

## PROJECT PLAN – XList

### 1. Project Overview

The idea of this project is to create a simple Movie & TV Watchlist website that helps users keep track of what they want to watch. The site allows users to add movies or shows, update their status, edit information, and delete entries when they are no longer needed. It is designed to be clean, minimal, and easy to use.

The purpose of this project is to practice development using Node.js, Express, MongoDB, Mongoose, and EJS. It also shows good practices such as using route files, controllers, templates, and environment variables for security.

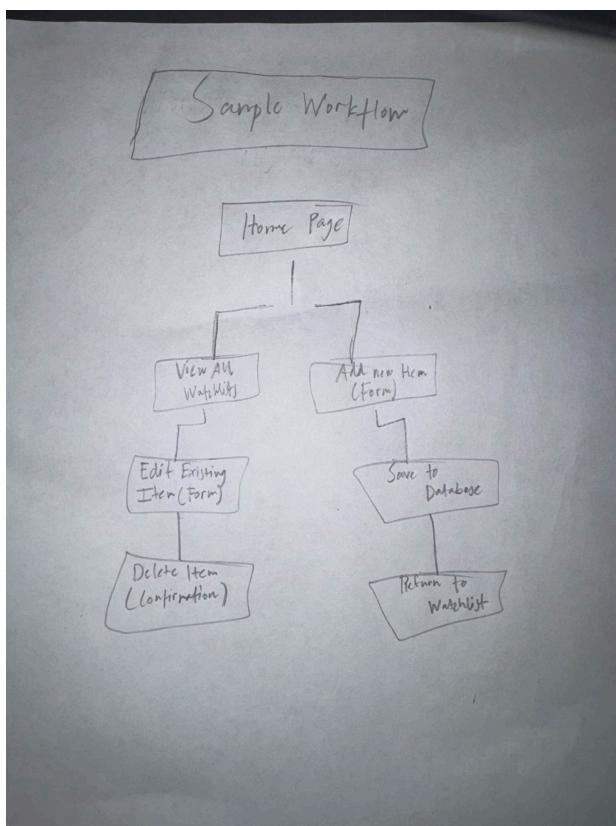
### 2. Project Goals

- Provide a clean interface for users to manage a personal watchlist.
- Allow full CRUD functionality:
  - Create a new item
  - Read (view the list)
  - Update item details
  - Delete items
- Use Bootstrap and light custom CSS to keep the interface simple and visually appealing.
- Use MongoDB Atlas to store the watchlist items.
- Follow secure coding practices (e.g., using `.env` for credentials).

### 3. Key Features

- Home Page: A splash page with buttons leading to the watchlist and “Add Item” page.
- Watchlist Table: Shows all stored items from the MongoDB database.
- Add Item Form: Allows users to add a new movie or TV show.
- Edit Form: Enables updates to an existing item.
- Delete Confirmation: Asks the user to confirm before deleting.
- Navigation Bar: Clean and simple; helps users move between pages.
- MongoDB Integration: Stores movie/TV items using a Mongoose model.

### 4. Simple Workflow Diagram



## 5. Technologies Used

- Node.js – server-side JavaScript runtime
- Express.js – web framework
- MongoDB Atlas – cloud database
- Mongoose – schema/model for MongoDB
- EJS – templating engine
- Bootstrap – basic styles & layout
- CSS – small amount of custom styling
- .env – secure environment variables

## 6. Why This Project?

The Movie + TV Watchlist is simple but effective for learning:

- It covers all CRUD operations clearly.
- It avoids unnecessary complexity.
- It uses modern tools similar to real-world development.
- It provides a clean and minimal UI that anyone can navigate.

## 7. Screens and Flow (Short Description)

### Home Page

- Basic splash page with two buttons:
  - "View Watchlist"
  - "Add Item"

## Watchlist Page

- Table showing all movies & TV shows.
- Buttons for Edit and Delete next to each item.

## Add Item Page

- Simple form to enter:
  - Title
  - Type (Movie or TV)
  - Status (Planned, Watching, Finished)
  - Notes (optional)

## Edit Page

- Same form as Add, but pre-filled.

## Delete Page

- “Are you sure?” confirmation message.

## 8. Project Conclusion

This project successfully demonstrates a full CRUD web application using Express, MongoDB, and EJS. It combines a clean UI with simple backend logic and follows secure coding habits. The features are minimal but complete, making it easy to expand in the future if needed.