Proposal: Media Collection Management System

Emily Weed
Feb 2, 2025

Content

Problem Statement

As media consumption continues to rise, managing a collection of books, movies, music, and other forms of media has become increasingly complex. This is especially true for individuals or establishments (like libraries, rental stores, or content distributors) that need an efficient system to catalog, track, and retrieve media items. Currently, there is a lack of seamless, user-friendly systems that can organize such diverse collections, handle loans, and ensure accurate tracking. The problem this project aims to solve is providing a comprehensive media collection management system that centralizes ownership information, simplifies media retrieval, and supports tracking of media loans.

System Description

The Media Collection Management System (MCMS) will provide a platform for managing collections of media items such as books, movies, music, and other formats. The system will allow for easy categorization, tracking, and retrieval of media assets.

Key features will include:

- A user-friendly interface for both owners and customers (if applicable).
- The ability to track ownership, including detailed metadata such as author, director, actors, genre, format, and more.
- Loan management functionality to track items borrowed or rented.
- Search functionality that allows users to filter media by categories, titles, genres, and other attributes.
- User accounts (for both administrators and customers) to track loans, favorites, and wish list items.
- Integration with external systems for media metadata (e.g., movie databases, book APIs).

Objectives of the System

The primary objectives of the Media Collection Management System are:

- To enable users to track their owned media, including books, movies, and other formats.
- To create an organized and searchable system to manage and retrieve media items based on various criteria such as genre, title, author/director, format, and more.
- To implement a loan tracking system for establishments like libraries or rental stores, allowing the system to track who borrowed what, when, and for how long.
- To support metadata integration for automatic media data population (e.g., movie descriptions, book synopsis).
- To provide users with a wishlist or recommendations feature to help manage future acquisitions.

System Requirements

Hardware Requirements

- Server (for hosting the application and database)
- User devices (desktops, tablets, or mobile phones for accessing the system)

Software Requirements

- **Front-end**: React.js or Angular (for building the interactive user interface)
- **Back-end**: Node.js with Express.js (for handling server-side requests and business logic)
- **Database**: MySQL or MongoDB (for storing media metadata, loan information, user accounts, and other system data)
- **External APIs**: Integration with book/movie metadata APIs (e.g., Open Library, OMDB, The Movie Database)

Typical Customers

- **Owners/Administrators**:
    - Add new media items with full metadata (title, author/director, genre, year, etc.)
    - Edit or delete media entries in the collection
    - Track the media available for loan and the status of loans
    - Manage user accounts, permissions, and media access levels
    - Integrate with external media databases for easy metadata population
- **Customers/Users** (for establishments like libraries or rental stores):
    - Browse, search, and filter through available media items
    - Borrow and return media items, with the system tracking loan dates and return deadlines
    - Maintain a personal collection of media (owned or borrowed) and manage a wishlist
    - Receive notifications about overdue media and new arrivals

The system will be ideal for:

- **Libraries**: For tracking physical and digital media like books, audiobooks, DVDs, and other media available for loan.
- **Rental Stores**: Video rental stores, game rental services, and similar businesses that need to manage inventory and customer loans.
- **Media Collectors**: Individuals who want to keep track of their personal media collections, whether physical or digital.
- **Educational Institutions**: Schools, universities, or colleges that need a system for managing educational resources like textbooks, eBooks, or educational films.

Project Planning

Software Requirements

- **Front-end**: React.js or Angular (for dynamic, responsive UI)
- **Back-end**: Node.js with Express.js (for server-side handling of requests)
- **Database**: MySQL (relational database for structured media data) or MongoDB (document-based storage)
- **External APIs**: Open Library API (for books), OMDB or The Movie Database API (for movies)

Hardware Requirements

- **Server**: Cloud-based or on-premise server for hosting the application
- **User Devices**: Desktop and mobile devices for accessing the system

Network Requirements

- Reliable internet connection for accessing the system remotely
- Secure HTTPS protocol for safe data transmission

Development Approach

The development of the system will follow a modular, full-stack approach, utilizing the following technologies:

- **Frontend**:
  - React.js or Angular (to create an interactive, dynamic interface)
- **Backend**:
  - Node.js with Express.js (for creating RESTful APIs and handling backend logic)
- **Database**:
  - MySQL or MongoDB (MySQL for structured media records or MongoDB for flexibility with various media types)
- **External Metadata APIs**:
  - Open Library (for book metadata)
  - OMDB (for movie metadata)

Development Plan

Week 1-2

- Design system architecture and define database schema (media types, loan tracking, user data).
- Set up the development environment and start building the front-end with React.js.
- Connect the front-end with the back-end API.

Week 3-4

- Implement media cataloging functionality (adding, editing, deleting items).
- Develop user account management (authentication, roles, permissions).
- Begin building the media search functionality.

Week 5-6

- Implement a loan tracking system (borrow, return, overdue notifications).
- Integrate with external APIs (Open Library, OMDB).
- Test search, loan tracking, and metadata population features.

Week 7-8

- Build wishlist functionality and user media collection management.
- Develop recommendations for media based on user preferences.
- Test all user-facing features and improve UI/UX.

Week 9-10

- Begin testing the system with real users, including librarians, rental store employees, and personal media collectors.
- Collect feedback and refine features based on input.
- Address any bugs or performance issues.

Week 11-12

- Conduct final testing, especially on media metadata integration and loan tracking.
- Finalize documentation and prepare user manuals.
- Prepare a demo for the final presentation.

With this Media Collection Management System, both individuals and establishments will have an efficient way to track and manage their media collections, ensuring easy retrieval, organized storage, and accurate tracking of loans. The inclusion of search, metadata integration, and loan tracking makes the system versatile and beneficial for various types of users.