SILESIAN UNIVERSITY OF TECHNOLOGY

FACULTY OF AUTOMATIC CONTROL, ELECTRONICS AND COMPUTER SCIENCE

Internet Technologies – project work

Movies for everyone

Gliwice, January 2017

Table of Content

1. Introduction

Movies for everyone is a website addressed to people who like movies and cinema. On this site users can find information about many movies. The assumption of the project is that the portal should be created for users and be very comfortable for them.

Functionality is different for registered users and guests. Guests have access only to main page, wallpapers, categories and movie sites. Registered user can also add his favorite film categories in account settings. Based on them our website can display other users with similar taste and recommended films.

This website have a database of movies belong to multiple categories. One movie cans belong to few categories. Displaying favorite movies and users depends on favorite movie categories selected by user.

Our movies database will be often updated so users will not miss any new movie.
There it also The countdown to the launch of new movies for this purpose.

At the moment our home page looks like this:

2. Aim and scope of the project

The aim of the project is to create very good portal about movies. We expect that final website will contain:

- Ability to login and registration.
- Election of the preferred categories.
- Display proposed movies based on the selected category.
- Display users with similar movies preferences.
- Automatic wallpaper gallery.
- Menu that move when we scroll the site down
- The countdown to the launch of new movies
- Movies, categories and users database with relations many to many

This website can be use as database and the encyclopedia about movies. There is also system that can connect users with similar movie preferences.

To create this website fully functionality we had to use few technologies:

a)On the client side:

HTML(and. HyperText Markup Language) to create design, elements that contains every part of the page.

CSS(Cascading Style Sheets) to describing the presentation of a document written in a markup language.

JavaScript to add few functionalities e.g. automatic wallpaper gallery.

b)On the server side:

MySQL - Structured Query Language for operations on database

PHP – to create logic, connection with database and partial views shared between many .php files that include header, menu and end of the website.

In the future our project Movies for everyone have a possibility to become popular movies website and a site that can connect people. Later there can be implemented few new functionalities, for example communication system between users and option that allow users add new movies to the database. These functions may allow extend the website by the users.

3. Schedule

3.1.Schedule approved at the beginning

1. Project Schedule.
2. Main page design.
3. Create database.
4. Communication between main page and the other sites
5. Create wallpaper gallery and a clock
6. Preparing the login and registration mechanism.
7. Communication with database(Add proposed movies and users list)
8. Refined appearance of the website.
9. Corrections.

3.2. Schedule reflecting actual work

1. Project Schedule.
2. Main page design.
3. Create database.
4. Communication between main page and the other sites
5. Create wallpaper gallery.
6. Preparing the login and registration mechanism.
7. Communication with database(Add proposed movies and users list) and a clock.
8. Communication with database part 2 + Refined appearance of the website.
9. Corrections.

We have done all point of schedule but point 7 – communication with database took us a little more time. We had to solve several problems with this point. We implemented countdown clock in week 7 instead of 5 because we had to created sites based on database first. However now everything works fine and according to the plan.

4. Software and/or hardware implementation (correct section title)

4.1. Defining the problem

We decided to create website friendly for users. On the site users will be able to find information about the oldest and the newest movies. More precisely the website will display movie title, release date, short and long description, budget, director, writers and boxoffice and youtube trailer. Portal have to allow users to find movies and other users based on movie categories settings. We expect that our website will have registration system with database.

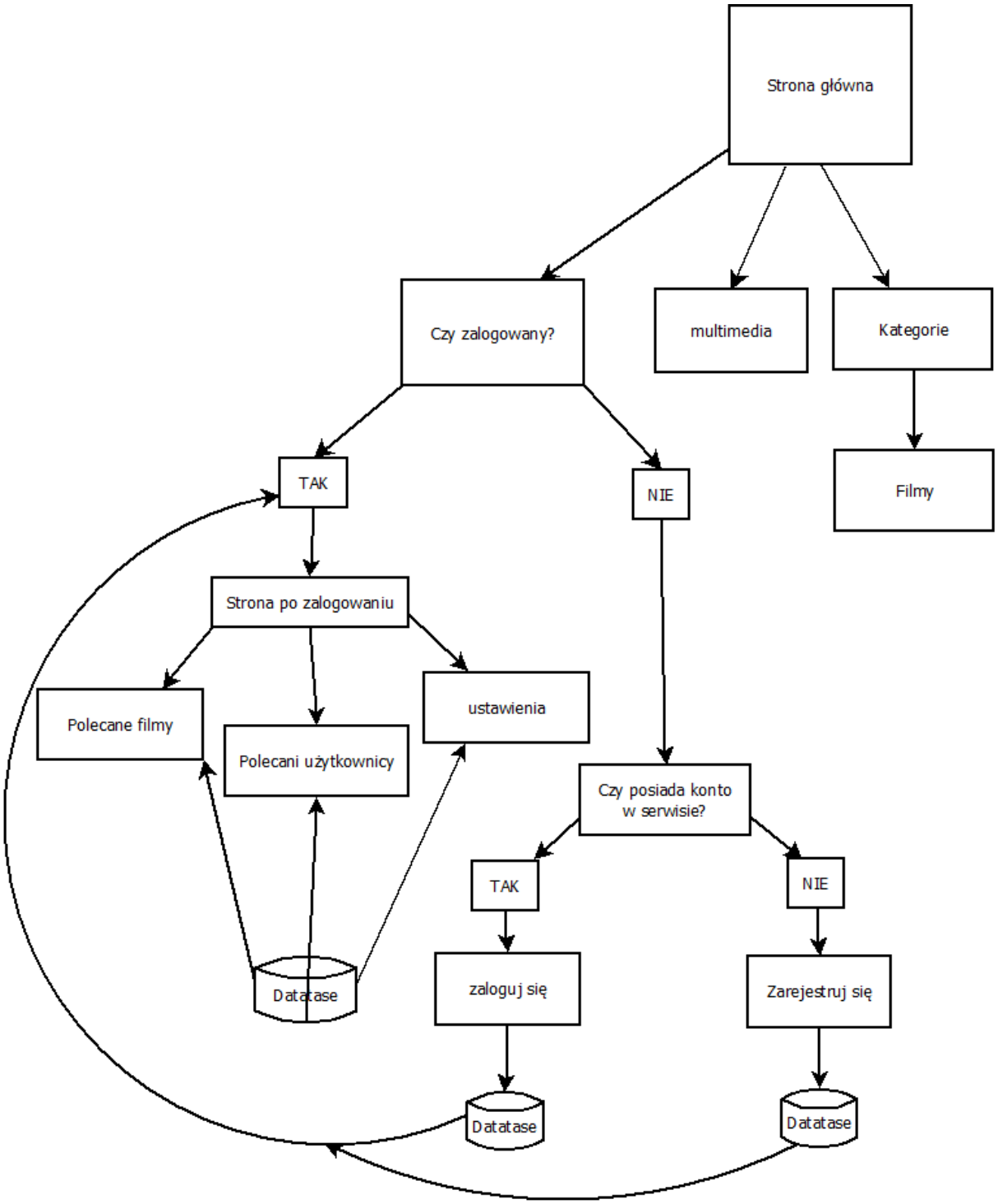
To create this project we had to use client site and server site language and Structured Query Language for operations on database.

4.2. Solution

We decided to build client site of our website based on HTML and CSS that can be used to create elements that contains every part of the page and describing presentation.
Server site was created in PHP to to create logic, connection with database and partial views. MySQL allow us to operate with database. As the text editor we used Notepad++ that is very friendly for users and allow us to work with many files at the same time.
We create database and server in the Xampp – free program that contains Apache local server, PHP language and very good database system.

4.3 Website block schema:

As mention at the beginning, our goal was creating a website about movies friendly for users with many options. We achieved it as shown in schema:



In the first step users enter to our website and has 2 options: continue using website as unlogged user or sign in. New users can create new account by entering appropriately Nickname, E-mail and password. They know that after this step they will have access to new cool functions like recommended movies and users. Whatever user choose(sing in or register) after this step website connect to database using SQL query and PHP language.
Next users have access to new site of website(only for members of our community). They have also access to standard function - can find information about movies.
Every user have access to free wallpapers that can be downloaded.
Logged user can sign out and become standard unlogged user without access to special functionalities.

4.4 Implementation

1. At the beginning we developed project schedule.

2.Main page design

At the second week we created main page design. To get that goal we used only HTML and CSS. We learnt how to divide file to 2 sections: head and body. Inside head we inserted meta data like keywords, content, words that will be helpful when somebody will try to find website in searcher and project title.
We decided to divide project to few divs. First div – “pojemnik” was the main div that contain other divs. Inside class “pojemnik” we created classes “header”, “menu” and “tlo”.
Header contain our website logo “Movies for everyone”.
Menu div contain menu. From menu user have access to wallpapers, sign in(inside sign in is registration) and movie categories where he can search information about movies.

The appearance of the website was created by using CSS. We have chosen this method because it provides best global appearance management. The CSS files contains definitions of objects and describes its features such as size, fonts, colors etc.
CSS language allow us easy changing website appearance. We can create multiple class that are described in one place. One class can be implement in many places and we have to only add .css file in html code to use our appearance template.

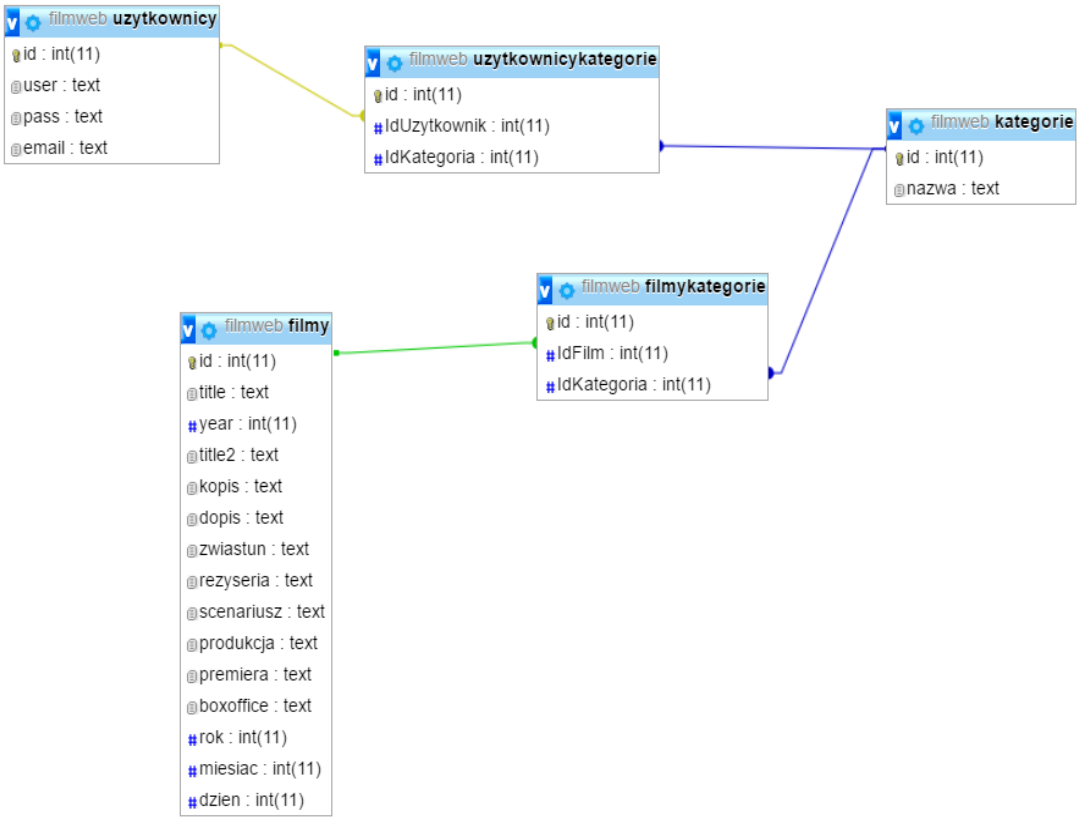
Example of css code that describe content class:

```
.content
{
        background-color: #13171A;
        opacity: 0.92;
        width: 1000px;
        margin-left: auto;
        margin-right: auto;
        text-align: justify;
        padding: 20px;
}
```

We used this class in almost all files. As you can see, few lines of CSS code can describe a large part of the page appearance.

3. Create database

Next we created database. To do that we used Xampp that allowed us to work with localhost.



It is our database schema. For full functionality we created 5 tables. Each element of every table has a unique identification number.

Users – that stores every registered user data(username,password and email),
Films – stores every film data. Each column of the table is displayed on the site for appropriately movie. For this purpose, we use PHP and MySQL queries.
Categories – it is a table, that contains movie categories.
UsersCategories - contains relations between User and his favorite categories. Every relation has unique id number. For example when one user like 2 categories, this table contains 2 elements that related user Id and Category Id.
FilmsCategories – this table have similar function but contains relations between Film id and Category id.

4. Communication between main page and the other sites

Then we create system that allow to communicate between main page and the other sites. First we created multiple href attribute and different .php files for each movie but that was bad idea. We decided to use PHP language and GET variable that allows us to create multiple sites based on one .php file that can load data from database and display that data on the site.

We used php function include() that allows us to divide our source code of every page, create few .php files that are included in all others: header.php, menu.php and stopka.php.

Example: <?php include('main/naglowek.php'); ?>

By using this simple thing all our files contain less amount of code.

5. Add multimedia content:

The In the next step we created wallpapers site inside multimedia content. We decided to create system that can change displayed wallpaper every few seconds. To that purpose we used JavaScript, more precisely jquery - small, and feature-rich JavaScript library that makes things like HTML document traversal and manipulation, event handling, animation.

To add jQuery librare we included this code inside script block:

```
<script src="http://code.jquery.com/jquery-1.11.2.min.js"></script>
```

After adding jQuery library we only had to write little code that can automatic change images and display them from project folder and insert to element of our website:

```
var file = "<img src=\"slajdy/slajd" + number + ".png\" />";

document.getElementById("slider").innerHTML = file;
```

6. Preparing the login and registration mechanism

At the 6 week we created login and registration system. To create this system we used PHP and MySQL queries.
When new user try to register he have to enter date and click “zarejestruj się”. But that is the beginning. When user try to enter data php cone check if everything is correct according to assumptions in code. If something is wrong system throw appropriate exception and display error message so user have to enter correct data.
If everything is ok system can connect to database, send data and create new record in Users table.

Login system works similar. First php have to connect to database and check if there is any user with the same name and password as the data entered in the login form.

Connecting with the database implemented by using this code:

```
require_once "connect.php";
try
	{
		$polaczenie = @new mysqli($host, $db_user, $db_password,
$db_name);
		if($polaczenie->connect_errno!=0)
			{
				throw new Exception(mysqli_connect_errno());
			}
	}
catch(Exception $e)
{
}

// Operations on database by using MySQL queries:
For example:

if($rezultat = @$polaczenie->query(
		sprintf("SELECT * FROM uzytkownicy WHERE user='%s'",
		mysqli_real_escape_string($polaczenie,$login))))
{
//code that is executing during user sign in
}

$polaczenie->close(); // Close connection with database
```

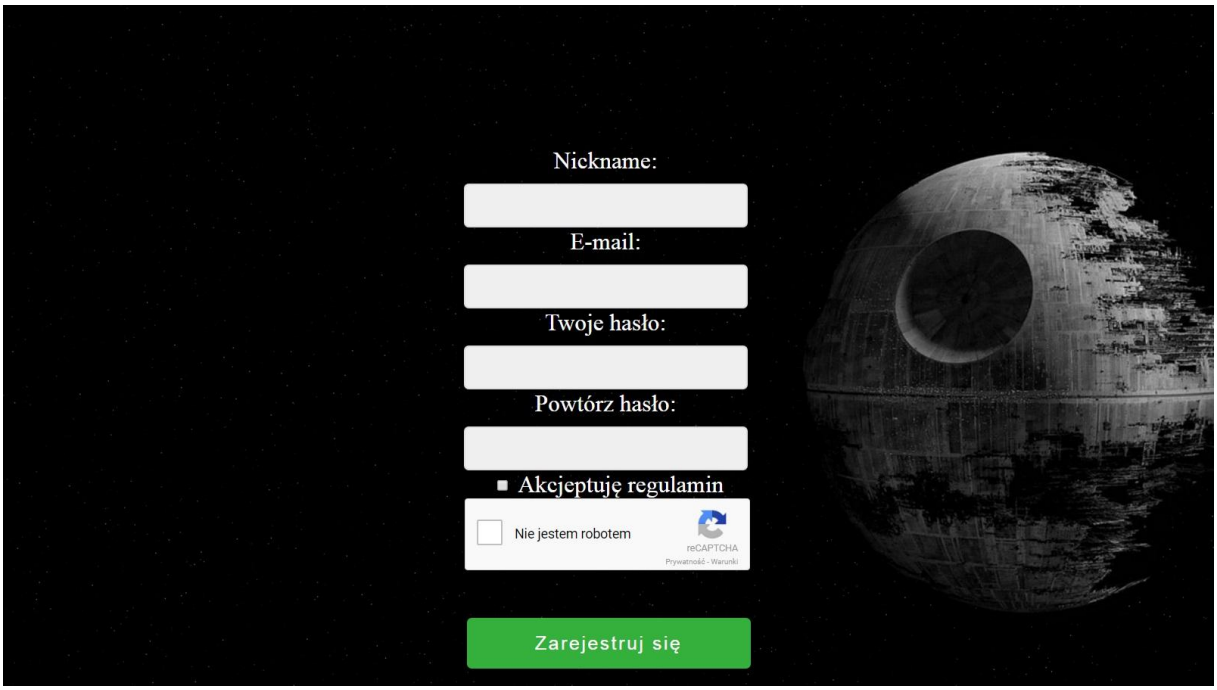
7. Communication with database

Next we created communication between website and database that allows us to display other users with similar taste and recommended films for logged user based on his settings – favorite movie categories. At the beginning we did not know how to implement this system and we had few problems, which are described in next part of the raport – “Problems during application development ” To achieve this goal we had to use PHP language and MySQL queries.
At this moment we used database tables with many-to-many relationship. That allow us to join appropriately table rows.
This part of project was the longest one to implement. It is described in the next part of the raport - “Problems during application development” .

4.4 Instruction for new users(How to use our website):

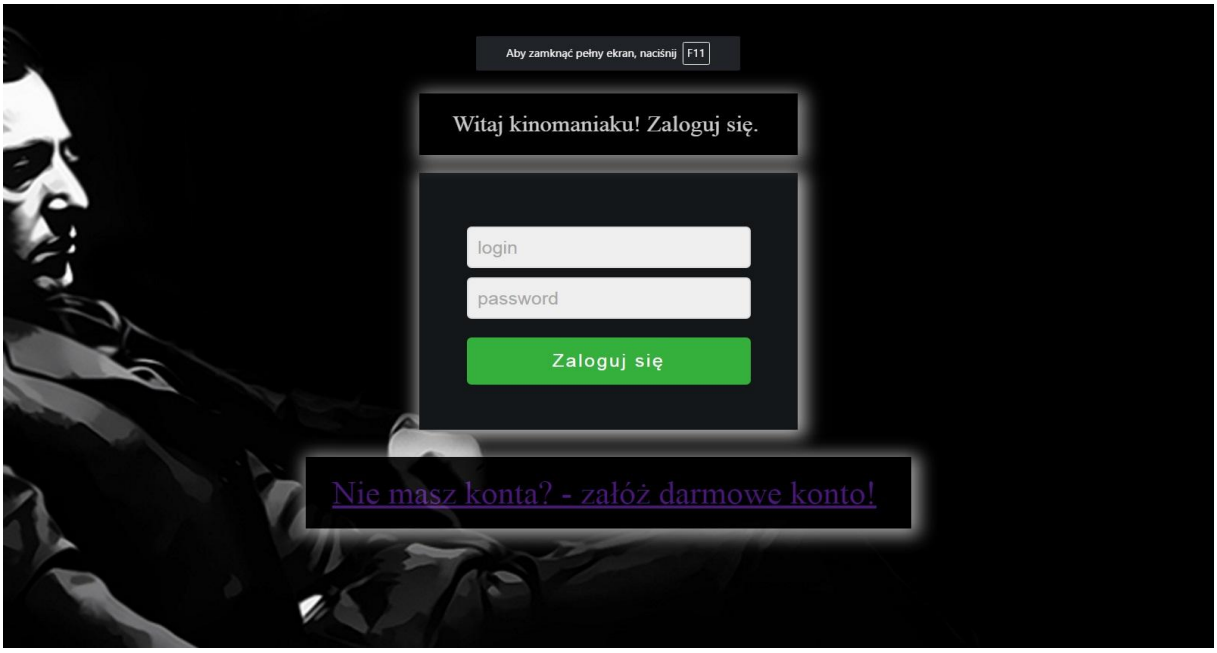
1. Join to us and create new account.

   First have to enter Your Nickname, E-mail and password. Our system will check if you have entered the correct data.



2. Sign in:

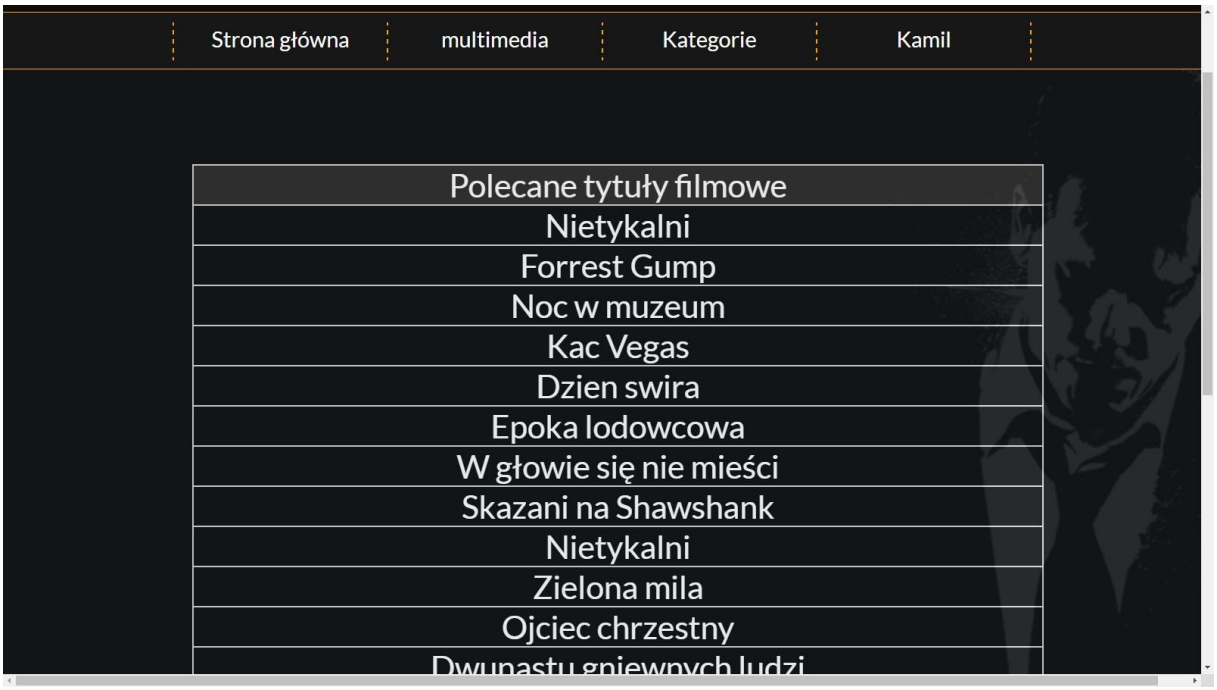
   Enter your Nickname and password and sign in to your account.



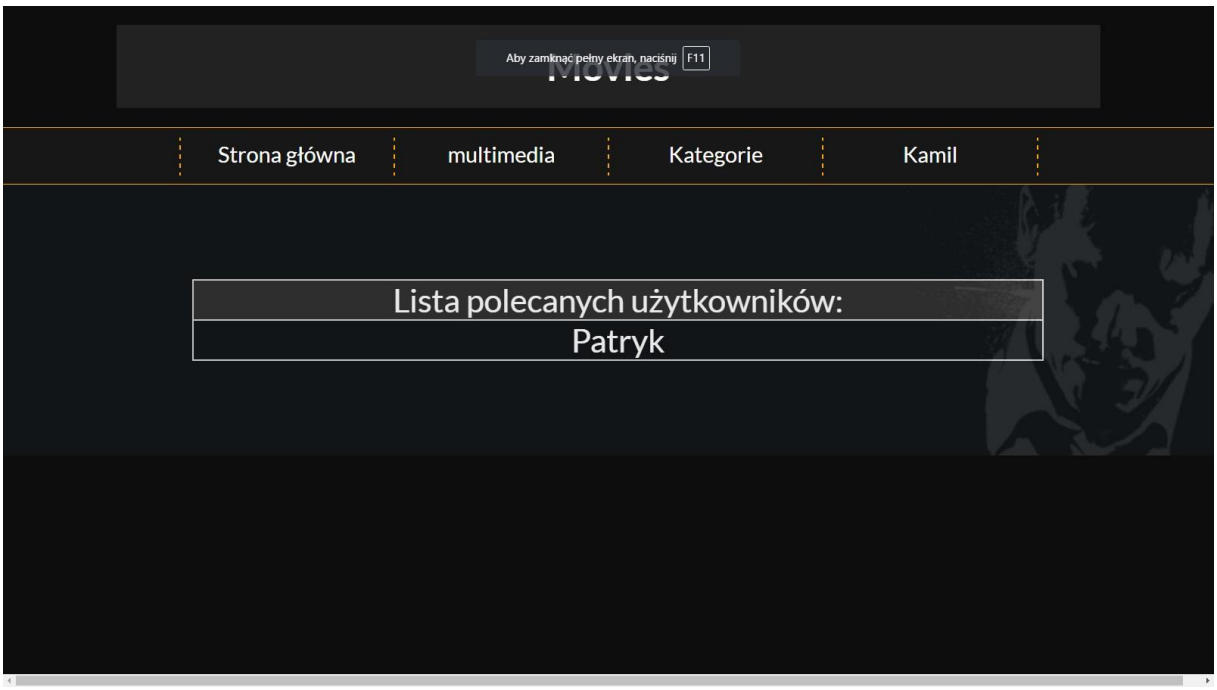
3. Set preferred film categories in settings.

   Based on your preferences our website can display other users with similar taste and recommended films.

4. Now you can find movies based on categories:



5. You can also find users with similar movie preferences:



6. You have also access to our movie database.

   Find your favorite movies and learn more about them.

7. Select movie title and display all information :

4.5 Problems during application development

We had a problem with MySQL queries. We did not know how to write query that can relete 3 tables: Films, FilmsCategories and Users. We had to display movies from category, which is liked by logged user.
Problem was solved by use appropriately nested query witch keyword IN() which can consist another query inside brackets. Results of this query are arguments for the main query.
We solve this by using this query related to variable zapytanietxt.

```
$zapytanietxt = "SELECT f.title
FROM filmy f INNER JOIN filmykategorie fk
ON f.id=fk.IdFilm
where idKategoria IN(
SELECT distinct uk.IdKategoria
FROM uzytkownicy u INNER JOIN uzytkownicykategorie uk ON u.id=uk.IdUzytkownik
where u.id = '$userNowId'
) ";
```

As you can see first we take Category id “idKategoria” from table named “uzytkownicykategorie” that contain categories id related to users id. We choose categories related to user that is currently logged in. We can choose many categories. Then the query find all movies title that are belong to chosen movie category. You can see this dependence in our database diagram.

To display featured users we used very similar MySQL query:

```
$zapytanietxt = "SELECT distinct u.user
FROM uzytkownicy u INNER JOIN uzytkownicykategorie uk
ON u.id=uk.IdUzytkownik
where idKategoria IN(
SELECT distinct uk.IdKategoria
FROM uzytkownicy u INNER JOIN uzytkownicykategorie uk ON u.id=uk.IdUzytkownik
where u.id = '$userNowId'
)
AND u.user != '$userNow'";
```

Then we had another problem - we did not know how to display all movies sites from one php file. We solved this problem by using keyword GET. We pass the unique variable for every movie in the URL address for movie.php file. Inside this file we can use GET variable in query.

Below you can see our solution:

```
<?php


echo "METODA GET:<br>";
    $zmienna = $_GET['zmienna'];
    echo "Zmienna: ".$zmienna;

    echo "<br><br><br><br>";


$userNow = $_SESSION['user']; //aktualny user
$userNowId = $_SESSION['id'];  //aktualny user id

ini_set("display_errors", 0);
require_once 'dbconnect.php';
$polaczenie = mysqli_connect($host, $user, $password);
mysqli_query($polaczenie, "SET CHARSET utf8");
mysqli_query($polaczenie, "SET NAMES 'utf8' COLLATE 'utf8_polish_ci'");
mysqli_select_db($polaczenie, $database);

$zapytanietxt = "SELECT * FROM filmy WHERE id = $zmienna";

$rezultat = mysqli_query($polaczenie, $zapytanietxt);
$ile = mysqli_num_rows($rezultat);

if ($ile>=1)
{
echo<<<END

```

We get this variable in another .php file(kategorie.php), where is creating url link based on MySql query, you can see this mechanism below:

First we create query, that select movies title and id from database that belong to category that we impute to variable $zmienna:

```
$zapytanietxt = "SELECT f.id,f.title FROM filmy as f , filmykategorie as fk WHERE f.id = fk.IdFilm AND fk.IdKategoria = $zmienna";
```

Then we send query to database by using connection created before this steps:

```
$rezultat = mysqli_query($polaczenie, $zapytanietxt);
```

Here we count how many results returns this query and set variable “ile”:

```
$ile = mysqli_num_rows($rezultat);
```

Then by using HTML we create table, that contain all results. Each element of the table is a link to next page – film.php that is creating based on variable $id which is sending inside URL address.

```
if ($ile>=1)
{
echo<<<END
<td width="100" align="center" bgcolor="333333">Tytuły filmowe:</td>

</tr><tr>
END;
}
        for ($i = 1; $i <= $ile; $i++)
        {

                $row = mysqli_fetch_assoc($rezultat);
                $id = $row['id'];
                $title = $row['title'];

                $link = '<a href="kategoria.php?zmienna=$id">{$title}</a>';

                echo<<<END
                <td width="100" align="center"><a
                href="film.php?zmienna=$id">{$title}</a></td>
                </tr><tr>
                END;

        }
```

5. Summary

At the end we have achieved all our assumptions. All point of schedule were developed and implemented. We created good portal about movies which can be improving and maybe one day become perfect.
Right now our website have many functionalities but there are even more that can be implement in the future. We are planning to create system that allows users to communication, for example private messages. There can be implemented rating and commentary system to movies. We can also create movies ranking based on movies rating. Website design and appearance can be improved. One day there may be system that can contain and display movies added by users.

6. Literature

1)http://www.w3schools.com/html/ - November/December 2016
2) http://www.w3schools.com/css/ - November/December 2016
3) http://www.w3schools.com/php/ - November/December 2016
4)Courses developed by Mirosław Zelent about HTML,CSS,JavaScript and PHP.
a)https://www.youtube.com/watch?v=tD0Q5QwoQJI&list=PLOYHgt8dIdox81dbm1JWXQbm2geG1V2uh&index=2 - November/December 2016
b)https://www.youtube.com/watch?v=OcwON22ctYc - November/December 2016
5)https://www.youtube.com/watch?v=fGdd9qNwQdQ&list=PLoYCgNOIyGACTDHuZtn0qoBdpzV9c327V&index=1 – Javascript Tutorials by LearnConde.academy - November/December 2016