

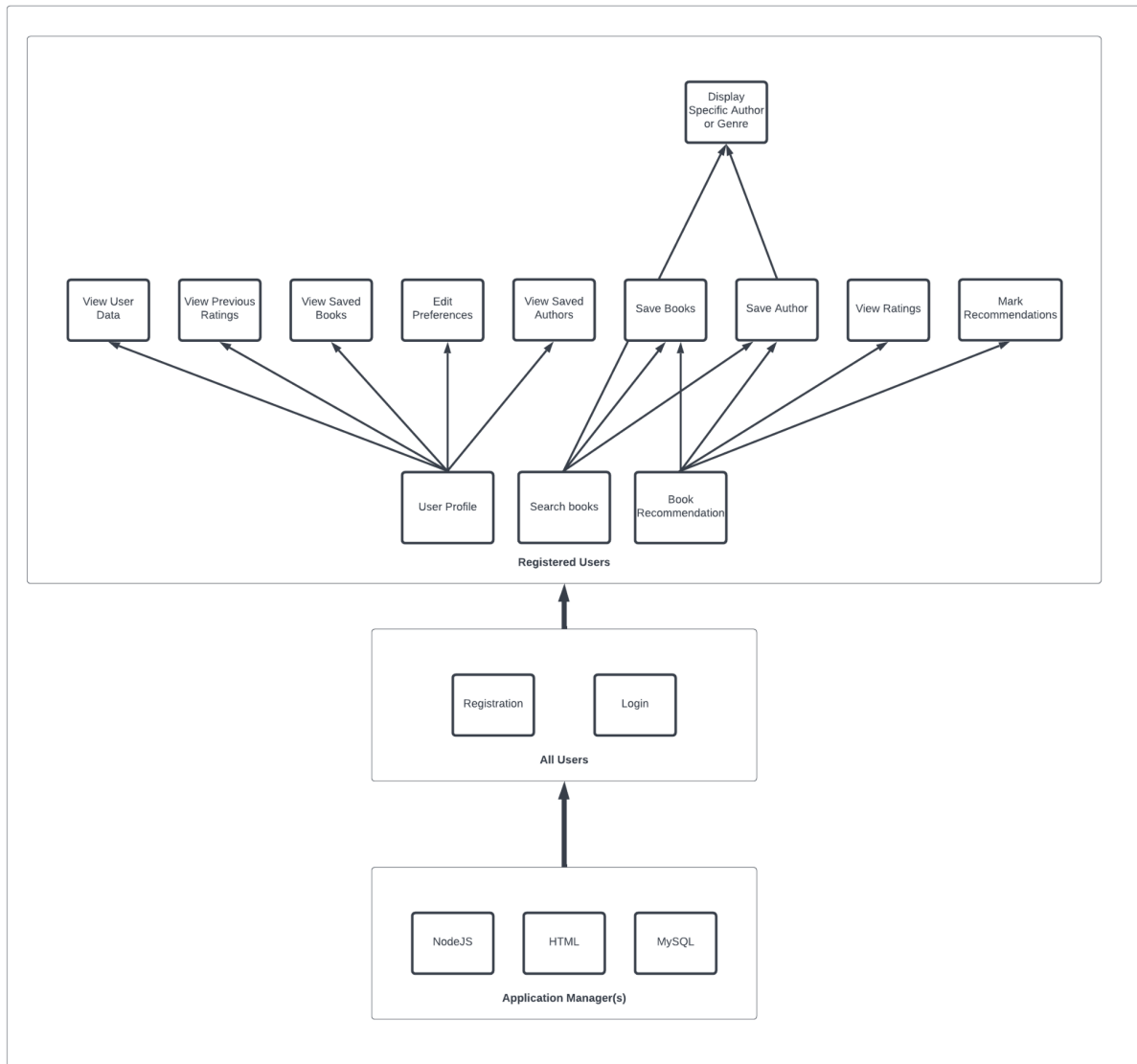
**Team Number:** 011 - 6

**Team Name:** JAAAC

**Revised List of Features (In Priority Order):**

- Login (required)
  - Common login for registered users and admins to access all features
- Account Setup (optional)
  - Right after the user creates a new account, the website will ask them to choose preferred genres and mark other preferences in order to give the algorithm something to work with (with no information, the algorithm will simply recommend top books according to others' reviews)
- Recommendations (required)
  - The recommendation page will calculate how relevant each book is to the user (using the search algorithm), then show the list in order of highest to lowest relevance.
- Rating System (required)
  - Allow users to rate books in order to receive tailored results
- Search algorithm (required)
  - Allow users to find specific recommendations based on criteria such as previous rating, author, genre, and date of publication. The algorithm will examine each of these features, then score books according to which are most similar to ones the user already likes. This will use books from the Kaggle database or an API.
- Saving System (optional)
  - Allow users to save books and authors for later. These books/authors will be marked in the database, and the user will be able to access this list.
- User Profile (optional)
  - Allows registered users and authors to see their saved books/authors, rated books/authors, and preferred genres in the same place

**Architectural Diagram:**



## **Front End Design:**

### **Wireframes:**

Login

Registration

Home

Recommended books

Saved Books/Authors

Search Page

## **Login Page**

A login form for 'NextPage'. It features the title 'NextPage' and subtitle 'User Login'. Below these are two input fields: 'Username' and 'Password'. A yellow 'Login' button is positioned below the password field. At the bottom, there is a link that says 'Don't have an account? Sign up here!'.

## Registration

A registration form on a grid background. At the top is a box with an 'X' representing a logo. Below it are three input fields labeled 'Username:', 'Password:', and 'Confirm Password:'. Under these is a section titled 'Genre Preferences' with five checkboxes: 'Horror', 'Mystery', 'Fantasy', 'Sci Fi', and 'Romance'. A 'Submit' button is at the bottom.

The box with the X in it represents our logo.

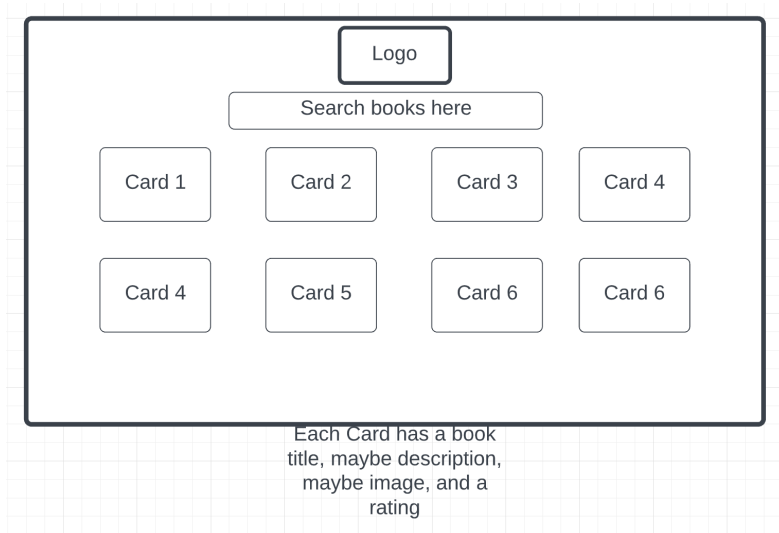
## Home Page

A hand-drawn sketch of a home page. At the top is a search bar with a magnifying glass icon and a user profile icon with a gear icon. Below this is a section titled 'Featured Titles' which contains four book covers. The covers are labeled 'The Great Gatsby', 'The Night Circus', 'Snow Crash', and 'The Help'.

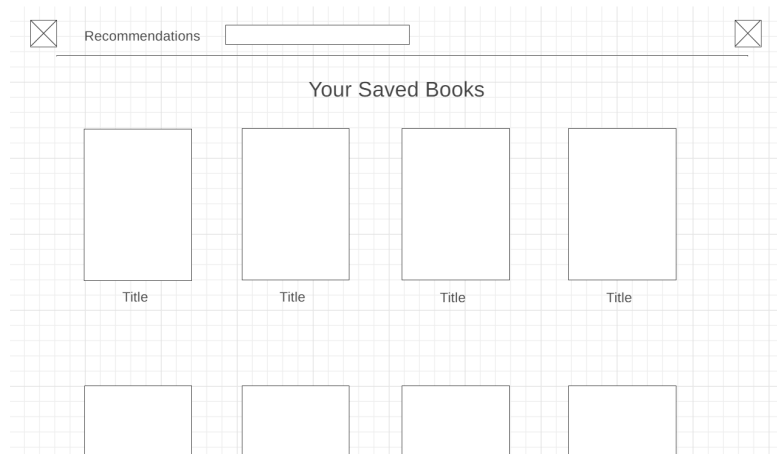
## Recommendations



## Search

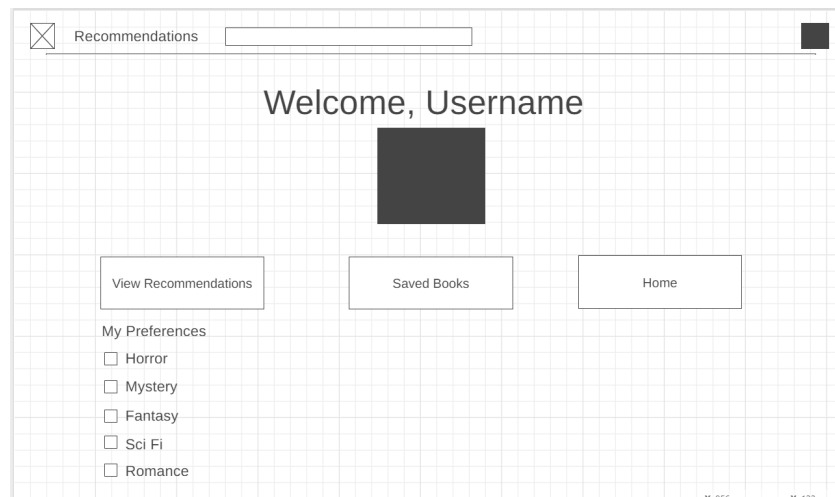


## Saved Books



The X box on the top left represents the application logo. The X box on the top right represents the user's profile picture.

## User Profile



Here, the dark gray squares represent the user profile picture. The X box still represents the application logo.

## Book View

### **Web Service Design:**

We are not using web services via APIs. Our books data is stored on Github, so we will be able to draw directly from the CSV files accompanying the webpages. So we will not need to access data from an API.

### **Database Design:**

We are working with our application data through MySQL. We have two databases: one to store user profiles (used in the Login, Registration, and User Profile pages) and one to store the books (used in the Search, Saved Books, and Recommendation pages). Our databases will look like the following, with some example data to demonstrate our intended columns better.

#### **Users Database**

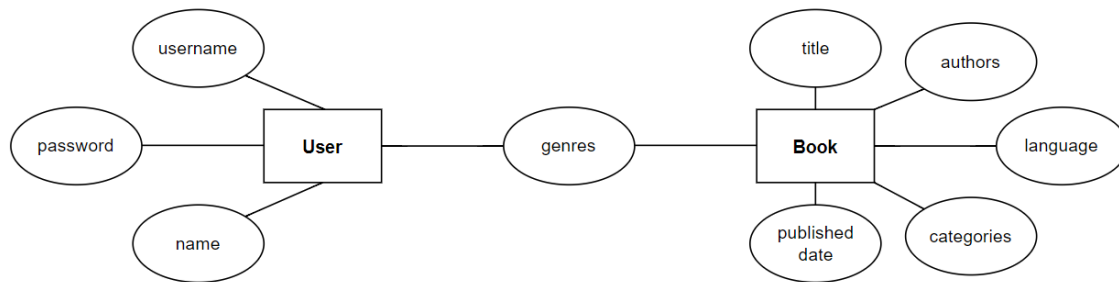
<b>Username (string)</b>	<b>Password (string)</b>	<b>Name (string)</b>	<b>Preferred Genres (array of strings)</b>
annalrahn	gh\$Q1@*mf	Anna Rahn	[‘Computers’, ‘Science’, ‘Engineering’]
EleTeacher1	tH1rDGr@d3	Emily Bire	[‘Education’, ‘Juvenile Fiction’]

#### **Books Database**

<b>Title (string)</b>	<b>Authors (array of strings)</b>	<b>Language (string)</b>	<b>Categories (array of strings)</b>	<b>Published Date (date)</b>
The CIO’s Guide to Oracle Products and Solutions	[‘Jessica Keyes’]	en	[‘Business & Economics’]	9/2/2014
Growing Through	[‘James Reapsome’,	en	[‘Religion’]	12/16/2009

Life's Challenges	'Martha Reapsome']			
-------------------	--------------------	--	--	--

## Entity Relationship Diagram



### Challenges:

- 1) **Recommendation Algorithm:** We are going to have code book recommendations based on what books the user likes. This algorithm could potentially be complex and rely on skills/technology we do not know how to use. To prevent this, we are going to simplify the recommendations so that it recommends books based on favorite genres.
- 2) **NodeJS:** Part of our project is going to depend on NodeJS. The current risk is making sure we fully understand how to use NodeJS so we can efficiently incorporate it into our project. We will get extra help if needed from the TAs and lecture/class material.
- 3) **UI Functionality:** We have working HTML but need to make sure that the frontend is easy for users to navigate. This will involve creating a layout that is intuitive and standard across every page and ensuring that each feature is functional for users. In order to do this, we will have people test our UI to make sure it is easily understandable.

### Individual Contributions:

Cody Aker: Created one of the new wireframes (search page), revised/ created new architecture diagram

Al Haddad: Added front end design from previous milestones and created a few new ones. Updated feature list to reflect priority and deleted redundant goals for clarity.

Anna Rahn: I created the entity relationship diagram and wrote the Database Design section of this milestone, then digitized the Registration, Saved Books, User Profile, and Book View wireframes. I wrote our to-do list detailing the next steps we need to take before our next team meeting sorted by type of task (HTML/CSS, Javascript, Node, etc.). I have also been working on the post request for the registration page (to add a new user to the users database).

Abigail Sullivan: I wrote the challenges section of the Milestone and worked on standardizing my teammates' HTML pages with my own.