

Assignment Implementation Summary

Features Implemented:

1. **Home Page:**
 - Includes a search bar and a button to search for movies.
 - Displays a section for popular movies.
2. **Search Result Listing Page:**
 - Shows the results of movie searches with pagination.
3. **Movie Detail Page:**
 - Provides detailed information about each selected movie.
4. **Watchlist Management:**
 - Allows adding, removing, and listing movies in the watchlist using SQLite and Prisma as the ORM.

Bonus Features:

1. **Pagination:**
 - Implemented pagination for both watchlist and search results.
 2. **Popular Movies Section:**
 - Added a section on the home page for showing popular movies.
-

Architectural Approach

- **App Router with Next.js:**
 - Utilized the Next.js app router to manage all the pages within the app folder.
 - **API Operations:**
 - Complex logic for API operations, specifically for the watchlist, is handled within the `api` folder.
 - **Component Structure:**
 - Created a `components` folder inside the `src` directory for reusable non-UI components (e.g., pagination, navbar).
 - **Utility Functions:**
 - Added a `utils` folder in the `src` directory to store utility functions that can be reused across the application.
 - **Code Quality:**
 - Integrated a linter to maintain code quality and consistency.
-

Working Approach

- **Design:**
 - Mainly used Tailwind CSS classes and NextUI for the design.
- **State Management and Side Effects:**
 - Used `useEffect` for handling side effects.
 - Used `useState` for managing data that affects the UI.