CS6.401 Software Engineering

Spring 2024

Project - 2

Welcome to the second phase of the course project! In the first phase, we delved into the Books repository, and understood its code structure. In this second phase, we will continue our exploration of Books by expanding its capabilities and functionality using software engineering principles. This phase will focus on enhancing Books' features to make it more useful and user-friendly.

GitHub Classroom

You will be continuing to work on the same repository that you used in the first phase of the project.

Specification

You need to extend Books to support each feature described below using appropriate design patterns. You may need to refactor existing code to support this. Use at least 2-3 different design patterns, excluding trivial patterns such as singleton pattern. You will be graded on efficient and precise use of design patterns.
Note that the changes to the frontend do not need to be extensive and you will not be graded on this.

Feature 1: Better user management (20)

Our app has a user management system already in place, which you have already looked at in detail. The current flow of creating a user involves logging into an admin account and creating a new user manually. However, this is neither intuitive nor convenient, and thus demands an improvement.

Objective: Enhance user experience and convenience by enabling self user registration directly from the login page.

Detailed Description:

- Allow the user to enter his/her username, email and password in the registration page.

- The email should be checked for uniqueness (i.e. the same email shouldn't have been registered before)

- The user should be asked to enter the password twice for confirmation.

- Apply checks such as valid email id, at least 8 digit password etc.

Feature 2: Common Library (40)

How about a common library system accessible to all users, allowing them to contribute books, rate existing books, and explore the library's collection!

Objective: Implement a common library accessible to all users for contributing and exploring books. This feature allows users to add books in a common library and rate the books.

Detailed Description:

- **Book Details:** Each Book in the library should have the following information:

    - Title

    - Author(s)

    - Genre(s) (supporting multiple genres per book)

    - Rating (numerical value, calculated and averaged from user ratings)

- ○ Thumbnail image URL (Optional)

- ● **User Management:** Users can register and interact with the system, including:

  - ○ Adding books to the library with the required information

  - ○ Rating existing books on a defined scale (1-10)

  - ○ Viewing all books in the library

- ● **Book Ranking:** A dynamic list displays the top 10 books based on the following two criterias (will be selected by user).

  - ○ Average Rating

  - ○ Number of Ratings

- ● **Filtering:** Users can filter displayed books by:

  - ○ Author(s) (multi-select supported)

  - ○ Genre(s) (multi-select supported)

  - ○ Ratings (Single select, options like >6, >7, >8, >9 etc.)

Feature 3: Online Integration (40)

Ever wished your library could groove to your favorite tunes or narrate your go-to podcast? Say hello to our latest feature: integrating Spotify and iTunes right into our platform!

Objective: The primary objective of this task is to enhance the user experience by integrating the selection of Spotify or iTunes as service providers for accessing audiobooks and podcasts. This integration will empower users to choose their preferred service provider and content type seamlessly within the platform, thereby enriching their overall experience and expanding the platform's functionality.

Detailed Description:

- **Selection of service providers**: Implement functionality to allow users to select either Spotify or iTunes as the service provider for accessing audiobooks and podcasts.

- **Content type selection**: Enable users to choose between audiobooks and podcasts within the selected service provider.

- **Searching and results**: Allow searching using a simple string which the user types in a search bar and then the results get displayed to the user.

- **Saving favorites:** Allow users to mark any audiobook or podcast as favorite which is then saved in the database for that user and can be accessed by the user in a separate favorites section.

Important Note: These APIs do not allow you to actually play the audiobook/podcast. Therefore, it is sufficient to show a simple string representation of the results to the user.

Documentation Links for APIs to be accessed are given below:
- [iTunes Search API](#)
- [Spotify API](#)

For iTunes you should go through given documentation and see how you can access podcasts and audiobooks. And for Spotify, to access podcasts go through the episodes section in the given link and for audiobooks look into the audiobooks section.

Important points to be considered:

Please ensure that each feature implementation prioritizes the effective utilization of design patterns and Object-Oriented Programming (OOP) principles.
Evaluation will heavily consider the degree to which these principles are applied, rather than simply completing the implementation in any manner.

- The system should adhere to Object-Oriented Programming (OOP) principles.
- Utilize appropriate design patterns to ensure modularity, extensibility, and maintainability of the system.
- For feature 2, implement filtering and book ranking functionalities in the backend(in Java) to avoid excessive front-end processing.
- Ensure proper error handling and validation of user inputs.
- Create clear and concise documentation explaining the system architecture and the design patterns used.
- The frontend design can be minimal as the main objective of this project is to help you learn design patterns by implementing it in a real application.

Bonus Feature: Public & Private Bookshelves

Objective: The objective of this task is to introduce the capability for users to adjust the privacy settings of their bookshelves. This enhancement empowers users to mark their bookshelves as either private or public.

Detailed Description:

- Implement functionality within the app to enable users to mark their bookshelves as either private or public.
- Define "private" bookshelves as accessible only to the respective user who created them, ensuring confidentiality and privacy of personal reading collections.
- Designate "public" bookshelves as accessible to all users of the common library system, enabling others to view the books within these shelves in a read-only mode.

Submission Guidelines

- All the above code needs to be pushed in the main branch of the same github repository alloted to you.
- You need to submit a report containing the details of implementation, including the design patterns used and any changes made to existing code. The report should be present in the docs directory, titled project2_<team_number>.pdf. Accurately report the contribution of each team member at the end of this report.
- In case you attempt the bonus feature, you need to submit a separate report titled project2_bonus_<team_number>.pdf.

Soft deadline: 18th March, 2024
Hard deadline: 25th March, 2024