List

About

Contact

Dokumentacija

Design by © Oja Ivkovic

This site is a project for educational purposes. I don't intend to profit nor gain any rights. All rights go to their rightful owners.

Browse stranica daje nam mogućnost pregleda svih stripova u bazi, dovučenih 8 po strani. Paginacija se dinamično pravi, u zavisnosti od broja stripova koje dovlačimo, bili oni filtrirani ili ne. Takodje, kada se izaberu filteri, prikazuju se stripovi tog izabranog filtera, takodje 8 po strani. Logovan korisnik vidi dugmiće za dodavanje stripa u svoju listu, ili sklanje sa iste.

MYCOMICSLIST

ADMIN PANEL

LOGOUT

HOME

BROWSE

MY LIST

ABOUT

CONTACT

FILTERS

Genre

Action

Crime

Fantasy

Horror

Romance

Science fiction

War

Superhero

Mystery

Adventure

Drama

Comedy

Publisher

Marvel

DC

Shounen Jump

Type

Comic book

Manga

Origin

American

Italian

British

AMERICAN GODS

ADD

This supernatural American road trip fantasy tells the story of a war between the ancient and modern gods. Shadow Moon gets out of jail only to discover his wife is dead. Defeated, broke, and uncertain where to go from here, he meets the mysterious Mr. Wednesday, who employs him to serve as his bodyguard—thrusting Shadow into a deadly world where ghosts of the past come back from the dead, and a civil war is imminent.

Issues/Chapters: 9

DEATH NOTE

REMOVE

A shinigami, as a god of death, can kill any person—provided they see their victim's face and write their victim's name in a notebook called a Death Note. One day, Ryuk, bored by the shinigami lifestyle and interested in seeing how a human would use a Death Note, drops one into the human realm. High school student and prodigy Light Yagami stumbles upon the Death Note and—since he depletes the state of the world—tests the deadly notebook.

Issues/Chapters: 97

FULLMETAL ALCHEMIST

ADD

Alchemists are knowledgeable and naturally talented individuals who can manipulate and modify matter due to their art. Yet despite the wide range of possibilities, alchemy is not as all-powerful as most would believe. Human transmutation is strictly forbidden, and whoever attempts it risks severe consequences. Even so, siblings Edward and Alphonse Elric decide to ignore this great taboo and bring their mother back to life.

Issues/Chapters: 116

NARUTO

REMOVE

Whenever Naruto Uzumaki proclaims that he will someday become the Hokage—a title bestowed upon the best ninja in the Village Hidden in the Leaves—no one takes him seriously. Since birth, Naruto has been shunned and ridiculed by his fellow villagers. But their contempt isn't because Naruto is loud-mouthed, mischievous, or because of his ineptitude in the ninja arts, but because there is a demon inside.

Issues/Chapters: 700

ONE PIECE

ADD

Gold D. Roger, a man referred to as the "Pirate King" is set to be executed by the World Government. But just before his demise, he confirms the existence of a great treasure, One Piece, located somewhere within the vast ocean known as the Grand Line. Announcing that One Piece can be claimed by anyone worthy enough to reach it, the Pirate King is executed and the Great Age of Pirates begins. Twenty-seven years later...

Issues/Chapters: 893

SANDMAN

ADD

The Sandman is the universally lauded masterpiece following Morpheus, Lord of the Dreaming—a vast hallucinatory landscape housing all the dreams of any and everyone who's ever existed. Regardless of cultures or historical eras, all dreamers visit Morpheus' realm. Upon his escape from an embarrassing captivity at the hands of a mere mortal, Morpheus finds himself at a crossroads, forced to deal with the enormous challenges...

Issues/Chapters: 10

BLEACH

REMOVE

For as long as he can remember, high school student Ichigo Kurosaki has been able to see the spirits of the dead, but that has not stopped him from leading an ordinary life. One day, Ichigo returns home to find an intruder in his room who introduces himself as Rukia Kuchiki, a Soul Reaper tasked with helping souls pass over. Suddenly, the two are jolted from their conversation when a Hollow—an evil spirit known for eating human souls—...

Issues/Chapters: 705

ARKHAM ASYLUM

ADD

The inmates of Arkham Asylum have taken over Gotham's detention center for the criminally insane on April Fool's Day, demanding Batman in exchange for their hostages. Accepting their demented challenge, Batman is forced to endure the personal hells of the Joker, Scarecrow, Poison Ivy, Two-Face and many other sworn enemies in order to save the innocents and retake the prison. To win this run through this, you are...

Issues/Chapters: 63

1

2

Home

My List

About

Contact

Dokumentacija

Design by © Olla Movic
This site is a project for ICT College, made for educational purposes. I don't intend to profit nor gain any rights. All rights go to their rightful owners.

Klikom na link “Add”, korisnik dodaje strip u svoju listu, i automatski mu se pojavljuje link “Remove” za sklanjanje stripa iz liste.

2.4.3. MyList page

Svaki korisnik ima svoju sopstvenu listu kojoj pristupa samo on na stranici My List. Preko ove stranice takodje korisnik može skloniti strip iz liste.

MYCOMICS LIST

ADMIN PANEL LOGOUT

HOME BROWSE MY LIST ABOUT CONTACT

REMOVE

Death Note

A shinigami, as a god of death, can kill any person—provided they see their victim's face and write their victim's name in a notebook called a Death Note. One day, Ryuk, bored by the shinigami lifestyle and interested in seeing how a human would use a Death Note, drops one into the human realm. High school student and prodigy Light Yagami stumbles upon the Death Note and—since he deplores the state of the world—tests the deadly notebook by writing a criminal's name in it. When the criminal dies immediately following his experiment with the Death Note, Light is greatly surprised and quickly recognizes how devastating this power that has fallen into his

Issues/Chapters: 97

REMOVE

Naruto

Whenever Naruto Uzumaki proclaims that he will someday become the Hokage—a title bestowed upon the best ninja in the Village Hidden in the Leaves—no one takes him seriously. Since birth, Naruto has been shunned and ridiculed by his fellow villagers. But their contempt isn't because Naruto is loud-mouthed, mischievous, or because of his ineptitude in the ninja arts, but because there is a demon inside him. Prior to Naruto's birth, the powerful and deadly Nine-Tailed Fox attacked the village. In order to stop the rampage, the Fourth Hokage sacrificed his life to seal the demon inside the body of the newborn Naruto. And so when he is accidently in Team 7...alone with his new

Issues/Chapters: 700

REMOVE

Attack On Titan

Hundreds of years ago, horrifying creatures which resembled humans appeared. These mindless, towering giants, called "titans," proved to be an existential threat, as they preyed on whatever humans they could find in order to satisfy a seemingly unending appetite. Unable to effectively combat the titans, mankind was forced to barricade themselves within large walls surrounding what may very well be humanity's last safe haven in the world. In the present day, life within the walls has finally found peace, since the residents have not dealt with titans for many years. Eren Yeager, Mikasa Ackerman, and Armin Arlert are three young children who dream of experiencing all

Issues/Chapters: 126

REMOVE

Bleach

For as long as he can remember, high school student Ichigo Kurosaki has been able to see the spirits of the dead, but that has not stopped him from leading an ordinary life. One day, Ichigo returns home to find an intruder in his room who introduces herself as Rukia Kuchiki, a Soul Reaper tasked with helping souls pass over. Suddenly, the two are jolted from their conversation when a Hollow—an evil spirit known for consuming souls—attacks. As Ichigo makes a rash attempt to stop the Hollow, Rukia steps in and shields him from a counterattack. Injured and unable to keep fighting, Rukia suggests a risky plan—transfer half of her Soul Reaper powers to Ichigo. He accepts and, in

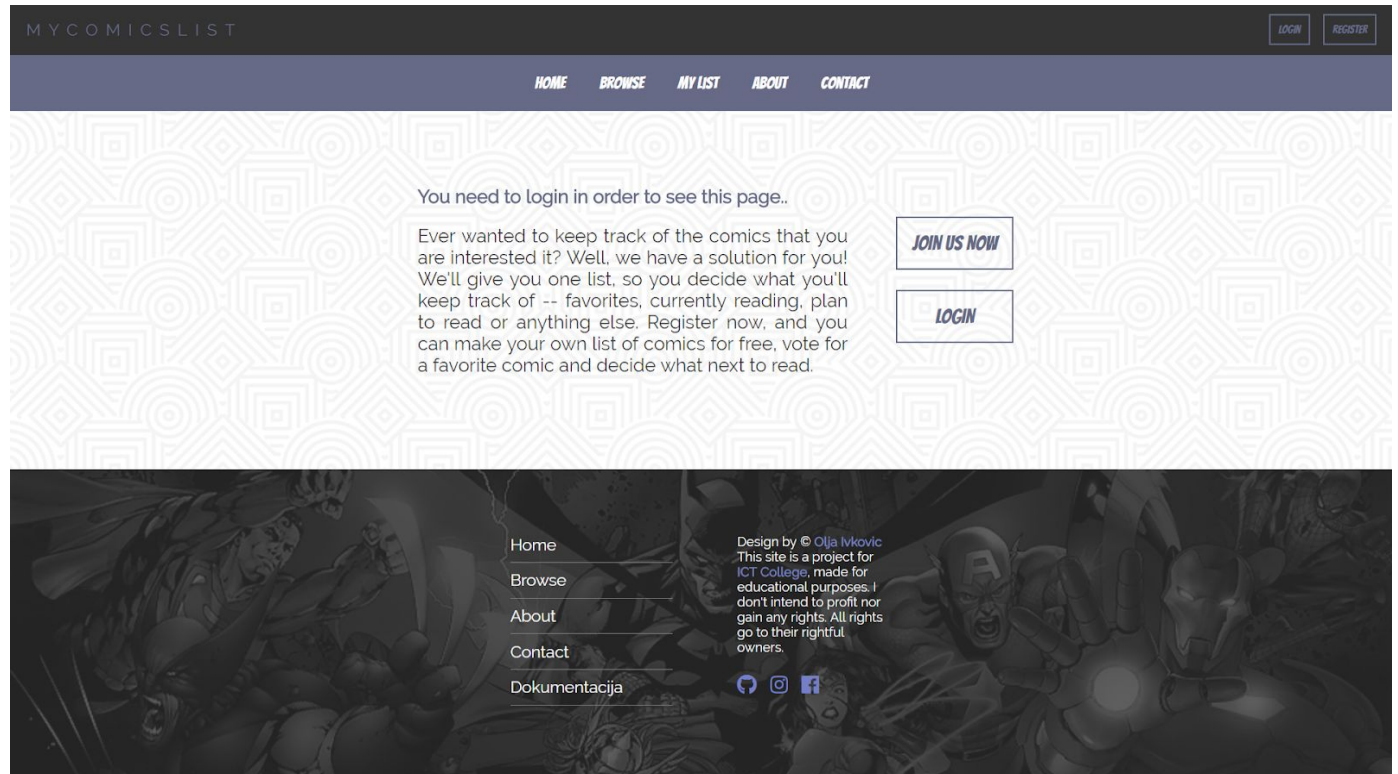
Issues/Chapters: 705

Home Browse About Contact Dokumentacija

Design by © Oja Ibric
This site is a project for
ICT College, made for
educational purposes. I
don't intend to profit nor
gain any rights. All rights
go to their rightful
owners.

Facebook Twitter Instagram

Ne ulogovani korisnici ovako vide MyList stranicu:



2.4.4. Contact page

MYCOMICS LIST

ADMIN PANEL LOGOUT

HOME BROWSE MY LIST ABOUT CONTACT

CONTACT US

Your email address

Your message...

Contact us

Home
Browse
My List
About
Dokumentacija

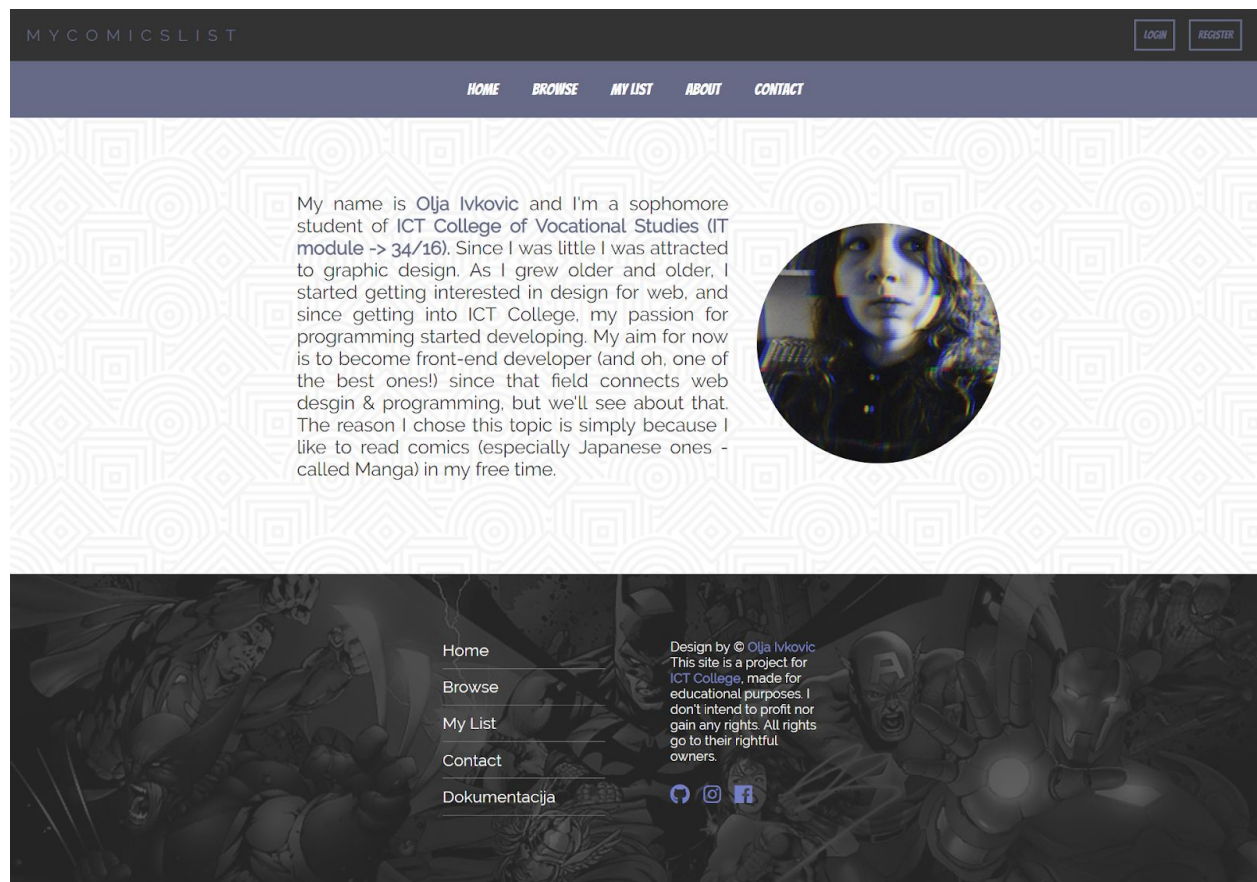
Design by © Olja Ivkovic
This site is a project for
ICT College, made for
educational purposes. I
don't intend to profit nor
gain any rights. All rights
go to their rightful
owners.

Twitter Instagram Facebook

Kontakt stranica služi za kontaktiranje administratora putem mail-a. Podaci se validiraju i na klijentskoj, i na serverskoj strani. Kada se forma ispravno popuni, dobija se response:



2.4.5. About me page



2.4.6. Admin panel -- panel page

MYCOMICSLIST

ADMIN PANEL LOGOUT

HOME BROWSE MY LIST ABOUT CONTACT

Comics Stats

ADD A NEW COMIC

Name of the comic

Description

Number of issues (chapters)

1

Choose subfilters

Upload a picture

Choose File No file chosen

Please upload a picture that is portrait oriented!

Insert

CHOOSE A COMIC

Izaberite...

Delete

Home

Browse

My List

About

Contact

Dokumentacija

Design by © Olja Ivkovic
This site is a project for ICT College, made for educational purposes. I don't intend to profit nor gain any rights. All rights go to their rightful owners.

Twitter Instagram Facebook

Forma za insertovanje stripova koja se validira regularnim izrazima sa serverske strane. Svakom stripu koji se insertuje potrebno je popuniti sva polja. Barem jedan filter mora da se odabere, ime stripa mora se sastojati od jedne ili više reči sa velikim početnim slovom. U opis stripa je potrebno upisati barem 10 karaktera. Potrebno je i upisati koliko

strip ima issue-a odnosno chaptera. Maksimalna vrednost je 1000. Issue number je stilizovan **jquery-knob** bibliotekom.

Slika koja se uploaduje mora biti vertikalna. Moja funkcija skalira slike tako da se aspect ratio ne uništi. Height se uvek skalira na 355px., a width se akomodira. Maksimalna veličina slike je 2MB i dozvoljeni su samo JPG i JPEG formati. Nakon ispravno unetih podataka, podaci se upisuju u bazu, a slika se prebacuje na server.

Da bi update-ovali ili obrisali strip potrebno je odabrati isti iz selectbox-a. Nakon odabira, klikom na dugme Delete moguće je obrisati strip. Ako želimo da update-ujemo strip, forma se popuni podacima iz baze za odabrani strip. Klikom na dugme Update podaci za taj strip se update-uju. Regularni izrazi važe prilikom update-ovanja takodje.

MYCOMICSLIST

ADMIN PANEL

LOGOUT

HOME

BROWSE

MY LIST

ABOUT

CONTACT

Comics

Stats

UPDATING A COMIC

Name of the comic

Age Of Ultron

Description

Humanity's time is over. Submit or perish, because the Age of Ultron has arrived! The artificial intelligence known as Ultron has fought for years to eradicate mankind...and now, it has all but succeeded. The few remaining heroes are battered, broken, almost beaten and left considering desperate

Number of issues (chapters)

15

Choose subfilters

Action

Comic book

Upload a picture

Choose File

No file chosen


Please upload a picture that is portrait oriented!

Update

CHOOSE A COMIC

Age Of Ultron

Delete



Home

Browse

My List

About

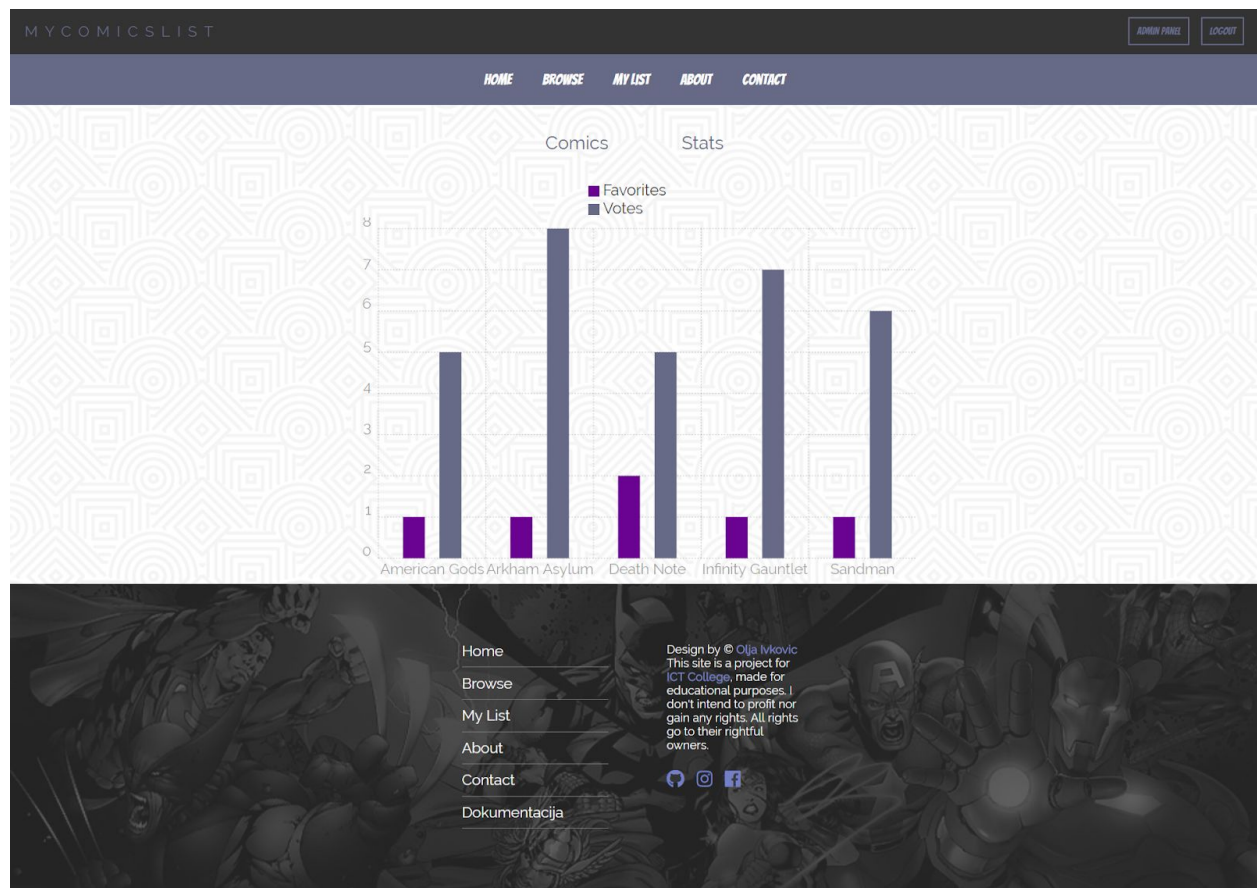
Contact

Dokumentacija

Design by © Olja Ivkovic

This site is a project for ICT College, made for educational purposes. I don't intend to profit nor gain any rights. All rights go to their rightful owners.

2.4.7. Admin panel -- stats page



Na ovoj stranici može se videti statistika glasanja, kao i statistika stripova koje su ljudi dodali u svoju listu, iscrtana pomoću **chartist** biblioteke.

3. Kodovi

3.1. Root

3.1.1. index.php

```
<?php

    session_start();

    require_once "php/utilities.php";

    require_once "../database.php";

?>

<!DOCTYPE html>

<html>

    <?php

        include_once "views/head.php";

    ?>

    <body>

        <?php

            $page = null;

            if(isset($_GET['page'])) {

                $page = $_GET['page'];

            }

            include_once "views/navigation.php";

            switch($page) {

                case 'browse':
```



```
        include_once "views/browse.php";

        break;

    case 'panel':

        if(isset($_SESSION['user']) && $_SESSION['user']->role ==
"administrator") {

            include_once "../admin/panel.php";

        } else {

            include "views/main.php";

        }

        break;

    case 'stats':

        if(isset($_SESSION['user']) && $_SESSION['user']->role ==
"administrator") {

            include_once "../admin/stats.php";

        } else {

            include "views/main.php";

        }

        break;

    case 'my_list':

        include "views/my_list.php";

        break;

    case 'about':

        include "views/about.php";

        break;

    case 'contact':
```

```
        include "views/contact.php";

        break;

    default:

        include "views/main.php";

        break;

    }

    include_once "views/footer.php";

?>

</body>

</html>
```

3.2. Admin panel

3.2.1. panel.php

```
<div class='flex-row center panel-navigation'>

    <a href='index.php?page=panel'>Comics</a>

    <a href='index.php?page=stats'>Stats</a>

</div>

<div class='flex-row center add-comic'>

    <form action='/php/addNewComic.php' method='POST'
    enctype='multipart/form-data'>

        <input type='hidden' name='_method' id="_method" value='POST' />

        <input type='hidden' name='comicID' id="comicID" value='' />

        <h1 id="naslov">Add a new comic</h1>
```

```

<div class='input-group'>

    <label>Name of the comic</label>

    <input type='text' name='comicName' id='comicName' />

</div>

<span class='form-error'><?= error_for("comicName", "comicErrors");
?></span>

<div class='input-group'>

    <label>Description</label>

    <textarea name='desc' id='desc'></textarea>

</div>

<span class='form-error'><?= error_for("description", "comicErrors");
?></span>

<div class='input-group'>

    <label>Number of issues (chapters)</label>

    <input type="text" name='issues' value="1" data-min="1"
data-max="1000" class="dial" data-fgColor="#666A86">

</div>

<span class='form-error'><?= error_for("issues", "comicErrors");
?></span>

<div class='input-group'>

    <label>Choose subfilters</label>

    <select class='multipleSelect' multiple name='tags[]'>

        <?php

```

```

        $upit = "SELECT * FROM sub_filters";

        $filters = selectMultipleRows($conn, $upit);

        foreach($filters as $filter): ?>

            <option value="<?= $filter->id ?>"><?= $filter->name ?></option>

        <?php endforeach ?>

    </select>

</div>

<span class='form-error'><?= error_for("tags", "comicErrors");
?></span>

<div class='input-group'>

    <label>Upload a picture</label>

    <input type='file' name='comic-por-pic'>

    <p>Please upload a picture that is portrait oriented!</p>

    <?php

        $subIndexes = ['type', 'size'];

        foreach($subIndexes as $subIndex):

            ?>

                <p class='form-error'><?= multi_error_for("comic-por-pic",
"comicErrors", $subIndex); ?></p>

            <?php endforeach ?>

        <?php unset($_SESSION['comicErrors']); ?>

```

```

    <button id='insertComic' class='change-btn'
name='insert'>Insert</button>

    <?php

        if(isset($_SESSION['upload'])) {

            echo "<strong>" . $_SESSION['upload'] . "</strong>";

            unset($_SESSION['upload']);

        }

    ?>

</form>

</div>

<div class="flex-col add-comic update-comic">

    <h1>Choose a comic</h1>

    <?php $stripovi = selectMultipleRows($conn, "select id, name from
comics") ?>

    <select id='izbor-stripa'>

        <option value='0'>Izaberite...</option>

        <?php foreach($stripovi as $strip): ?>

            <option value='<?= $strip->id ?>'><?= $strip->name ?></option>

        <?php endforeach ?>

    </select>

    <button id="delete" disabled class='disabled change-btn'>

        Delete

    </button>

    <img id="preview" />

</div>

```

```
<script>

window.addEventListener('DOMContentLoaded', () => {

    $(".dial").knob({

        min: 1,

        max: 1000,

        width: 100,

        height: 100

    });

    $(".multipleSelect").select2();

    $("#izbor-stripa").select2();

    $("#delete").click(function() {

        var id = $("#izbor-stripa").val();

        ajaxPost("ajax/deleteComic.php", {id: id}, (resp) => {

            console.log(resp);

        });

    });

    $("#izbor-stripa").change(function() {

        $("#delete").removeAttr('disabled');

        $("#delete").removeClass('disabled');

        ajaxPost("ajax/getComic.php", {id: $(this).val()}, (data) => {

            let comic = data.comic;
```

```

        let filterIds = data.filters.map((filter) => filter.id_sub_filter +
        "");

        var options = $(".multipleSelect option")

            .filter(function() {

                return filterIds.includes($(this).val());

            })

            .map(function() { return $(this).val(); })

            .toArray();

        $("#comicName").val(comic.name);

        $("#desc").val(comic.description);

        $(".dial").val(comic.issues);


        $(".multipleSelect").val(null).trigger('change');

        $(".multipleSelect").val(options).trigger('change');


        $("#preview").attr("src", comic.path);

        $("#naslov").text("Updating a comic");

        $("#insertComic").text("Update");

        $("#_method").val("UPDATE");

        $("#comicID").val(comic.id);

    })

});

});

</script>

```

3.2.2. stats.php

```
<div class='flex-row center panel-navigation'>

  <a href='index.php?page=panel'>Comics</a>

  <a href='index.php?page=stats'>Stats</a>

</div>

<div class='flex-row center legend'>

  <ul>

    <li class='series-a'>Favorites</li>

    <li class='series-b'>Votes</li>

  </ul>

</div>

<div class='chart-container'>

</div>

<script>

  window.addEventListener('DOMContentLoaded', () => {

    ajaxGet("/ajax/comicStats.php", (data) => {

      let comics = data.reduce((acc, stat) => {

        acc[stat.name] = {favorites: stat.favorite_count, votes:
stat.vote_num};

        return acc;

      }, {});

      var data = {
```

```
labels: Object.keys(comics),

series: [

    Object.values(comics).map((metric) => metric.votes),

    Object.values(comics).map((metric) => metric.favorites)

]

};

var options = {

    axisY: {

        onlyInteger: true,

    },

    seriesBarDistance: 50

};

new Chartist.Bar('.chart-container', data, options).on('draw',
function(data) {

    if(data.type === 'bar') {

        data.element.attr({

            style: 'stroke-width: 30px'

        });

    }

});

});

});

</script>
```

3.3. Views

3.3.1. head.php

```
<head>

<title>MyComicsList</title>

<meta charset="utf-8" />

<meta name="keywords" content="browse, comic, login, register, list,
read, manga, book, fantasy, horror, action, crime, science, favorites,
filters, vote"/>

<meta name="description" content="Ever wanted to keep track of the comics
that you are interested it? Well, we have a solution for you! We'll give
you one list, so you decide what you'll keep track of!"/>

<meta name="author" content="mailto:olja.ivkovic.34.16@ict.edu.rs"/>

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1">

<link
href="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.6-rc.0/css/select
2.min.css" rel="stylesheet" />

<link rel="stylesheet" type="text/css" media="screen"
href="css/chartist.min.css">

<link rel="stylesheet" type="text/css" media="screen"
href="css/utils.css">

<link rel="stylesheet" type="text/css" media="screen"
href="css/style.css">

<link rel="icon" href="images/favicon.ico">

<link rel="stylesheet"
href="https://use.fontawesome.com/releases/v5.0.13/css/all.css"
```



```
integrity="sha384-DNOHZ68U8hZfKXOortjWvjxusGo9WQnrNx2sqG0tfsghAvtVlRW3tvkXW
Zh58N9jp"
```

```
crossorigin="anonymous">
```

```
</head>
```

3.3.2. navigation.php

```
<header class='flex-col center'>
```

```
<nav class='log-reg flex-row nav center'>
```

```
<h1>
```

```
<a class='logo' href='/index.php'>MyComicsList</a>
```

```
</h1>
```

```
<ul>
```

```
<?php
```

```
$html = "";
```

```
if(isset($_SESSION['user'])) {
```

```
if($_SESSION['user']->id_role == 1) {
```

```
$html .= "
```

```
<li><a href='index.php?page=panel' class='btn-style'>Admin
panel</a></li>
```

```
";
```

```
}
```

```
$html .= "
```

```
<li>
```

```
<a href='../php/logout.php' class='btn-style'>Logout</a>
```

```
</li>
```

```
        ";

    } else {

        $html .= "<li>

        <a href='#' class='btn-style open-modal-login'>Login</a>

        </li> <li>

        <a href='#' class='btn-style open-modal-register'>Register</a>

        </li>";

    }

    echo $html;

?>

</ul>

</nav>

<?php

$linkovi = selectMultipleRows($conn, "SELECT * FROM links"); ?>

<nav class='navigation nav center'>

    <ul>

        <?php foreach($linkovi as $link): ?>

            <li>

                <a href='<?=$link->path ?>'><?=$link->name ?></a>

            </li>

        <?php endforeach ?>

    </ul>

</nav>

</header>
```

3.3.3. main.php

```
<section class='slider flex-row center'>

<span id="slide-left" class='fas fa-chevron-left'></span>

<article class='list-of-comics'>

    <div class='sneak-peek'>

        <h2>Sneak Peek</h2>

    </div>

    <div class='image-slider-wrap'>

        <div class='image-slider'>

            <?php

                $upit = "SELECT p.path, p.alt FROM comics c INNER JOIN pictures p
ON c.id = p.id_comic ORDER BY issues LIMIT 0,10";

                $pics = selectMultipleRows($conn, $upit);

                foreach($pics as $pic):

                    ?>

                    <img src='<?= $pic->path ?>' alt='<?= $pic->alt ?>' />

                    <?php endforeach ?>

                </div>

            </div>

        </div>

    </article>

    <span id="slide-right" class='fas fa-chevron-right'></span>

</section>

<?php
```

```

if(isset($_SESSION['user'])): ?>

<?php

    $upit = "SELECT c.name,p.path, p.alt, COUNT(*) AS votes FROM votes v
JOIN comics c ON v.id_comic = c.id JOIN pictures p ON c.id = p.id_comic
GROUP BY c.name, p.path, p.alt ORDER BY votes DESC LIMIT 0,3";

    $comics = selectMultipleRows($conn, $upit);

?>

<section class='list-adv flex-col center'>

    <h2>Top three by votes:</h2>

    <article class='last-added flex-row'>

        <?php

            if(count($comics) == 0) {

                echo "<span>Be the first one to vote!</span>";

            }

            foreach($comics as $comic): ?>

                <figure class='l3-item'>

                    <img src='<?= $comic->path ?>' alt='<?= $comic->alt ?>' />

                    <figcaption class='l3-desc'>

                        <p><?= $comic->name ?></p>

                        <p>Votes: <?= $comic->votes ?></p>

                    </figcaption>

                </figure>

            <?php endforeach ?>

        </article>

```

```

<article class='vote flex-col center'>

  <h2>Vote for your favorite comic:</h2>

  <select id='vote' name='vote'>

    <?php

      $upit = "SELECT id,name FROM comics";

      $comics = selectMultipleRows($conn, $upit);

      foreach($comics as $comic):

        ?>

          <option value='<?= $comic->id ?>'><?= $comic->name ?></option>

        <?php endforeach ?>

      </select>

      <span class='vote-msg'></span>

      <span class='click-here'> Click <a href=''>here</a> to remove your
vote.</span>

    </article>

  </section>

<?php else: ?>

<section class='list-adv flex-row center'>

  <article class='la-group'>

    <p>Ever wanted to keep track of the comics that you are interested
it? Well, we have a solution for you! We'll give you one list, so you
decide what you'll keep track of -- favorites, currently reading, plan to
read or anything else. Register now, and you can make your own list of

```

```

comics for free, vote for a favorite comic and decide what next to
read.</p>

</article>

<article class='links-lr'>

    <a href='#' class='open-modal-register btn-style bs-white'>JOIN US
NOW</a>

    <a href='#' class='open-modal-login btn-style bs-white'>LOGIN</a>

</article>

</section>

<?php endif; ?>

<section class='section-filter flex-row center'>

<article class='sf-article'>

    <h3>BROWSE BY FILTERS</h3>

    <?php

        $rows = selectMultipleRows($conn, "SELECT DISTINCT sf.name FROM
filters f INNER JOIN sub_filters sf ON f.id = sf.id_filter;");

        foreach($rows as $row) {

            echo "<a href='index.php?page=browse&filter={$row->name}'
class='filter'>{$row->name}</a>";

        }

    ?>

</article>

</section>

```

3.3.4. browse.php

```

<div class='flex-col center'>

  <div class='browse-container row-no-wrap'>

    <div class='filters'>

      <h1 class='text-center heading'>Filters</h1>

      <?php $filters = selectFiltersWithSubfilter($conn) ?>

      <ul>

        <?php foreach($filters as $filter): ?>

          <li>

            <div class='title'>

              <span><?=$filter->name ?></span> <i class="fas
fa-caret-up"></i>

            </div>

            <ul class='sub-items'>

              <?php foreach($filter->subfilters as $subfilter): ?>

                <li>

                  <label class='custom-checkbox'>

                    <span class='search-text'><?=$subfilter->name ?></span>

                    <input class='filter' type='checkbox' value='<?=$subfilter->id ?>' />

                    <span class='checkmark'></span>

                  </label>

                </li>

              <?php endforeach ?>

            </ul>

```

```

        </li>

        <?php endforeach ?>

    </ul>

</div>

<div class='comics flex-row center'>

</div>

</div>

<div class='comics-control'>

</div>

</div>

```

3.4.5. my_list.php

```

<?php

$id = isset($_SESSION['user']) ? $_SESSION['user']->id : null;

if(empty($id)): ?>

    <section class='list-adv flex-row center'>

        <article class='la-group'>

            <p><strong class='warning'>You need to login in order to see this
page..</strong></p>

            <p>Ever wanted to keep track of the comics that you are interested
it? Well, we have a solution for you! We'll give you one list, so you
decide what you'll keep track of -- favorites, currently reading, plan to
read or anything else. Register now, and you can make your own list of
comics for free, vote for a favorite comic and decide what next to
read.</p>

```



```

</article>

<article class='links-lr'>

<a href='#' class='open-modal-register btn-style bs-white'>JOIN US
NOW</a>

<a href='#' class='open-modal-login btn-style bs-white'>LOGIN</a>

</article>

</section>

<?php else:

    $upit = "SELECT c.*, p.path, p.alt FROM comics c

                INNER JOIN list l ON l.id_comic=c.id INNER JOIN pictures p
ON p.id_comic = c.id

                WHERE l.id_user=$id";

    $myComics = $conn->query($upit)->fetchAll();

    if(!empty($myComics)) {

?>

<div class='wrapper-for-comics flex-row center'>

<?php foreach($myComics as $myComic): ?>

<div class='comic'>

    <img src='<?= $myComic->path ?>' alt='<?= $myComic->alt ?>' />

    <div class='ar-btn'>

        <a href='#' data-id='<?= $myComic->id ?>' class='btn-style bs-white
add-to-list remove'>Remove</a>

    </div>

    <h2><?= $myComic->name ?></h2>

```

```
<p class='scroll'><?= $myComic->description ?></p>

<div class='info'>

    <p><strong>Issues/Chapters: </strong><?= $myComic->issues ?></p>

</div>

</div>

<?php endforeach ?>

</div>

<?php } else { ?>

<section class='list-adv flex-row center'>

    <article class='la-group'>

        <p><strong class='warning'>You don't have anything in your list...
browse and add some!</strong></p>

    </article>

</section>

<?php } ?>

<?php endif; ?>
```

3.4.6. about.php

```
<section class='list-adv flex-row center'>

  <article class='la-group'>

    <p>My name is <strong class='warning'>Olja Ivkovic</strong> and I'm a
sophomore student of <strong class='warning'>ICT College of Vocational
Studies (IT module -> 34/16) .</strong> Since I was little I was attracted
to graphic design. As I grew older and older, I started getting interested
in design for web, and since getting into ICT College, my passion for
programming started developing. My aim for now is to become front-end
developer (and oh, one of the best ones!) since that field connects web
desgin & programming, but we'll see about that.

    The reason I chose this topic is simply because I like to read comics
(especially Japanese ones - called Manga) in my free time.</p>

  </article>

  <article class='img-circle'>

    <img src='images/me.jpg' alt='me' />

  </article>

</section>
```

3.4.7. contact.php

```
<div class='flex-row center add-comic'>

  <form action='php/contact.php' name='contact' method='POST'>

    <h1>Contact us</h1>

    <div class='input-group'>

      <label>Your email address</label>

      <input type='text' name='contact-email' />

    </div>
```

```
<span class='form-error errContactEmail'><?= error_for("email",  
"contactErrors"); ?></span>  
  
<div class='input-group'>  
    <label>Your message...</label>  
    <textarea name='contact-message' id='contact-message'></textarea>  
</div>  
  
<span class='form-error errContactMsg'><?= error_for("message",  
"contactErrors"); ?></span>  
  
<button class='change-btn' name='contact'>Contact us</button>  
  
<span class='contactSuccess'><?= error_for("success",  
"contactSuccess"); ?></span>  
  
<?php unset($_SESSION['contactErrors']); ?>  
  
<?php unset($_SESSION['contactSuccess']); ?>  
  
</form>  
</div>
```

3.4.8. footer.php

```

<footer>

<div class="footer-picture flex-row center">

    <ul class="links">

        <?php

            $linkovi = selectMultipleRows($conn, "SELECT * FROM links");

            $trenutna = isset($_GET['page']) ? 'index.php?page=' .
$_GET['page'] : "index.php";

            foreach($linkovi as $link) {

                if($trenutna != $link->path) {

                    echo "<li><a href='$link->path'>$link->name</a></li>";

                }

            }

            echo "<li><a href='MyComicsList.pdf'>Dokumentacija</a></li>";

        ?>

    </ul>

    <div class="disclaimer">

        <p>Design by ©

        <strong>Olja Ivkovic</strong></p>

        <p>This site is a project for <strong>ICT College</strong>, made
for educational purposes. I don't intend to profit nor gain any rights.
All

        rights go to their rightful owners.</p>

```

```

    <a href="https://github.com/" class="fab fa-github"></a>

    <a href="https://www.instagram.com/?hl=sr" class="fab
fa-instagram"></a>

    <a href="https://www.facebook.com/" class="fab fa-facebook"></a>

</div>

</div>

<!-- LOGIN MODAL -->

<div class='modal'>

    <div class='card'>

        <div class='card__face card__face--front flex-col center'>

            <h1>Login</h1>

            <span class='modal-exit'>&times;</span>

            <form action='php/login.php' name='login-forma' method='POST'>

                <div class='relative'>

                    <input type='text' name='email' placeholder='Email' />

                    <span class='fas fa-at fa-span fs-login absolute'></span>

                </div>

                <p class='form-error errEmail'><?= error_for("email",
"greske"); ?></p>

                <div class='relative'>

                    <input type='password' name='password' placeholder='Password'
/>

```

```

        <span class='fas fa-key fa-span fs-login absolute'></span>

    </div>

    <p class='form-error errPass'><?= error_for("password",
"greske"); ?></p>

    <button id='login' name='login'
class='btn-style'>Login</button>

    <button id='register-btn' class='btn-style'>Register</button>

</form>

</div>

<!-- REGISTER MODAL -->

<div class='card__face card__face--back flex-col center'>

    <h1>Register</h1>

    <span class='modal-exit'>&times;</span>

    <form action='php/register.php' name='reg-forma' method='POST' >

        <div class='relative'>

            <input type='text' name='reg-email' placeholder='Email' />

            <span class='fas fa-at fa-span fs-reg absolute'></span>

        </div>

        <p class='form-error errEmail'><?= error_for("reg_email",
"greske"); ?></p>

        <div class='relative'>

            <input type='password' name='reg-password'
placeholder='Password' />

```

```

        <span class='fas fa-key fa-span fs-reg absolute'></span>

    </div>

    <p class='form-error errPass'><?= error_for("reg_password",
"greske") ;?></p>

    <div class='relative'>

        <input type='password' name='reg-confirm'
placeholder='Confirm password' />

        <span class='fas fa-unlock fa-span fs-reg absolute'></span>

    </div>

    <p class='form-error errConfirm'><?= error_for("reg_confirm",
"greske") ;?></p>

    <button id='login-btn' name='login'
class='btn-style'>Login</button>

    <button id='register' class='btn-style'
name='register'>Register</button>

</form>

</div>

</div>

</div>

</footer>

<script
src='https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js'></s
cript>

```



```
<script
src="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.6-rc.0/js/select2.
min.js"></script>

<script>

    <?php if(isset($_SESSION['greske'])): ?>

        window.modalOpen = true;

        $(".fs-login").css("color", "crimson");

    <?php if(isset($_SESSION['greske']['turn_modal'])): ?>

        $(".fs-reg").css("color", "crimson");

        $(".fs-login").css("color", "#333");

        window.turnModal = true;

    <?php endif ?>

    <?php unset($_SESSION['greske']); ?>

    <?php else: ?>

        window.modalOpen = false;

    <?php endif ?>

    $('#vote').select2();

</script>

<script src='scripts/utils.js'></script>

<script src='scripts/jquery.knob.min.js'></script>

<script src='scripts/validations.js'></script>

<script src='scripts/modal.js'></script>

<script src='scripts/mainslider.js'></script>

<script src='scripts/comics.js'></script>

<script src='scripts/chartist.min.js'></script>
```

3.4. Php logika

3.4.1. configuration.php

```
<?php

define("DB_HOST", "localhost");

define("DB_DATABASE", "comics_list");

define("DB_USER", "*****");

define("DB_PASSWORD", "*****");
```

3.4.2. database.php

```
<?php

require_once "configuration.php";

try {

    $dsn = "mysql:host=" . DB_HOST . ";dbname=" . DB_DATABASE .
";charset=utf8";

    $conn = new PDO($dsn, DB_USER, DB_PASSWORD);

    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $conn->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE, PDO::FETCH_OBJ);

} catch(PDOException $e) {

}
```

3.4.3. utilities.php

```
<?php

define('PROJECT_ROOT', dirname(dirname(dirname(__FILE__))));

define('DS', DIRECTORY_SEPARATOR);

function relativeToProjectOS($path) {

    $final = PROJECT_ROOT . $path;

    if(DS != '/') {

        return str_replace('/', "\\ ", $final);

    }

    return $final;

}

function error_for($key, $arr) {

    if(isset($_SESSION["$arr"][$key])) {

        return $_SESSION["$arr"][$key];

    }

    return "";

}

function multi_error_for($key, $arr, $subKey) {

    if(isset($_SESSION["$arr"][$key][$subKey])) {

        return $_SESSION["$arr"][$key][$subKey];

    }

}
```

```
        return "";
    }

    function selectMultipleRows($conn, $upit) {
        $result = $conn->query($upit);
        return $result->fetchAll();
    }

    function bind($conn, $upit, $bindings) {
        $result = $conn->prepare($upit);

        foreach($bindings as $label => &$value) {
            ${"$label"} = $value;
            $result->bindParam(":$label", $$label);
        }

        try {
            $result->execute();
        } catch (PDOException $e) {
            return null;
        }

        return $result;
    }

    function bindWithException($conn, $upit, $bindings) {
```

```
$result = $conn->prepare($upit);

foreach($bindings as $label => &$amp;value) {

    ${"$label"} = $value;

    $result->bindParam(":$label", $$label);

}

$result->execute();

return $result;
}

function bindAndSelect($conn, $upit, $bindings, $fetchOne) {

    $result = bind($conn, $upit, $bindings);

    $selected = null;

    if($fetchOne) {

        if($result->rowCount() > 1) {

            throw new Exception("Rezultat je vratio vise od jednog!!");

        }

        $selected = $result->fetch();

    } else {

        $selected = $result->fetchAll();

    }

    return $selected;
}
```

```

function selectFiltersWithSubfilter($conn) {

    $final = [];

    $records = $conn->query("

        SELECT f.id AS filter_id, f.name AS filter_name,

        sf.id AS sf_id, sf.name AS sf_name FROM filters f

        JOIN sub_filters sf ON f.id=sf.id_filter

    ")->fetchAll();

    foreach($records as $record) {

        $idFilter = $record->filter_id;

        if(!array_key_exists($idFilter, $final)) {

            $single = [

                "id"          => $idFilter,

                "name"         => $record->filter_name,

                "subfilters" => []

            ];

            $single['subfilters'][] = (object)[

                "id"      => $record->sf_id,

                "name" => $record->sf_name

            ];

            $final[$idFilter] = (object)$single;

        } else {

            $final[$idFilter]->subfilters[] = (object)[

                "id"      => $record->sf_id,

                "name" => $record->sf_name

            ];

        }

    }

}

```

```

        ];

    }

}

return $final;

}

function uploadPicture($picture, $lastInsertId, $conn, Callable
$callback, $target) {

    $tmpPath = $picture['tmp_name'];

    $picName = strstr($picture['name'], ".", true);

    $targetPath = relativeToProjectOS("/public/images/comics/") . time() .
    "_" . $picture['name'];

    $targetDB = "images/comics/" . time() . "_" . $picture['name'];

    try {

        resize_image($tmpPath, $target, $callback);

        move_uploaded_file($tmpPath, $targetPath);

        $upit = "INSERT INTO pictures(path, alt, id_comic) VALUES (:putanja,
:alt, :idComic)";

        $uploaded = bind($conn, $upit, [

            "putanja" => $targetDB,

            "alt" => $picName,

            "idComic" => $lastInsertId

        ]);

        return $uploaded;

    } catch(Exception $e) {

```

```
}

function validatePicture($picture, &$errors, $value) {

    $type = $picture['type'];

    $size = $picture['size'];

    $formats = ['image/jpg', 'image/jpeg'];

    if(!in_array($type, $formats)) {

        $errors[$value]['type'] = "Picture must be jpg / jpeg";

    }

    if($size > 2000000) {

        $errors[$value]['size'] = "You need to upload a picture lighter than 2MB";

    }

}

function resize_image_by_width($ratio, $targetWidth) {

    if($ratio > 1) {

        $targetHeight = $targetWidth / $ratio;

    } else {

        throw new Exception("Wrong image form supplied");

        return;

    }

}
```

```
    return [$targetWidth, $targetHeight];
}

function resize_image_by_height($ratio, $targetHeight) {
    if($ratio < 1) {
        $targetWidth = $targetHeight * $ratio;
    } else {
        throw new Exception("Wrong image format supplied");
        return;
    }

    return [$targetWidth, $targetHeight];
}

function resize_image($file, $target, Callable $callback) {
    list($originalWidth, $originalHeight) = getimagesize($file);

    $ratio = $originalWidth / $originalHeight;

    list($targetWidth, $targetHeight) = $callback($ratio, $target);

    $originalImage = imagecreatefromjpeg($file);
    $targetImage = imagecreatetruecolor($targetWidth, $targetHeight);
    imagecopyresampled($targetImage, $originalImage,
```

```

        0, 0, 0, 0,

        $targetWidth, $targetHeight,

        $originalWidth, $originalHeight);

    imagejpeg($targetImage, $file, 100);

}

```

3.4.4. login.php

```

<?php

session_start();

require_once "../../database.php";

require_once "utilities.php";

if(isset($_POST['login'])) {

    $email      = $_POST['email'];

    $password = $_POST['password'];

    $errors     = [];

    $passwordMessage = "A password must have at least one digit, at least
one uppercase char, lowercase chars and it should be at least 8 chars
long";

    $emailMessage     = "You must enter a valid format for email address";

    $rePassword = '/^(?=.*\d)(?=.*[a-z])(?=.*[A-Z])[0-9a-zA-Z]{8,}$/';

    if(!preg_match($rePassword, $password)) {

        $errors["password"] = $passwordMessage;

    }
}

```

```

    if(!filter_var($email, FILTER_VALIDATE_EMAIL)) {

        $errors["email"] = $emailMessage;

    }

    if(empty($errors)) {

        $password = sha1($password);

        $upit = "SELECT u.*, r.role FROM users u INNER JOIN roles r ON
u.id_role = r.id

                WHERE email = :email AND password = :pass";

        $selectedUser = bindAndSelect($conn, $upit, [

            "email" => $email,

            "pass" => $password

        ], true);

        if($selectedUser) {

            $_SESSION['user'] = $selectedUser;

        } else {

            $_SESSION['greske'] = ['email' => 'Email or password are not
valid'];

        }

    } else {

        $_SESSION['greske'] = $errors;

    }

}

header("Location: ../index.php");

```

3.4.5. register.php

```
<?php

session_start();

require_once "../../database.php";

require "utilities.php";

if(isset($_POST['register'])) {

    $email      = $_POST['reg-email'];

    $password    = $_POST['reg-password'];

    $confirm     = $_POST['reg-confirm'];

    $errors      = [];

    $passwordMessage = "A password must have at least one digit, at least
one uppercase char, lowercase chars and it should be at least 8 chars
long";

    $emailMessage = "Your email must be of a valid format";

    $confirmMessage = "Your passwords are not matching";

    $uniqueMessage = "This email is taken";

    if($password !== $confirm) {

        $errors["reg_confirm"] = $confirmMessage;

    } else {

        $rePassword = '/^(?=.*\d)(?=.*[a-z])(?=.*[A-Z])[0-9a-zA-Z]{8,}$/';

        if(!preg_match($rePassword, $password)) {

            $errors["reg_password"] = $passwordMessage;

        }

    }

}
```

```
}  
  
}  
  
if(!filter_var($email, FILTER_VALIDATE_EMAIL)) {  
  
    $errors["reg_email"] = $emailMessage;  
  
}  
  
if(empty($errors)) {  
  
    $password = sha1($password);  
  
    $upit = "INSERT INTO users (email, password, id_role) VALUES (:email,  
:pass, 2)";  
  
    $inserted = bind($conn, $upit, [  
  
        "email" => $email,  
  
        "pass" => $password  
  
    ]);  
  
    $id = $conn->lastInsertId();  
  
    $user = (object) [  
  
        "id" => $id,  
  
        "email" => $email,  
  
        "password" => $password,  
  
        "id_role" => 2  
  
    ];
```

```
if($inserted) {

    $_SESSION['user'] = $user;

} else {

    $errors['reg_email'] = $uniqueMessage;

    $errors["turn_modal"] = true;

    $_SESSION['greske'] = $errors;

}

} else {

    $errors["turn_modal"] = true;

    $_SESSION['greske'] = $errors;

}

}

header("Location: ../index.php");
```

3.4.6. logout.php

```
<?php

session_start();

unset($_SESSION['user']);

session_destroy();

header("Location: ../index.php");
```

3.4.7. contact.php

```
<?php

session_start();
```

```
require_once "utilities.php";

if(isset($_POST['contact'])) {

    define("TO", "oljawildchild@gmail.com");

    $email = $_POST['contact-email'];

    $message = $_POST['contact-message'];

    $errors = [];

    $emailMess = "You must enter a valid format for email address";

    $messageMess = "Your message must have at least 10 characters";

    if(!filter_var($email, FILTER_VALIDATE_EMAIL)) {

        $errors["email"] = $emailMess;

    }

    if(strlen($message) < 10) {

        $errors["message"] = $messageMess;

    }

    if(empty($errors)) {

        $subject = "MyComicsList contact";

        $header = "From: $email";

        mail(TO, $subject, $message, $header);
```



```

        $_SESSION['contactSuccess'] = ['success' => 'Administrator will
respond as soon as possible'];

    } else {

        $_SESSION['contactErrors'] = $errors;

    }

}

header("Location: /index.php?page=contact");

```

3.4.8. addNewComic.php

```

<?php

session_start();

include "../php/utilities.php";

include "../../database.php";

function updateRecord($conn, $comicId, $name, $desc, $issues, $tags) {

    $upit = "UPDATE comics SET name=:name, description=:desc,
issues=:issues WHERE id=:id";

    bind($conn, $upit, [

        "name" => $name,

        "desc" => $desc,

        "issues" => $issues,

        "id" => $comicId

    ]);
}

```

```
$oldState = bindAndSelect($conn,

    "SELECT id_sub_filter as id FROM comics_sub_filters WHERE
id_comic=:id", ["id" => $comicId], false);

$oldStateIds = array_map(function($e) {

    return $e->id;

}, $oldState);

$toDelete = [];

$insert = [];

foreach($oldStateIds as $id) {

    if(!in_array($id, $tags)) $toDelete[] = $id;

}

foreach($tags as $newId) {

    if(!in_array($newId, $oldStateIds)) $insert[] = $newId;

}

foreach($toDelete as $idSub) {

    $query = "DELETE FROM comics_sub_filters WHERE id_comic=:comic AND
id_sub_filter=:sub";

    bind($conn, $query, ["comic" => $comicId, "sub" => $idSub]);

}

foreach($insert as $idSub) {
```

```

        $query = "INSERT INTO comics_sub_filters(id_comic,id_sub_filter)
VALUES (:idComic, :subFilter)";

        bind($conn, $query, ["idComic" => $comicId, "subFilter" => $idSub]);

    }

}

function insertRecord($conn, $name, $desc, $issues, $picPor, $tags) {

    $upit = "INSERT INTO comics(name, description, issues) VALUES (:name,
:desc, :issues)";

    $inserted = bind($conn, $upit, [

        "name" => $name,

        "desc" => $desc,

        "issues" => $issues

    ]);

    if($inserted) {

        $idComic = $conn->lastInsertId();

        if(uploadPicture($picPor, $idComic, $conn, 'resize_image_by_height',
355)) {

            $upit = "INSERT INTO comics_sub_filters(id_comic,id_sub_filter)
VALUES (:idComic, :subFilter)";

            foreach($tags as $tag) {

                bind($conn, $upit, [

                    "idComic" => $idComic,

                    "subFilter" => $tag

                ]);

            }

        }

    }

}

```

```

    }

    $_SESSION['upload'] = "Successfully inserted comic!";

} else $_SESSION['upload'] = "Error, not inserted";

}

}

```

```

if(isset($_POST['insert'])) {

    $name      = trim($_POST['comicName']);

    $desc      = trim($_POST['desc']);

    $issues    = $_POST['issues'];

    $tags      = isset($_POST['tags']) ? $_POST['tags'] : [];

    $picPor    = $_FILES['comic-por-pic'];

    $method    = $_POST['_method'];

    $errors    = [];

    $nameError = "Every first letter of a word must be a capital letter";

    $descError = "You need to write more than 10 characters";

    $issuesError = "Number of issues cannot be zero or less";

    $tagsError = "You must choose at least one subfilter";

    $reName =
'/^[A-Z][a-z]{2,48}(\s([A-Z][a-z]{1,48}|[0-9]{1,4}(\\.)*))*$/';

```

```
if(!preg_match($reName, $name)) {

    $errors['comicName'] = $nameError;

}

if(strlen($desc) < 10) {

    $errors['description'] = $descError;

}

if($issues < 1 || $issues > 1000) {

    $errors['issues'] = $issuesError;

}

if(empty($tags)) {

    $errors['tags'] = $tagsError;

}

if($method == "POST") {

    validatePicture($picPor, $errors, "comic-por-pic");

}

if(empty($errors)) {

    if($method == "UPDATE") {

        $id = $_POST['comicID'];

        updateRecord($conn, $id, $name, $desc, $issues, $tags);

    }

}
```

```
    } else {  
  
        insertRecord($conn, $name, $desc, $issues, $picPor, $tags);  
  
    }  
  
    } else {  
  
        $_SESSION['comicErrors'] = $errors;  
  
    }  
  
    header("Location: ../index.php?page=panel");  
  
} else {  
  
    header("Location: ../index.php");  
  
}
```

3.5. AJAX logika

3.5.1. comics.php

```
<?php  
  
session_start();  
  
require_once "../../database.php";  
  
require_once "../php/utilities.php";  
  
  
  
define("PER_PAGE", 8);  
  
header("Content-type: application/json");  
  
  
  
  
$status = 200;
```

```
$userId = isset($_SESSION['user']) ? $_SESSION['user']->id : 0;

$loggedIn = isset($_SESSION['user']) ? true : false;

// PAGINACIJA

$page = 1;

if(isset($_POST['page'])) {

    $page = $_POST['page'];

}

$od = ($page-1) * PER_PAGE;

$koliko = PER_PAGE;

$limit = "LIMIT :od, $koliko";

$base_query = "SELECT DISTINCT c.*,p.path, p.alt, l.id_user FROM comics c
LEFT JOIN list l ON c.id = l.id_comic AND l.id_user=$userId
JOIN pictures p ON c.id = p.id_comic
JOIN comics_sub_filters cf ON c.id = cf.id_comic";

if(isset($_POST['ids'])) {

    $ids = $_POST['ids'];

    $page = 1;

    if(isset($_POST['page'])) {

        $page = $_POST['page'];

    }

}
```

```

$array = "(";

$count = count($ids);

$bindings = [];

foreach($ids as $index => $id) {

    $array .= ":a$index";

    $bindings["a$index"] = $id;

    if($index < $count - 1) {

        $array .= ", ";

    }

}

$array .= ")";

$stmt = $conn->prepare("SELECT COUNT(distinct csf.id_comic) AS num FROM
comics c

JOIN comics_sub_filters csf ON c.id = csf.id_comic

WHERE csf.id_sub_filter IN $array");

foreach($bindings as $index => &$val) {

    $stmt->bindParam("$index", $val, PDO::PARAM_INT);

}

$stmt->execute();

$count = $stmt->fetch()->num;

$pages = ceil($count / PER_PAGE);

$base_query .= " AND cf.id_sub_filter IN $array " . $limit;

try {

```

```
$stmt = $conn->prepare($base_query);

$stmt->bindParam(":od", $od, PDO::PARAM_INT);

foreach($bindings as $index => &$val) {

    $stmt->bindParam("$index", $val, PDO::PARAM_INT);

}

$stmt->execute();

$result = $stmt->fetchAll();


$final = [];

foreach($result as $row) {

    $novi = [];

    $novi["id"] = $row->id;

    $novi["name"] = $row->name;

    $novi["description"] = $row->description;

    $novi["issues"] = $row->issues;

    $novi["path"] = $row->path;

    $novi["alt"] = $row->alt;

    $novi["flag"] = $row->id_user;


    if($novi["flag"] == $userId) {

        $novi["flag"] = true;

    } else {

        $novi["flag"] = false;

    }

}
```

```
        $final[] = (object) $novi;
    }

    $resp = [

        "total" => $pages,

        "page"  => $page,

        "data" => $final,

        "loggedIn" => $loggedIn

    ];

    echo json_encode($resp);
} catch (Exception $e) {

    $status = 500;

}

} else {

    $count = $conn->query("SELECT COUNT(*) AS num FROM
comics")->fetch()->num;

    $pages = ceil($count / PER_PAGE);

    try {

        $stmt = $conn->prepare($base_query . " " . $limit);

        $stmt->bindParam(":od", $od, PDO::PARAM_INT);

        $stmt->execute();

        $result = $stmt->fetchAll();
```

```
$final = [];  
  
foreach($result as $row) {  
  
    $novi = [];  
  
    $novi["id"] = $row->id;  
  
    $novi["name"] = $row->name;  
  
    $novi["description"] = $row->description;  
  
    $novi["issues"] = $row->issues;  
  
    $novi["path"] = $row->path;  
  
    $novi["alt"] = $row->alt;  
  
    $novi["flag"] = $row->id_user;  
  
    if($novi["flag"] == $userId) {  
  
        $novi["flag"] = true;  
  
    } else {  
  
        $novi["flag"] = false;  
  
    }  
  
    $final[] = $novi;  
  
}  
  
$resp = [  
  
    "total" => $pages,  
  
    "page"  => $page,  
  
    "data" => $final,  
  
    "loggedIn" => $loggedIn
```

```
];  
  
echo json_encode($resp);  
  
} catch (Exception $e) {  
  
    $status = 500;  
  
}  
  
}  
  
http_response_code($status);
```

3.5.2. addToList.php

```
<?php  
  
session_start();  
  
require "../database.php";  
  
$status = 404;  
  
if($_SERVER['REQUEST_METHOD'] == 'POST') {  
  
    $status = 403;  
  
    if(isset($_SESSION['user'])) {  
  
        $status = 400;  
  
        $idUser = isset($_SESSION['user']) ? $_SESSION['user']->id : null;  
  
        if(isset($_POST['stripId'])) {  
  
            $idStrip = $_POST['stripId'];
```

```
$status = 201;

if(!empty($idUser)) {

    $query = "INSERT INTO list(id_user, id_comic)

        VALUES (:id_kor, :id_com)";

    $stmt = $conn->prepare($query);

    $stmt->bindParam(":id_kor", $idUser);

    $stmt->bindParam(":id_com", $idStrip);

    $stmt->execute();

    echo json_encode(["success" => true]);

}

}

}

}

http_response_code($status);
```

3.5.3. removeFromList.php

```
<?php

session_start();

require "../../database.php";

$status = 404;

if($_SERVER['REQUEST_METHOD'] == 'POST') {

    $status = 403;

    if(isset($_SESSION['user'])) {

        $status = 400;

        $idUser = isset($_SESSION['user']) ? $_SESSION['user']->id : null;

        if(isset($_POST['stripId'])) {

            $idStrip = $_POST['stripId'];

            $status = 204;

            if(!empty($idUser)) {

                $query = "DELETE FROM list WHERE id_user = :id_user AND id_comic
= :id_comic";

                $stmt = $conn->prepare($query);

                $stmt->bindParam(":id_user", $idUser);

                $stmt->bindParam(":id_comic", $idStrip);

                $stmt->execute();

            }

        }

    }

}
```

```

    }

    }

}

```

```
http_response_code($status);
```

3.5.4. getComic.php

```
<?php
```

```
session_start();
```

```
require "../../database.php";
```

```
require "../php/utilities.php";
```

```
$status = 404;
```

```
if($_SERVER['REQUEST_METHOD'] == "POST") {
```

```
    $status = 403;
```

```
    if($_SESSION['user']->role === 'administrator')
```

```
        $status = 400;
```

```
        if(isset($_POST['id'])) {
```

```
            $id = $_POST['id'];
```

```
            try {
```

```
                $status = 500;
```

```
                $comic = bindAndSelect($conn, "select c.*, p.path from comics c
join pictures p on c.id = p.id_comic where c.id=:id", ["id" => $id],
true);
```

```
$filters = bindAndSelect($conn,

    "select * from sub_filters join comics_sub_filters csf

        on sub_filters.id = csf.id_sub_filter where

csf.id_comic=:id", ["id" => $comic->id], false);

if($comic) {

    $status = 200;

    if(!$filters) $filters = [];

    echo json_encode([

        "comic" => $comic,

        "filters" => $filters

    ]);

}

} catch(Exception $e) {

    $status = 500;

}

}

}

http_response_code($status);
```


3.5.5. deleteComic.php

```
<?php

session_start();

require "../../database.php";

require "../php/utilities.php";

function obrisiSliku($putanja) {

    $publicPath = relativeToProjectOS("public/$putanja");

    if(file_exists($publicPath)) {

        unlink($publicPath);

    }

}

$status = 404;

if($_SERVER['REQUEST_METHOD'] == 'POST') {

    $status = 403;

    if(true) {

        $status = 400;

        if(isset($_POST['id'])) {

            $status = 204;

            $id = $_POST['id'];

            $conn->beginTransaction();

            try {

                $upit = "SELECT path FROM pictures WHERE id_comic=:id";

                $slike = bindAndSelect($conn, $upit, ["id" => $id], false);
```

```
$upit = "DELETE FROM pictures WHERE id_comic=:id";

bindParam($conn, $upit, ["id" => $id]);

$upit = "DELETE FROM list WHERE id_comic=:id";

bindParam($conn, $upit, ["id" => $id]);

$upit = "DELETE FROM comics_sub_filters WHERE id_comic=:id";

bindParam($conn, $upit, ["id" => $id]);

$upit = "DELETE FROM comics WHERE id=:id";

bindParam($conn, $upit, ["id" => $id]);

foreach($slike as $slika) {

    obrisiSliku($slika->path);

}

} catch (PDOException $e) {

    $conn->rollBack();

    echo $e->getMessage();

} catch (Exception $e) {

    $conn->rollBack();

    echo $e->getMessage();

}

$conn->commit();

}

}

http_response_code($status);
```

3.5.6. addVote.php

```
<?php

session_start();

require "../../database.php";

require "../php/utilities.php";

$status = 404;

header('Content-Type: application/json');

if($_SERVER['REQUEST_METHOD'] == 'POST') {

    $status = 403;

    if(isset($_SESSION['user'])) {

        $status = 400;

        $idUser = isset($_SESSION['user']) ? $_SESSION['user']->id : null;

        if(isset($_POST['votedFor'])) {

            $votedFor = $_POST['votedFor'];

            if($votedFor > 0) {

                $status = 200;

                if(!empty($idUser)) {

                    $query = "INSERT INTO votes(id_user, id_comic)

                        VALUES (:id_user, :id_comic)";

                    try {

                        bindWithException($conn, $query, [

                            "id_user" => $idUser,
```


3.5.7. removeVote.php

```
<?php

session_start();

require "../../database.php";

require "../php/utilities.php";

$status = 404;

header('Content-Type: application/json');

if($_SERVER['REQUEST_METHOD'] == 'GET') {

    $status = 403;

    if(isset($_SESSION['user'])) {

        $status = 400;

        $idUser = isset($_SESSION['user']) ? $_SESSION['user']->id : null;

        $status = 200;

        if(!empty($idUser)) {

            try {

                $query = "DELETE FROM votes WHERE id_user = $idUser";

                $conn->query($query);

                $upit = "SELECT c.name,p.path, p.alt, COUNT(*) AS votes FROM
votes v JOIN comics c ON v.id_comic = c.id JOIN pictures p ON c.id =
p.id_comic GROUP BY c.name, p.path, p.alt ORDER BY votes DESC LIMIT 0,3";

                $result = selectMultipleRows($conn, $upit);

                $resp = [
```

```

        "results" => $result,

        "poruka" => "You can vote again!"

    ];

    echo json_encode($resp);
} catch(PDOException $e) {

    echo json_encode(["poruka" => "Something went wrong"]);

}

}

}

}

http_response_code($status);

```

3.5.8. comicStats.php

```

<?php

require "../../database.php";

require "../php/utilities.php";

session_start();

$stats = 403;

if(isset($_SESSION['user']) && $_SESSION['user']->role ==
'administrator') {

```

```

$status = 200;

$upit = "select c.name, count(*) as vote_num, (select count(*) from
comics_sub_filters csf

        where csf.id_comic=c.id) as favorite_count from votes v

        join comics c on v.id_comic=c.id

        group by c.name

        limit 5";

$result = selectMultipleRows($conn, $upit);

echo json_encode($result);
}

```

```
http_response_code($status);
```

3.6. JavaScript

3.6.1. comics.js

```

$(document).ready(function() {

    filters.init();

    filters.getAllComics();

});

function addLink(id) {

    return `<a href='#' data-id='${id}' class='btn-style bs-white add-to-list
add'>ADD</a>`;
}

```

```
}
```

```
function deleteLink(id) {  
  
    return `  
}
```

```
var myList = (function() {  
  
    function init() {  
  
        var add = $(".add");  
  
        var remove = $(".remove");  
  
        console.log("Reinit?");  
  
        add.click(handleAdd);  
  
        remove.click(handleRemove);  
  
    }  
  
}
```

```
function handleRemove(e) {  
  
    e.preventDefault();  
  
    var id = $(this).data("id");  
  
    ajaxPost("ajax/removeFromList.php", {  
  
        stripId: id  
  
    },
```



```
() => {

    var insertNode = $(addLink(id));

    insertNode.click(handleAdd);

    $(this).replaceWith(insertNode);

},

(status) => {

    console.log(status);

});

}

function handleAdd(e) {

    e.preventDefault();

    var id = $(this).data("id");

    ajaxPost("ajax/addToList.php", {

        stripId: id

    },

    () => {

        var deleteNode = $(deleteLink(id));

        deleteNode.click(handleRemove);

        $(this).replaceWith(deleteNode);

    },

    (status) => {

        console.log(status);

    });

}
```

```
}

return {

  init: init

}

})();

var filters = (function() {

  function init() {

    $(".sub-items").hide();

    $(".title i").css("transition", "all ease 450ms");

    $(".sub-items").each((i, el) => $(el).data('height', $(el).height()));

    $(".title").click(handleClick);

    $(".filter").click(handleFilterClick);

    var urlString = window.location.href;

    var url = new URL(urlString);

    var defaultFilter = url.searchParams.get("filter");

    if(defaultFilter) {

      var found = null;

      $(".search-text").each((i, el) => {

        if($(el).text() == defaultFilter) {

          found = $(el);

          return;

        }

      });

    }

  }

});
```

```
    }

    });

    if(found) {

        var title = found.closest('.sub-items').siblings('.title');

        title.click();

        found.click();

        var half = $(window).height() / 2;

        setTimeout(function() {

            $("html, body").animate({

                scrollTop: found.offset().top - half

            }, 750);

        }, 450);

    }

}

}

}

function handleFilterClick() {

    var ids = $(".filter")

        .filter(function() { return this.checked; })

        .map(function() { return $(this).val(); })

        .toArray();

    var data = {ids: ids};
```

```
ajaxPost("ajax/comics.php", data,

  (resp) => {

    iscrtajSve(resp);

    myList.init();

    iscrtajNav(resp);

  },

  (status) => {

    console.log(status);

  }

);

}

function getAllComics(page) {

  var page = page || 1;

  var ids = $(".filter")

    .filter(function () { return this.checked; })

    .map(function () { return $(this).val(); })

    .toArray();

  ajaxPost("ajax/comics.php", {

    page: page,

    ids: ids

  },

  (resp) => {

    iscrtajSve(resp);
```

```
    myList.init();

    iscrtajNav(resp);

},

(status) => {

    console.log(status);

});

}

function iscrtajNav(resp) {

    var div = $(".comics-control");

    div.html("");

    for(let i=0; i<resp.total;i++) {

        let link = $('<a href="#">${i+1}</a>');

        link.css(

            {

                "fontSize": "3rem",

                "margin": "1em",

                "color": "#666A86"

            });

        link.click(function(e) {

            e.preventDefault();

            getAllComics(i+1);

        });

        div.append(link);

    }

}
```

```
}
```

```
function resolveFlag(flag, id) {  
    return flag ? deleteLink(id) : addLink(id);  
}
```

```
function iscertajJednog(comic, loggedIn) {  
    return `

<br>  
        <img src='${comic.path}' alt='${comic.alt}' /><br>  
        <h2>${comic.name}</h2><br>  
        <div class='aid ar-btn'><br>  
            ${loggedIn ? resolveFlag(comic.flag, comic.id):''}<br>  
        </div><br>  
        <p class='scroll'>${comic.description}</p><br>  
        <div class='info'><br>  
            <p><strong>Issues/Chapters: </strong>${comic.issues}</p><br>  
        </div><br>  
    </div>`;<br>  
}


```

```
function iscertajSve(resp) {  
    var html = "";  
    var loggedIn = resp.loggedIn;  
    resp.data.forEach((comic) => {
```

```
    html += iscrivitiJednog(comic, loggedIn);

  });

  $(".comics").html(html);
}

function handleClick() {

  var elem = $(this).parent().find('.sub-items');

  var toHeight = elem.data('height');

  var carret = $(this).find('i');

  elem.finish();

  if(!elem.data("expanded")) {

    elem.css("height", 0).show();

    elem.animate({height: toHeight + "px"}, 450);

    elem.data("expanded", true);

    carret.addClass('go');

  } else {

    elem.animate({height: 0}, 450);

    elem.data("expanded", false);

    carret.removeClass('go');

  }

}
```

```

return {

    init: init,

    getAllComics: getAllComics

};

})();

```

3.6.2. mainslider.js

```

$(document).ready(function() {

    slider.init();

    $("#slide-left").click(slider.slideRight);

    $("#slide-right").click(slider.slideLeft);

    $(".image-slider-wrap").on('touchmove', slider.handleTouch);

    $(".image-slider-wrap").on('touchstart', slider.handleTouchStart);

    $(".image-slider-wrap").on('touchend', slider.handleTouchEnd);

    $(window).on('resize', slider.handleResize);

});

var slider = (function () {

    var index, slideAmount, imagesShown, imageCount;

    var lastTouchX, touchedPos, sliderPosition;

    function init() {

        index = 0;

        lastPos = 0;

```



```
sliderPosition = 0;

imageCount = $(".image-slider img").length;

slideAmount = $(".image-slider img").eq(0).outerWidth(true);

length = $(".image-slider img").length;

$(".image-slider").css('transition', 'all ease 1s');

syncImageCount();

}
```

```
function handleTouchStart(e) {

    e.preventDefault();

    if(e.targetTouches.length == 1) {

        var touch = e.targetTouches[0];

        touchedPos = touch.pageX + sliderPosition;

    }

    $(".image-slider").css("transition", "");

}
```

```
function handleTouch(e) {

    e.preventDefault();

    if (e.targetTouches.length == 1) {

        var touch = e.targetTouches[0];

        var movement = touchedPos - touch.pageX;

        $(".image-slider").css({
```

```
        transform: `translateX(${ -movement }px)`
    });

    sliderPosition = movement;
}

}

function syncImageCount() {

    var imageWidth = $(".image-slider img").eq(1).outerWidth(true);

    var sliderWidth = $(".image-slider-wrap").width();

    imagesShown = Math.ceil(sliderWidth / imageWidth);
}

function handleResize() {

    syncImageCount();
}

function handleTouchEnd() {

    $(".image-slider").css('transition', 'all ease 1s');
}

function toMaxIndex() {

    var jumpIndex = imageCount - imagesShown;

    index = jumpIndex;

    var toSlide = index * slideAmount;
```

```
$(".image-slider").css({  
    transform: `translateX(${ -toSlide }px)`  
});  
  
lastPos = -toSlide;  
}  
  
function revertToZero() {  
  
    index = 0;  
  
    $(".image-slider").css({transform: "translateX(0)"})  
  
    lastPos = 0;  
}  
  
function slideLeft() {  
  
    if(index + imagesShown > imageCount - 1) {  
  
        revertToZero();  
  
        return;  
    }  
  
    index++;  
  
    var toSlide = index * slideAmount;  
  
    $(".image-slider").css({  
        transform: `translateX(${ -toSlide }px)`  
    });  
  
    lastPos = -toSlide;  
}
```

```
function slideRight() {

    if(index - 1 < 0) {

        toMaxIndex();

        return;

    }

    index--;

    var toSlide = index * slideAmount;

    $(".image-slider").css({

        transform: `translateX(${ -toSlide}px)`

    });

    lastPos = -toSlide;

}

return {

    init: init,

    slideLeft: slideLeft,

    slideRight: slideRight,

    handleTouch: handleTouch,

    handleTouchStart: handleTouchStart,

    handleTouchEnd: handleTouchEnd,

    handleResize: handleResize

}

}());
```

3.6.3. modal.js

```
$(document).ready(function () {

    initModal();

    if (window.modalOpen) {

        if(window.turnModal) {

            openModal('register');

        } else {

            openModal();

        }

    }

    $(".open-modal-login").click(function (e) {

        e.preventDefault();

        e.stopPropagation();

        openModal();

    });

    $(".open-modal-register").click(function (e) {

        e.preventDefault();

        e.stopPropagation();

        openModal('register');

    });

});
```

```
function initModal() {

    $("#register-btn").click((e) => {

        e.preventDefault();

        $(".modal .card").addClass('is-flipped');

    });

    $("#login-btn").click((e) => {

        e.preventDefault();

        $(".modal .card").removeClass('is-flipped');

    });

    $(".modal").find(".modal-exit").click(function () {

        if (window.modalOpen) {

            closeModal();

        }

    });

    $(document).click(function (e) {

        if ($ (e.target).closest('.modal').length

            || e.target == '.open-modal-login') return;

        if (window.modalOpen) {

            closeModal();

        }

    });

}
```

```
}
```

```
function closeModal() {

    window.modalOpen = false;

    $(".modal").removeClass("show");

    $(".modal").addClass('hide');

    $("#crnilo").remove();

}


function openModal(rotation) {

    window.modalOpen = true;

    $(".modal").removeClass('hide');

    var crnilo = $("<div id='crnilo'>");

    crnilo.css({

        display: "block",

        position: "fixed",

        width: "100%",

        height: "100%",

        background: "rgba(0, 0, 0, 0.6)",

        zIndex: 999

    });

    $(".modal").addClass("show");

    if(typeof rotation == 'undefined') {
```

```
    rotation = "login"
  }

  if(rotation == 'login') {

    if($('.modal .card').hasClass('is-flipped')) {

      $('.modal').on('animationend', () => $('.modal
.card').removeClass('is-flipped'))

    }

  } else if(rotation == 'register') {

    if(!$('.modal .card').hasClass('is-flipped')) {

      $('.modal').on('animationend', () => $('.modal
.card').addClass('is-flipped'))

    }

  }

  $("body").prepend(crnilo);
}
```

3.6.4. utils.js

```
function ajaxGet(url, cbSuccess, cbFailure) {

  __ajax(url, cbSuccess, cbFailure, "GET");

}

function ajaxPost(url, data, cbSuccess, cbFailure) {

  __ajax(url, cbSuccess, cbFailure, "POST", data);

}
```

```
function __ajax(url, cbSuccess, cbFailure, verb, data) {

    var data = data || {};

    $.ajax({

        url: url,

        dataType: "json",

        method: verb,

        success: cbSuccess,

        data: data,

        error: function(xhr, textStatus, msg) {

            cbFailure(xhr.status, xhr.responseText);

        }

    })

}
```

3.6.5. validations.js

```
$(document).ready(function() {

    loginValidation();

    registerValidation();

    handleVote();

    contactValidation();

});

function loginValidation() {
```

```
$( 'form[name="login-forma"]' ).submit(function (e) {

    var errors = {};

    var validno = {};

    if (!validateLoginParams(errors, validno)) {

        e.preventDefault();

        prikazGresaka(errors, $(this));

    }

});

}

function validateLoginParams(errors, validno) {

    var validations = [];

    var email = $("input[name='email']").val();

    var password = $("input[name='password']").val();

    validations.push(validateEmail(email, errors, validno));

    validations.push(validatePassword(password, errors, validno));

    return !validations.includes(false);

}

function registerValidation() {

    $( 'form[name="reg-forma"]' ).submit(function (e) {

        var errors = {};

        var validno = {};
```

```
        if (!validateRegistrationParams(errors, validno)) {

            e.preventDefault();

            prikazGresaka(errors, $(this));

        }

    });

}

function validateRegistrationParams(errors, validno) {

    var validations = [];

    var email      = $("input[name='reg-email']").val();

    var password = $("input[name='reg-password']").val();

    var confirm    = $("input[name='reg-confirm']").val();

    validations.push(validateEmail(email, errors, validno));

    validations.push(confirmPassword(password, confirm, errors));

    validations.push(validatePassword(password, errors, validno));

    return !validations.includes(false);

}

function contactValidation() {

    $('form[name="contact"]').submit(function (e) {

        var errors = {};

        var validno = {};

        if(!validateContactParams(errors, validno)) {
```

```
e.preventDefault();

prikazGresaka(errors, $(this));

}

});

}

function validateContactParams(errors, validno) {

    var validations = [];

    var email = $("input[name='contact-email']").val();

    var message = $("#contact-message").val();

    validations.push(validateEmail(email, errors, validno));

    validations.push(validateMessage(message, errors, validno));

    return !validations.includes(false);

}

function validateEmail(email, errors, validno) {

    var reEmail = /^[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,6}$/;

    if(!reEmail.test(email)) {

        errors.email = "You must enter a valid format for email address";

        return false;

    } else {

        validno.email = email;

        return true;

    }

}
```

```
}  
  
}  
  
function validatePassword(password, errors, validno) {  
  
    var rePass = /^(?=.*\d) (?=.*[a-z]) (?=.*[A-Z]) [0-9a-zA-Z]{8,}$/;  
  
    if(!rePass.test(password)) {  
  
        errors.password = "A password must have at least one digit, at least  
one uppercase char, lowercase chars and it should be at least 8 chars  
long";  
  
        return false;  
  
    } else {  
  
        validno.password = password;  
  
        return true;  
  
    }  
  
}  
  
function confirmPassword(password, confirm, errors) {  
  
    if(password != confirm) {  
  
        errors.confirm = "Your passwords are not matching";  
  
        return false;  
  
    } else {  
  
        return true;  
  
    }  
  
}
```

```
function validateMessage(message, errors, validno) {

    if(message.length < 10) {

        errors.message = "Your message must have at least 10 characters";

        return false;

    } else {

        validno.message = message;

        return true;

    }

}

function prikazGresaka(errors, forma) {

    if (errors.password) {

        forma.find('.errPass').html(errors.password);

        forma.find(".fa-key").css("color", "crimson");

    } else {

        forma.find('.errPass').html("");

        forma.find(".fa-key").css("color", "#333");

    }

}

if (errors.email) {

    forma.find('.errEmail').html(errors.email);

    forma.find(".fa-at").css("color", "crimson");

    forma.find('.errContactEmail').html(errors.email);

} else {
```

```
    forma.find('.errEmail').html("");

    forma.find(".fa-at").css("color", "#333");

    forma.find('.errContactEmail').html("");
}

if (errors.confirm) {

    forma.find('.errConfirm').html(errors.confirm);

    forma.find(".fa-unlock").css("color", "crimson");
} else {

    forma.find('.errConfirm').html("");

    forma.find(".fa-unlock").css("color", "#333");
}

if (errors.message) {

    forma.find('.errContactMsg').html(errors.message);
} else {

    forma.find('.errContactMsg').html("");
}

}

function results(resp) {

    var html = "";

    $(".vote-msg").html(resp.poruka);

    console.log(resp.poruka);

    if (resp.poruka == "You already voted.") {
```

```
    $(".click-here").show();

} else {

    $(".click-here").hide();

}

if(resp.results) {

    resp.results.forEach((result) => {

        html += iscrtajSliku(result);

    });

    $(".last-added").html(html);

}

}

function iscrtajSliku(jedna) {

    return `<figure class='l3-item'>

        <img src='${jedna.path}' alt='${jedna.alt}' />

        <figcaption class='l3-desc'>

            <p>${jedna.name}</p>

            <p>Votes: ${jedna.votes}</p>

        </figcaption>

    </figure>`;

}

function handleVote() {

    $(".click-here").hide();
```



```
$("#vote").change(function () {

    var value = Number($("#this").val());

    if (value > 0) {

        ajaxPost("ajax/addVote.php", {

            votedFor: value

        },

        (resp) => {

            results(resp);

        },

        (status, responseText) => {

            console.log(responseText);

        });

    }

});

$(".click-here a").click(function (e) {

    $(".vote-msg").html("");

    $(".click-here").hide();

    e.preventDefault();

    ajaxGet("ajax/removeVote.php",

    (resp) => {

        results(resp);

    },

    (status, responseText) => {
```

```
        console.log(responseText);  
    });  
});  
}
```

3.7. CSS

3.7.1. utils.css

```
@import  
url('https://fonts.googleapis.com/css?family=Fjalla+One|Raleway|Bangers');  
  
* {  
  
    box-sizing: border-box;  
  
    margin:0;  
  
    padding:0;  
  
}  
  
html {  
  
    font-size: 62.5%;  
  
}  
  
body {  
  
    color:#333;  
  
    font-family: 'Raleway', sans-serif;  
  
    background-image: url("../images/hypnotize.png");
```

```
background-repeat: repeat;

}

ul {

    list-style-type: none;

}

a {

    text-decoration: none;

}

button {

    cursor: pointer;

}

header {

    background: #666A86;

    font-family: 'Bangers', cursive;

}

h1, h2 {

    font-family: 'Bangers', cursive;

}
```

```
.warning {  
  color: #666A86;  
}
```

```
.relative {  
  position: relative;  
}
```

```
.absolute {  
  position: absolute;  
}
```

```
.just-center {  
  justify-content: center;  
}
```

```
.row-no-wrap {  
  display: flex;  
  flex-flow: row nowrap;  
}
```

```
.text-center {  
  text-align: center;  
}
```

```
.flex-col {  
  
  display: flex;  
  
  flex-flow: column wrap;  
  
}  
  
.align-center {  
  
  align-items: center;  
  
}  
  
.center {  
  
  justify-content: center;  
  
  align-items: center;  
  
}  
  
.flex-row {  
  
  display: flex;  
  
  flex-flow: row wrap;  
  
}  
  
/* ----- Modal ----- */  
  
.modal {  
  
  position: fixed;  
  
  min-height: 40rem;
```

```
height: 40rem;

width: 50%;

max-width: 40rem;

background: transparent;

left: 50%;

top: 50%;

transform: translate(-50%, -50%);

display: none;

overflow: hidden;

z-index: 1000;

perspective: 1200px;

border-radius: .3em;
}

.modal .card {

width: 100%;

height: 100%;

transition: transform ease-out 1s;

transform-style: preserve-3d;

position: relative;
}

.card.is-flipped {

transform: rotateY(180deg);
```

```
}
```

```
.card__face {  
  
  position: absolute;  
  
  width: 100%;  
  
  height: 100%;  
  
  color: white;  
  
  text-align: center;  
  
  font-weight: bold;  
  
  backface-visibility: hidden  
  
}
```

```
.card__face--front {  
  
  background: #666A86;  
  
}
```

```
.card__face--back {  
  
  background: #666A86;  
  
  transform: rotateY(180deg);  
  
}
```

```
.modal .modal-exit {  
  
  font-family: 'Raleway', sans-serif;  
  
  display: block;
```

```
position: absolute;

right: 10px;

top: -5px;

font-size: 3rem;

color: black;

cursor: pointer;

transition: all ease 150ms;
}
```

```
.modal .form-error {

color: white;

font-size: 1.3rem;

text-align: justify;
}
```

```
.modal .modal-exit:hover {

color: white;
}
```

```
.modal.show {

min-height: 0 !important;

animation-name: modal-in;

animation-duration: 600ms;

display: flex;
```

```
}
```

```
.modal.hide {  
  
  min-height: 0 !important;  
  
  animation-name: modal-out;  
  
  animation-duration: 600ms;  
  
  display: flex;  
  
  width: 0;  
  
  height: 0;  
}
```

```
@keyframes modal-out {  
  
  from {  
  
    width: 50%;  
  
    height: 40rem;  
  
  }  
  
  to {  
  
    width: 0;  
  
    height: 40rem;  
  
  }  
}
```

```
@keyframes modal-in {  
  
  from {
```

```
    width: 0;

}

to {

    width: 50%;

}

}

.modal h1 {

    text-align: center;

    letter-spacing: .2em;

    font-size: 2.3rem;

}

.modal form {

    width: 70%;

}

.modal form input {

    display: block;

    border: none;

    background: rgba(255, 255, 255, 0.4);

    border-radius: .2em;

    padding: 1.3em 1.3em 1.3em 3em;

    color: #333;
```

```
width: 100%;  
  
margin: 1em auto;  
  
height: 2em;  
  
position: relative;  
}
```

```
.fa-span {  
  
position: absolute;  
  
font-size: 2rem;  
  
color: #333;  
  
opacity: 0.7;  
  
top: 8px;  
  
left: 10px;  
}
```

```
.modal form input::-moz-placeholder {  
  
color: #333;  
  
letter-spacing: .2em;  
}
```

```
.modal form input::placeholder {  
  
color: #333;  
  
letter-spacing: .2em;  
}
```

```
.modal button {  
  
  margin-top: 1em;  
  
  border: none;  
  
  background: transparent;  
  
  color: white;  
  
  border: 2px solid white;  
  
}  
  
.modal button:hover,  
.modal button:focus {  
  
  border: none;  
  
  background: white;  
  
  color: #666A86;  
  
  border: 2px solid white;  
  
}  
  
/* ----- NAVIGATION ----- */  
  
.nav li {  
  
  display: inline-block;  
  
}  
  
.logo {
```

```
color:#666A86;

font-family: 'Raleway',

sans-serif;

text-transform: uppercase;

font-weight: 300;

letter-spacing: .5em;

font-size:2rem;

}

.logo:hover {

color:rgba(255,255,255,0.4);

}

.navigation a {

color:white;

display:block;

text-align:center;

font-size: 2rem;

padding: .7em 1em .7em 0;

margin-left: 1em;

}

.navigation ul {

padding: 1em 0;
```

```
overflow: hidden;
}

.navigation ul li {
  position: relative;
}

.navigation ul li:after {
  display: block;
  position: absolute;
  left: 1em;
  width: 70%;
  height: 5px;
  background: white;
  content: "";
  transform: translateY(11px);
  transition: transform ease .3s;
}

.navigation ul li:hover:after {
  transform: translateY(6px);
}

.log-reg {
```

```
background:#333333;

width:100%;

padding:2em;
}

.log-reg ul {

margin-left: auto;
}

@media only screen and (max-width: 600px) {

.log-reg ul {

margin: 2.5em 0 2.5em 0;
}

.navigation a {

font-size: 1.4rem;
}
}

.btn-style {

font-size:1.4rem;

font-family: 'Bangers', cursive;

padding: .7em;

margin-left: 1em;

text-align: center;
```

```
color:#666A86;

border: 2px solid #666A86;

}

.btn-style:hover,

.btn-style:focus {

background:#666A86;

color:#333;

}

.bs-white:hover, .bs-white:focus {

color:white;

}

/* ----- FOOTER ----- */

footer {

background:#333;

width:100%;

}

.footer-picture {

background-image: url("../images/footer-wallp.png");

padding: 8em 0;
```



```
background-repeat: no-repeat;

background-size: cover;

background-position: center;

background-position-y: 15%;

flex-wrap: wrap;

}
```

```
.footer-picture strong {

color: #7680c9;

}
```

```
.disclaimer {

color: white;

font-size: 1.6rem;

width: 12em;

margin-left: 5em;

align-self: flex-start;

}
```

```
.links a {

color: white;

font-size: 2rem;

}
```

```
.links a:hover, .links a:focus {  
  color:#7680c9;  
}
```

```
.links li {  
  padding-bottom: 1em;  
  border-bottom: 1px solid gray;  
  margin-bottom: 1em;  
  min-width: 20em;  
}
```

```
.disclaimer .fab {  
  font-size: 2.5rem;  
  color: #7680c9;  
  margin: 1em 0;  
  padding-right: .5em;  
}
```

```
.update-comic button.disabled {  
  cursor: not-allowed;  
}
```

```
.select2-container {  
  width:300px !important;
```

```
margin: .5em 0;

}

.select2-container--default
.select2-results__option--highlighted[aria-selected] {

background-color: #666A86 !important;

color:white !important;

}

.select2-results__options li {

font-size: 1.7rem;

}

@media only screen and (max-width: 475px) {

.disclaimer {

margin: 1em 2.5em;

}

}

@media only screen and (max-width:700px) {

.sf-article {

margin: 0 2em;

}

}
```

3.7.2. style.css

```
.slider {  
  
  background: #333;  
  
  width: 100%;  
  
  min-height: 30rem;  
  
  margin-top: 6.5em;  
  
  padding: 4.5em;  
  
  flex-wrap: nowrap;  
  
}  
  
.list-of-comics {  
  
  position: relative;  
  
}  
  
.sneak-peek {  
  
  position: absolute;  
  
  top: -3.5em;  
  
  left: -2.2em;  
  
  background: #666A86;  
  
  min-width: 22em;  
  
  text-align: center;  
  
  color: white;  
  
  padding: .5em;  
  
  font-size: 1.9em;
```

```
letter-spacing: .2em;
}
```

```
.slider .fas {
  color:white;
  font-size: 4rem;
  margin: .5em;
  cursor:pointer;
}
```

```
.image-slider-wrap {
  overflow: hidden;
  width: 80rem;
}
```

```
.image-slider {
  display: flex;
  align-items: flex-start;
}
```

```
.image-slider img {
  width: 200px;
  flex: 0 0 200px;
  margin: 0 .6em;
```

```
}
```

```
@media only screen and (max-width: 1300px) {  
  
  .sneak-peek {  
  
    position: absolute;  
  
    top: -2.3em;  
  
    left: 50%;  
  
    right: 50%;  
  
    transform: translate(-50%, -50%);  
  
  }  
}
```

```
@media only screen and (max-width: 900px) {  
  
  .image-slider-wrap {  
  
    width: 60rem;  
  
  }  
}
```

```
@media only screen and (max-width: 900px) {  
  
  .image-slider-wrap {  
  
    width: 40rem;  
  
  }  
}
```

```
@media only screen and (max-width: 500px) {

  .sneak-peek {

    min-width: 16.8em;

  }

  .image-slider-wrap {

    width: 20rem;

  }

}

/* ----- ADVERTSIZMENT FOR LIST/REGISTER ----- */

.list-adv {

  font-size: 2.3rem;

  padding: 4em 0;

}

.la-group {

  width: 60rem;

  text-align: justify;

}

.la-group p {

  margin: 0 1.5em 1em;
```

```
}
```

```
.links-lr a {  
  
  font-size: 2.5rem;  
  
  display: block;  
  
  margin-bottom: 1em;  
  
}
```

```
/* ----- LAST ADDED ----- */
```

```
.last-added img {  
  
  width:100%;  
  
  height: 100%;  
  
}
```

```
.list-adv h2 {  
  
  font-size:2.4rem;  
  
  color:#666A86;  
  
  letter-spacing: .1em;  
  
  margin-bottom:.7em;  
  
  font-family:"Raleway", sans-serif;  
  
  text-transform: uppercase;  
  
  letter-spacing:.2em;  
  
}
```

```
.l3-item {  
  
  width:180px;  
  
  position:relative;  
  
  margin-right:.5em;  
  
  overflow:hidden;  
  
}  
  
@-moz-document url-prefix() {  
  
  .l3-item {  
  
    height:100%;  
  
  }  
  
}  
  
.l3-desc {  
  
  position:absolute;  
  
  z-index:100;  
  
  bottom:0em;  
  
  left:0em;  
  
  right:0em;  
  
  color:white;  
  
  background: rgba(0, 0, 0, .55);  
  
  padding-bottom:5em;  
  
  transition: all ease-in-out 250ms;
```

```
}
```

```
.l3-desc {  
  
  transform: translateY(150%);  
  
}
```

```
.l3-item:hover .l3-desc {  
  
  transform: translateY(0);  
  
}
```

```
.the-rest {  
  
  font-family: 'Raleway', sans-serif;  
  
  margin-top: .7em;  
  
  font-size: 1.8rem;  
  
}
```

```
.the-rest a {  
  
  margin: 0;  
  
  font-size: 1.8rem;  
  
  display: inline;  
  
  color: #666A86;  
  
  font-weight: 700;  
  
}
```

```
.vote {  
  
  margin-top: 2em;  
  
}  
  
.click-here, .vote-msg {  
  
  margin-top: 1em;  
  
}  
  
.click-here a {  
  
  color: #666A86;  
  
  font-weight: 900;  
  
}  
  
@media only screen and (max-width: 810px) {  
  
  .last-added {  
  
    justify-content: center;  
  
  }  
  
  .l3-item {  
  
    width: 250px;  
  
    height: auto;  
  
    margin-bottom: 1em;  
  
  }  
}
```

```
.l3-desc {  
    width:250px;  
    padding-bottom: 7em;  
}  
  
.the-rest, .the-rest a {  
    font-size: 2.2rem;  
}  
}  
  
@media only screen and (max-width: 510px) {  
    .last-added {  
        display:flex;  
        flex-direction:column;  
    }  
  
    .l3-item {  
        width: 250px;  
        -moz-height: auto;  
        height:auto;  
        margin-bottom: 1em;  
    }  
}
```

```
/* ----- FILTER SECTION ----- */

.section-filter {

  background-color: rgba(102, 106, 134, 0.6);

  padding: 5em 0;

  align-items: flex-start;

}


.sf-article {

  width: 700px;

  text-align: center;

}


.sf-article h3 {

  padding-bottom: 2em;

  letter-spacing: .5em;

  color: white;

  text-shadow: 3px 4px 3px grey;

}


.section-filter a {

  color: black;

  font-size: 2rem;

  padding-right: .7em;
```

```
display: inline-block;

line-height: 1.5em;

}

.section-filter a:nth-child(even) {

font-size: 3rem;

}

.section-filter a:nth-child(5n+2) {

font-size: 2.7rem;

}

.filter:hover, .filter:focus {

color: white;

}

/* ----- ABOUT ME ----- */

.img-circle img {

border-radius: 50%;

width: 300px;

}

/* ---- ADMIN PANEL ---- */

.add-comic {
```

```
    font-size: 1.7rem;
}

.update-comic {
    align-self: flex-start;
}

.add-comic h1 {
    font-family: 'Raleway', sans-serif;
    font-size: 2.5rem;
    letter-spacing: .2em;
    text-transform: uppercase;
    margin: 1em 0;
}

.input-group {
    margin: 1em 0;
    width: 70%;
}

.input-group input,
.input-group textarea {
    width: 30rem;
    height: 3em;
```

```
}
```

```
.input-group textarea {
```

```
  height: 10em;
```

```
  max-height: 30em;
```

```
  max-width: 30em;
```

```
  min-width: 100%;
```

```
  min-height: 100%;
```

```
}
```

```
.input-group input[type='text'],
```

```
textarea {
```

```
  border: 1px solid #D7D7D7;
```

```
}
```

```
.input-group label {
```

```
  display: block;
```

```
  font-weight: 700;
```

```
  color: #666A86;
```

```
  padding-bottom: .7em;
```

```
}
```

```
.input-group p {
```

```
  padding-left: 1em;
```

```
margin: 1em;

}

.form-error {

color: crimson;

}

.change-btn {

display: block;

padding: 1.4em;

border: none;

background-color: #666A86;

color: white;

text-align: center;

margin: 1em 0;

font-weight: 700;

letter-spacing: .2em;

}

.change-btn:hover,

.change-btn:focus {

background-color: #333;

}
```

```
.fstMultipleMode .fstControls {  
  width: 15em !important;  
}  
  
.fstResultItem,  
.fstChoiceItem,  
.fstMultipleMode .fstQueryInput {  
  font-size: 1em !important;  
}  
  
.panel-navigation a {  
  font-size: 2.5rem;  
  margin: .5em 1em;  
  color: #666a86;  
  padding: 1em;  
}  
  
.chart-container {  
  width: 80em;  
  height: 50em;  
  margin: 0 auto;  
}  
  
.ct-label {
```

```
font-size: 2rem;
}

.ct-series-a .ct-bar {
  stroke: #6b0392 !important;
}

.ct-series-b .ct-bar {
  stroke: #666a86 !important;
}

.series-a,
.series-b {
  position: relative;
  font-size: 2.1rem;
}

.series-a:before,
.series-b:before {
  position: absolute;
  left: -20px;
  top: 6px;
  content: '';
  display: block;
```

```
width: 15px;

height: 15px;

}

.series-a:before {

background: #6b0392;

}

.series-b:before {

background: #666A86;

}

/* ----- BROWSE ----- */

.browse-container {

justify-content: center;

margin: 5em 0;

}

.heading {

letter-spacing: .2em;

font-size: 4rem;

color: #666A86;

}
```

```
.filters {  
  
  flex: 1 0 25%;  
  
  margin-left: 10em;  
  
}  
  
.filters ul {  
  
  font-size: 2.5rem;  
  
}  
  
.filters>ul>li {  
  
  padding: .5em;  
  
  display: flex;  
  
  flex-direction: column;  
  
}  
  
.filters>ul>li:not(:last-child) {  
  
  border-bottom: 1px solid #333;  
  
}  
  
.title {  
  
  color: #666A86;  
  
  padding: .5em 0;  
  
}
```

```
.filters span {  
  margin-right: auto;  
}  
  
.sub-items {  
  font-size: 2rem !important;  
  overflow: hidden;  
}  
  
.sub-items li {  
  margin-top: .5em;  
  margin-bottom: .75em;  
}  
  
.title {  
  display: flex;  
  cursor: pointer;  
}  
  
.title span {  
  margin-right: auto;  
}
```

```
.title i.go {  
  
  transform: rotate(180deg);  
  
}
```

```
.custom-checkbox {  
  
  position: relative;  
  
  cursor: pointer;  
  
  display: block;  
  
  -moz-user-select: none;  
  
  -webkit-user-select: none;  
  
  -ms-user-select: none;  
  
  user-select: none;  
  
  margin-bottom: .33em;  
  
  margin-top: .33em;  
  
}
```

```
.custom-checkbox input {  
  
  display: none;  
  
  opacity: 0;  
  
}
```

```
.checkmark {  
  
  position: absolute;  
  
  display: block;
```

```
width: 1.3em;

height: 1.3em;

background: transparent;

border: 1px solid #333;

right: .5em;

top: 0;

border-radius: 3px;
}

.custom-checkbox input:checked~.checkmark:after {

position: absolute;

content: "\f00c";

font-family: "Font Awesome 5 Free";

font-weight: 900;

left: .1em;

top: 1px;

color: #666A86;
}

.comics,

.wrapper-for-comics {

width: 100%;

font-size: 1.5rem;
}
```

```
.comic {  
  
  flex: 1 0 20%;  
  
  margin: 0 1em;  
  
}
```

```
.comic img {  
  
  width: 200px;  
  
  margin: 1em 0;  
  
}
```

```
.comic h2 {  
  
  font-size: 3rem;  
  
  font-family: 'Raleway', sans-serif;  
  
  margin-bottom: .5em;  
  
}
```

```
.scroll {  
  
  width: 100%;  
  
  height: 15em;  
  
  overflow: auto;  
  
}
```

```
.info {
```

```
font-size: 1.8rem;

padding: .5em 0;

color: #666A86;

}
```

```
.ar-btn {

margin: 1.5em 0;

}
```

```
.ar-btn a:first-child {

margin-left: 0;

}
```

```
.comics-control {

margin: 3em 0;

}
```

```
.comics-control a:hover,

.comics-control a:focus,

.comics-control a:active {

color: #333;

}
```

```
.one-row {
```

```
margin: 1em 0;

}

.one-row img {

width: 100px;

}

.contactSuccess {

display: inline-block;

margin: 1em 0;

color: #666a86;

font-weight: 900;

}

@media only screen and (max-width: 770px) {

.browse-container {

display: flex;

flex-flow: column wrap;

align-items: center;

}

.filters {

width: 100%;

margin: 0;

text-align: center;
```

```
}  
  
.comics,  
  
.wrapper-for-comics {  
    min-height: 80vmax;  
    min-width: 50vmax;  
}  
  
}
```