UNIVERSITY OF REGINA
Department of Computer Science
CS 215 – Web & Database Programming
Fall 2019

Assignment #2: Movie Database Website – Interface Design & Mock-Up
Due: Wednesday October 2, 2019 by 11:55 PM

The goal of this assignment is for you to build the core interface design and elements for the movie database system that allows users to view movies based on genre and origin, rate movies, and add movies to their watchlist. In the subsequent assignments, you will implement form validation in JavaScript (Assignment #3), design and implement the database in MySQL (Assignment #4), implement the back-end programming in PHP (Assignment #5), and implement AJAX-based updating (Assignment #6). Because the other assignments build upon this one, you should take care to make a complete interface with all of the required elements (listed below). However, you will have a chance (and in some cases, be required to) update your previous work in the subsequent assignments.

The first step in creating the movie database website is to create the interface design and mock-up. There are a number of critical pages that will need to be designed, created, and linked together to illustrate the interface features. For those pages that will ultimately contain user-contributed or movie data, they can be loaded with sample data to illustrate your design. You should start by (1) sketching each page and refine your sketches until you are satisfied with the design, (2) draw storyboards of the critical task to be completed in the application, and then (3) build the pages using HTML5 and CSS.

Eventually, we will be loading all the data from a MySQL database (after assignment #4), but for now, you can add random placeholder data wherever mentioned.

The following pages are to be created:

1. Login Page

   The is the first page the user should see. The user can login, signup, or continue as a guest. Your task is to implement the following features:

   a) Signup form

      This form should collect the information required for creating a new account on your movie database website. At the least, it should collect an email address, username, avatar image/graphic, first name, last name, and a password. You can ask for more user details in the form such as date of birth. The form should ask the user to enter their password twice, to ensure that it is entered properly.

      After the signup, the user should be redirected to the same page (login page). The purpose of this is to let the user login after signup.

   b) Login form

This form should collect the user's email and password to login. We will implement the logic to login in future assignments.

After login, the user should be redirected to the Home page.

c) Continue as guest option

This option should redirect the user to the Home page.

2. Home Page

This page contains all the features of the movie database website. The user will be able to browse movies, rate movies, and add movies to their watchlist. Think of it as a Netflix homepage.

Your task is to create an interface that helps the user in the following tasks:

a) The user should be able to navigate or search movies based on a genre, or an origin. *List of genres and origins are provided in the spreadsheet.*
b) There should be a search bar to help the user search for movies. *Hint: You can use a form for this.*
c) Add a functionality to sort the search results (movies) based on rating, title, and year released. We will implement the logic of sorting in future assignments.
d) A list of movies as a search result. You can create a tile for each movie. Each movie must have a poster, title, rating, and "add to watchlist" button.

   Eventually, we will fetch movies as a search result from the database. Right now, you can add any movies data as placeholder.

e) Add a functionality for the user to navigate to the Watchlist page.
f) Add a functionality to show the user's avatar, username, and logout feature.
g) Add a small login form allowing users to login. Add an option to signup. By clicking on the signup, the user should be redirected to the login page.

3. Movie details page

This page will give more details about a movie. You have to implement the following:

a) Give all the details about a movie. It should contain the movie's title, year released, poster, rating, cast, genre, origin, director, and a Wikipedia link.

   We will eventually fetch all these data from the database. Right now, you can add some placeholder data from the spreadsheet.

b) Add functionality to rate the movie.
c) Add functionality to add the movie to a watchlist. By clicking on this link, the user should be redirected to the Watchlist page.

4. Watchlist page

   A user can create as many watchlist as they wish. This page will contain the list of watchlist, allow the user to create a new watchlist, and delete an existing one.

   You have to implement the following:

   a) A list of all the watchlist that a user has created. For now, you can randomly create few.
   b) An option to create a new watchlist. Make sure you ask the user what the new watchlist should be named.
   c) An option to delete a particular watchlist.

5. Watchlist details page

   This page will show all the movies that belong to a watchlist.

   You have to implement the followings:

   a) A view to show all the movies in a watchlist. For a movie, you should include a poster, movie name/title, and rating.
   b) An option to remove a movie from the watchlist.

All of these files should be linked together, so that it is possible to click through the pages to evaluate the design and implementation. You should give some thought to how these pages would logically connect with one another.

Each page on which a user is logged in (except the login page), you should include information about the logged in user (handle and avatar), and provide a link for the user to log out. The logic for logging out will be handled in a future assignment.

You must ensure that each of these pages makes appropriate use of HTML5 (following the syntax rules of XTHML) and CSS, and construct them such that there is a proper separation of the specification of the content from the specification of the presentation rules. Your submission should include five sketches (one per page), a set of storyboards, five HTML5 pages, and a single CSS file.

Grading Scheme

This assignment will be graded out of 10 marks, based on the following criteria:

2 marks:	Sketches of the interface pages, following appropriate design principles
2 marks:	Storyboards of the key user tasks
4 marks:	HTML5 that follows XHTML syntax rules for all five pages
2 marks:	A single CSS file to specify the presentation details for the web site

If there are any missing pages, your grade will be scaled down according to the number of missing pages. For example, if you submit only 4 of the 5 required pages, your grade will be 4/5 of what is assigned based on this marking scheme.

Submissions

All of the files for this assignment must be posted to your personal website on Hercules, as well as uploaded to UR Courses (a single zip file that includes scans of the sketches and storyboards, as well as your website directory). A simple submission log file should be provided that includes your name, student number, class number, the URL of where you have posted the files, and a listing of the files you have submitted along with a short explanation of the purpose of each file.

You must also provide a link to your web application (posted on Hercules) as part of the submission process, which will enable the TA to easily access a live version.

Failure to provide these support documents will result in delays in the grading of your assignment and possibly a deduction in your grade.

Late submissions will not be accepted. If there are exceptional circumstances that kept you from submitting your assignment on-time, you should consult with your instructor as soon as you are able to do so. See the syllabus for more details on the late policy for this class.