

CMPS 350 – Web Development Fundamentals

Midterm Practical Exam – Spring 2024

Student Name	
Student Id	
Grade	

Instructions » Read carefully

1. The exam duration is **120 minutes**. So, read the exam questions carefully and plan your time accordingly.
2. As you make progress, push your code to your GitHub repository under Midterm to avoid unpleasant surprises, such as a computer's hang: or the program accidentally shutdown.

Warning

3. In case of plagiarism, both parties will receive 0 and will be reported.
4. The use AI tools such as ChatGPT , Code Whisperer , Gemini etc .. is strictly forbidden.

Objective: This examination is designed to assess your proficiency in frontend web development through the creation of an online movie database. Your task is to construct a platform where users can view movie movies along with their ratings, watch movie trailers, and read detailed descriptions of the films. A visual representation of the user interface (UI) design is included below, and a live demonstration of the application will be provided at the beginning of the exam.

Implementation Tasks

1. [35 pts] Get Movies data and display as cards

Upon loading the page, retrieve the list of movies from <https://bit.ly/3HPIkB1> and display them on the homepage in a card format as illustrated in figure 1. Each card should showcase the movie's Title, Synopsis, Year, Duration, Rating, Score, and Image (Large version), along with "Delete" and "Show More" buttons. Also the list of movies should be responsive [Use grid with the RAM technique].



Figure 1 Home Page

2. [20 pts] Caching Movie Data in localStorage

Improve your application performance by caching the **movie data** in **localStorage**. Meaning, when you first get the movie data save it to **localStorage**. Subsequent visits should read the data from **localStorage**. Only fetch data from the remote API if the movie data is not available in the **localStorage**.

3. [25] Display Movie Details and trailer

When the user clicks on the "Show More" button found in the movie's card, pass the movie's ID to the **displayMovieDetails** function to retrieve its comprehensive details from **localStorage**. Display the movie's details such as **Title, Rating, Image, Synopsis, and Trailer** from youtube. Replace the list with the detailed card as shown below. Include a **"Back"** button that reverts the view from the detailed card back to the original list.



Figure 2 Details Screen

4. [20] Delete Movie from the Movie List and localStorage

If the user clicks on the Delete button, then delete the Movie from your **localStorage** and from the displayed list.