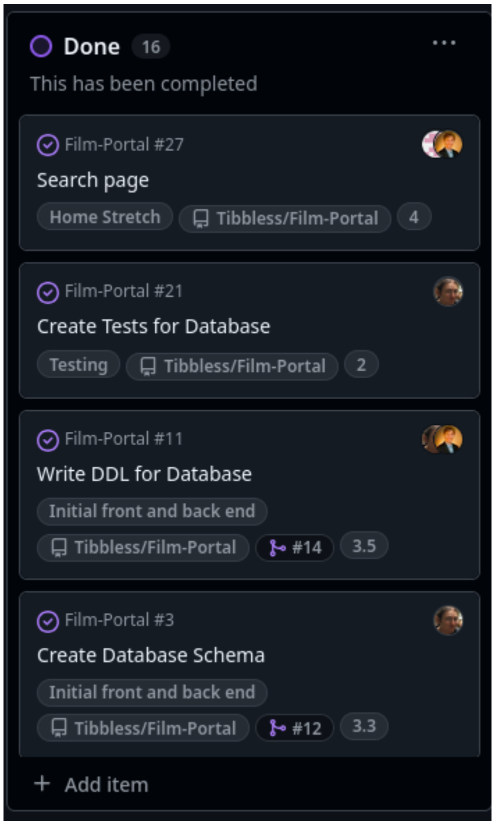
Film Portal

Oscar Bender-Stone: oscar-benderstone
Calvin Schaller: casc1204
Ethan Skamarock: Phatdeluxe
Nolan Tibbles: Tibbless

Description: Film Portal is an engaging social media style website where users can create and read reviews for movies. Users can search for movies on demand, create a profile with a custom image, and find reviews by user ID, movie title, and description. To access these features, users register accounts and login to the site. Having an account allows them to have persistent data through multiple sessions, which is handled by a backend to the server's database. Users can then see other posts on the home page and comment on them. Additionally, users can create posts. To provide seamless movie search, we call the TMDB API, which allows users to find movies by search title. Thir post is then stored in the database and displayed on the homepage. Users can also use the search function to find specific posts pertaining to a movie title, a specific user, or a posts body. These results of their query are then pulled up on a fine grained display page. Finally, via the profile page, a user can change their information (profile picture, password, email address, or username), or delete their account.

Project Tracker: https://github.com/users/Tibbless/projects/1/views/1



Done 16

This has been completed

Film-Portal #27
Search page
Home Stretch · Tibbless/Film-Portal · 4

Film-Portal #21
Create Tests for Database
Testing · Tibbless/Film-Portal · 2

Film-Portal #11
Write DDL for Database
Initial front and back end
Tibbless/Film-Portal · #14 · 3.5

Film-Portal #3
Create Database Schema
Initial front and back end
Tibbless/Film-Portal · #12 · 3.3

+ Add item

Video (Please go to Raw mode or download the video file, and then view the mp4 file): https://github.com/Tibbless/Film-Portal/blob/main/MilestoneSubmissions/Film%20Portal%20Video%20Recording.mp4
Repository: https://github.com/Tibbless/Film-Portal.

Contributions

- **Oscar** worked on incorporating external API calls to TMDB (The Movie DB) using axios and cached results into our server database to minimize calls. Additionally, he implemented the create post feature, consisting of the create-post HTML page with Handlebars and database integration with pg-promise. He also fixed issues within the database and collaborated with Calvin and Nolan on the search page.
- **Calvin** created the implementation for the Home Page and Profile page. This includes finding ways to implement API calls into results that the user can see and use. He also helped debug the search page so that it had a working implementation alongside Nolan and Oscar. He also worked with Ethan to create friendly user interfaces using the database.
- **Ethan** created the database, designing the schema and writing the DDL and DML. He created the settings page, allowing the user to change their username, password, profile picture, and delete their account. He also set up render hosting.
- **Nolan** worked on users being put into the database and front end systems involving that process such as the login and registration pages. Also worked to implement the search post functionality which queried our database for posts made by other users.

Use Case Diagram:
https://github.com/Tibbless/Film-Portal/blob/main/MilestoneSubmissions/lab-9-materials/Lab9Case.png

Wireframes:
https://github.com/Tibbless/Film-Portal/blob/main/MilestoneSubmissions/lab-9-materials/Inital-wireframe-for-Film-Portal.pdf

Test results:
Test 1:

1. Registration

    a. I was able to easily navigate there.
    b. A different look for the button might help differentiate it from the login button.

2. Login
    a. Redirected as expected.
    b. Signed in with full functionality available.
3. Review creation
    a. Found the create post option on the nav bar after a short scan of the home page.
    b. Though the title section was for the movie title, the user could change it after seeing the movie section.

Overall, few problems other than semantic issues.

Test 2:

1. Registration
    a. I was able to easily navigate there.
2. Login
    a. Were redirected as expected.
    b. Signed in no problem.
3. Review creation
    a. Quickly found the create post navbar option.
    b. Also made the mistake of putting the movie title in the review title field.

Better than the first test, but still small semantic issues.

Deployment (on Render): https://film-portal.onrender.com/login.