

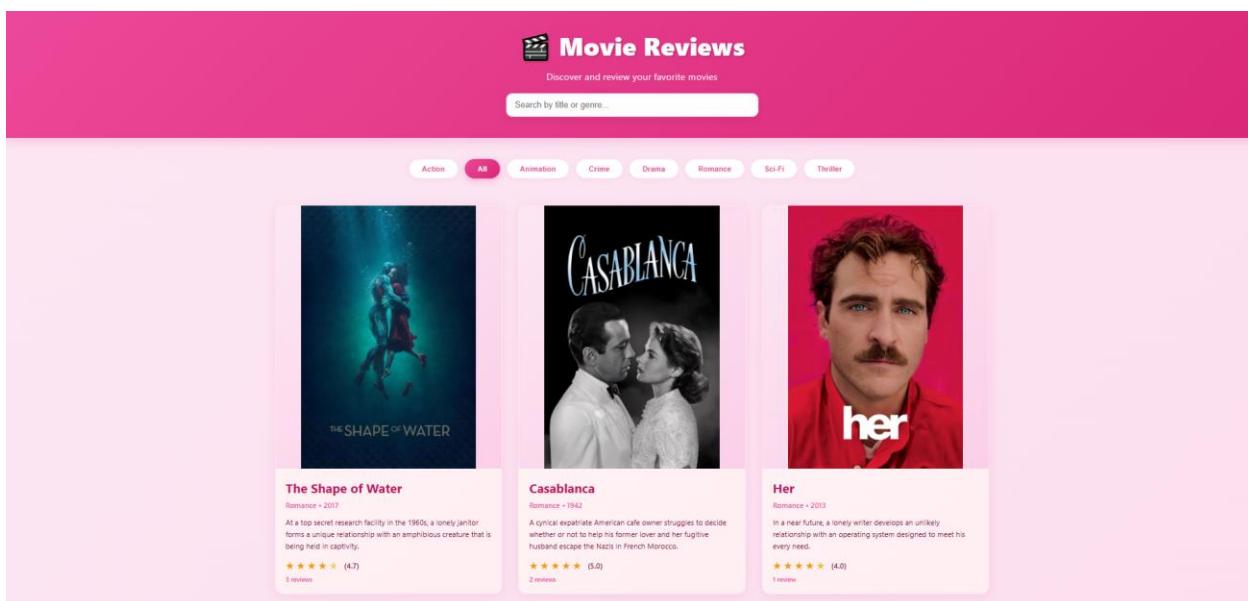
Activity 6: Movie Review Application

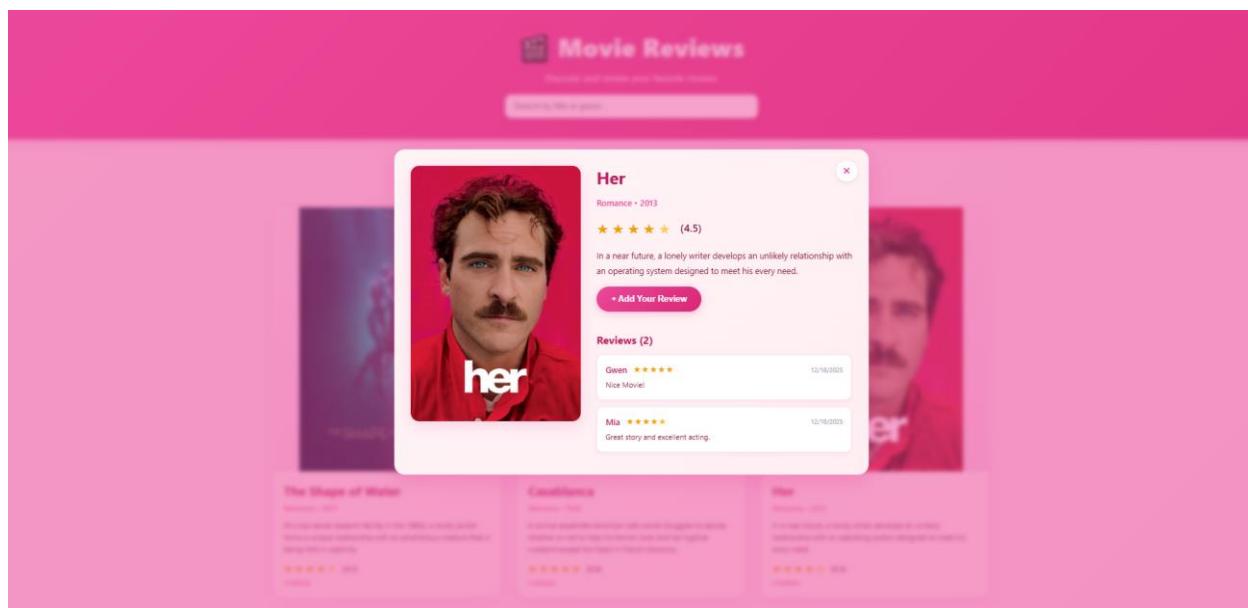
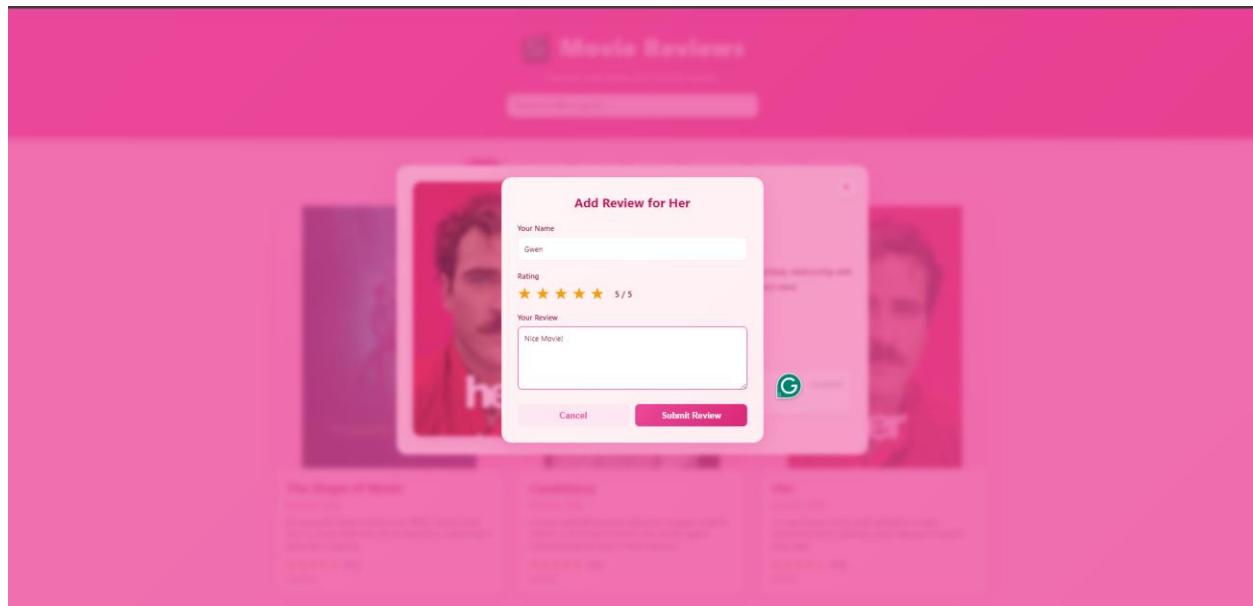
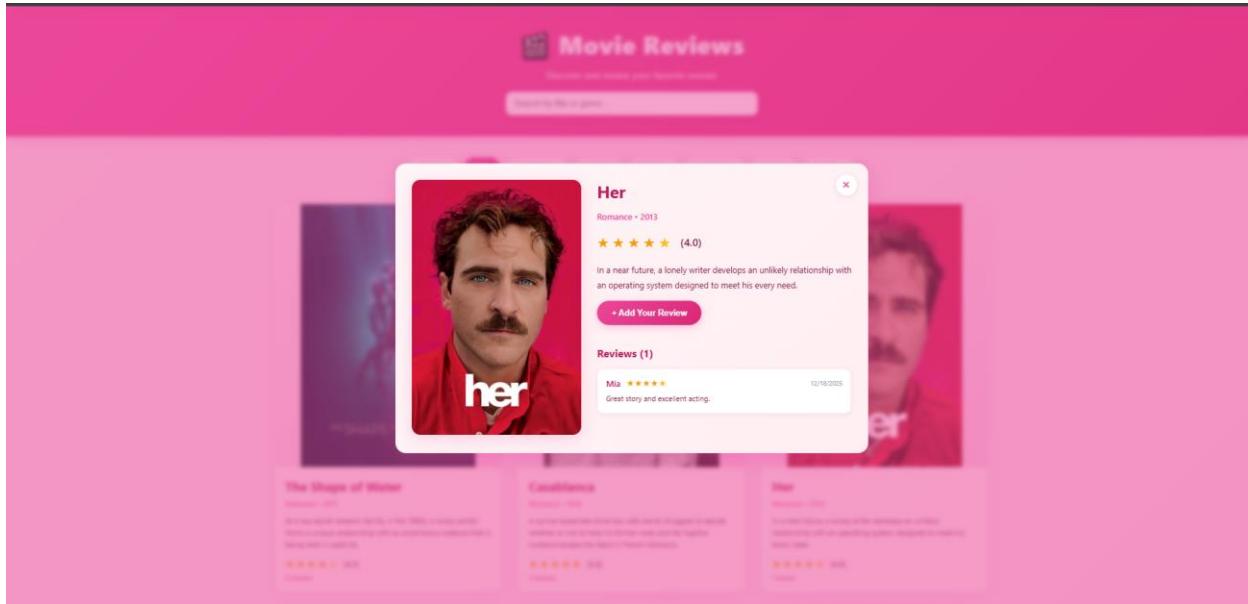
This is a full-stack web application that allows users to manage and review movies. The system provides a user-friendly interface where you can browse movies, view their details, and add reviews with ratings. The application automatically calculates and displays the average rating for each movie based on all submitted reviews. Users can search and filter movies by genre, and the system maintains a complete database of movies and their reviews.

The application consists of two main parts: a backend API built with NestJS that handles all the data operations, and a frontend interface built with React that provides an attractive pink-themed user interface. The backend uses SQLite database which requires no external setup, making it easy to run the application on any computer.

Screenshots

User Interface Screenshot





How to Run the Project

Before you start, make sure you have Node.js installed on your computer. You can download it from the official Node.js website if you don't have it already. You'll also need npm, which comes automatically with Node.js.

Option 1: Running Both Servers Together (Recommended)

This is the easiest way to start the application. It will run both the backend and frontend servers at the same time.

1. Open a terminal or command prompt and navigate to the Activity6 folder.
2. Install the required packages by running: `npm install`
3. Start both servers by running: `npm start`
4. Wait for both servers to start. You'll see messages indicating that the backend is running on port 3001 and the frontend is running on port 3000.
5. Your web browser should automatically open and show the Movie Review application. If it doesn't, simply go to <http://localhost:3000> in your browser.

Option 2: Running Servers Separately

If you prefer to run the backend and frontend in separate terminal windows, follow these steps:

Starting the Backend:

1. Open a terminal and navigate to the backend folder inside Activity6.
2. Install dependencies by running: `npm install`
3. Start the backend server by running: `npm run start:dev`
4. Wait until you see a message saying the application is running on <http://localhost:3001>
5. You can verify the backend is working by visiting <http://localhost:3001/api> in your browser to see the API documentation

Starting the Frontend:

1. Open a new terminal window and navigate to the frontend folder inside Activity6.
2. Install dependencies by running: `npm install`
3. Start the frontend server by running: `npm start`
4. The application will automatically open in your browser at <http://localhost:3000>

Using the Application

Once both servers are running, you can start using the application:

1. The main page will show all available movies in a grid layout. If there are no movies yet, you'll need to add some first.
2. To add movies, visit the API documentation page at <http://localhost:3001/api> and use the POST /movies endpoint, or you can use any API testing tool like Postman.
3. Click on any movie card to view its details and see existing reviews.
4. To add a review, click the "Add Review" button, fill in your name, select a rating from 1 to 5 stars, write your comment, and submit.
5. You can search for movies using the search bar at the top, or filter movies by genre using the genre tabs.

Adding Sample Movies

To add movies to the system, you have two options:

1. Use the Swagger UI: Go to <http://localhost:3001/api>, find the POST /movies endpoint, click "Try it out", fill in the movie details, and click "Execute".
2. Use an API client: Make a POST request to <http://localhost:3001/movies> with a JSON body containing the movie information.

Here's an example of what a movie entry looks like:

```
{  
  "title": "The Matrix",  
  "description": "A computer hacker learns about the true nature of reality",  
  "genre": "Sci-Fi",  
  "imageUrl": "https://example.com/matrix.jpg",  
  "releaseYear": 1999  
}
```

Important Notes

- Always start the backend server before starting the frontend server if you're running them separately.
- The database file is created automatically when you first run the backend, so you don't need to set up anything manually.
- If you see connection errors, make sure both servers are running and check that ports 3000 and 3001 are not being used by other applications.

- The application uses SQLite database, which stores all data in a file called movie_reviews.db in the backend folder.