Building a Movie Management Web Application with JSON Server Integration

**Objective**: In this workshop, you will embark on a hands-on journey to create a comprehensive movie management web application. The application will allow users to effortlessly organize, explore, and manipulate a collection of movies. The movie data will be stored and retrieved using a JSON server, adding a practical layer of data persistence to the application.

**Prerequisites**:

* Basic knowledge of HTML, CSS, and JavaScript.
* Familiarity with JSON data format.

**Task Steps:**

* **Setting up the JSON Server:**
  + Begin by installing JSON Server globally using npm: npm install -g json-server.
  + Create a dedicated JSON file, such as movies.json, to serve as the database for storing movie information.
  + Populate the movies.json file with an initial set of movies, encompassing crucial details such as title, genre, director, release year, and any other relevant information.
  + Launch the JSON Server by executing the command: json-server --watch movies.json.
* **Creating the Web Application**:
  + Establish the foundation for the movie management web application by crafting a well-structured HTML layout.
  + Design an intuitive and visually appealing user interface that will display a dynamic list of movies along with their pertinent details.
  + Construct an interactive form enabling users to add new movies. Implement appropriate input fields for each movie property, such as title, genre, director, and release year.
  + Utilize JavaScript's fetch API to retrieve movie data from the JSON server and seamlessly present it within the web application.
  + Implement event handlers to capture form submissions, transmitting the movie data to the JSON server utilizing the POST method.
  + Enrich the user experience by enabling movie updates and deletions. Incorporate suitable user interface elements to trigger actions such as editing movie details (utilizing PUT or PATCH requests) or removing movies from the collection (DELETE requests).
* **Integrating the Web Application with JSON Server:**
  + Harness the power of the fetch API to establish a connection with the JSON server and retrieve movie data, ensuring the web application remains in sync with the server's database.
  + Develop a seamless workflow that allows users to effortlessly submit new movies to the JSON server using the POST method, dynamically updating the user interface to reflect the changes.
  + Enhance the web application's interactivity by enabling movie updates. Implement logic to send PUT or PATCH requests to the JSON server, facilitating modifications to movie details, and promptly reflecting those changes within the user interface.
  + Empower users to remove movies from the collection by implementing a mechanism that sends DELETE requests to the JSON server, ensuring consistent data synchronization between the web application and the server.
* **Testing and Refinement:**
  + Thoroughly test the application's functionality, ensuring the seamless addition, modification, and removal of movies reflect accurately within the JSON server.
  + Employ client-side data validation techniques to validate user input, preventing the submission of invalid or incomplete data.
  + Elevate the user interface by implementing additional features and functionalities according to your creativity and skill level. Explore options such as search functionality, genre-based filtering, or sorting movies by release year.

**Additional Challenges (Optional):**

* Implement an authentication system to secure user access to the web application and JSON server.
* Incorporate server-side data validation to enforce specific rules and constraints when creating or modifying movies.
* Integrate an external movie API to enrich the movie details with additional information, such as plot summaries or cast members.