Team Number: 011 – 6
Team Name: JAAAC

Team Members:

Jingda Yu

Anna Rahn

Abigail Sullivan

Al Haddad

Cody Aker

Application Name: NextPage

Application Description:

NextPage provides recommendations for readers based on their previous reading material. Users can rate books they have previously read, and receive recommendations based on their rating, combined with their previous ratings. To get started, NextPage will ask a few questions about the genres and kinds of books the user is interested in and begin recommending other books. The user will be able to mark book as "read" in order to stop receiving recommendations, and further refine their recommendations.

In order to make the algorithm improve itself, readers will be able to mark recommendations as interesting or not interesting, which will make similar readers receive more consistent recommendations. Readers will also be able to leave reviews with a star feedback system as well as comment their thoughts. A search algorithm may also be implemented in order to help readers narrow their recommendations faster. NextPage will be the ultimate book finding tool for bookworms and casual readers alike.

Vision Statement: To keep modern readers engaged in a never-ending supply of material tailored to their individual taste

Github link: https://github.com/cub-csci-3308-spring-2022/Project

Development Method: We will use agile development during this project. Each sprint, which will last a couple of weeks, we will set a goal to ensure we make a sufficient amount of progress throughout the semester. Each sprint will result in a working piece of software. We will use our bi-weekly team meetings to review what each group member has done since the last meeting (in a scrum-like fashion), determine how close we are to our current goal, set tasks to be done by the next meeting, resolve any blocks team members may be facing, and revise our plans along the way if necessary. These sprints, stories, and tasks will all be recorded on JIRA so that we can review where we are at and what needs to be done at any given time.

JIRA Link: https://csci-3308-spring22-202-6.atlassian.net/jira/software/projects/TV/boards/1/backlog

Communication Plan:

We plan to communicate using a text group chat as well as possibly Zoom and Discord if needed. Meetings will be in-person at the Geology building so that we can work out details without any loss of information. Sprints, stories, and individual tasks will be recorded on JIRA so they are accessible at all times.

Meeting Plan:

Tuesdays 5:30-6:30 and Fridays 5-6 PM at the Geology building (Benson Earth Sciences Library) (in-person)

With TA: Friday 6-6:15

Meeting ID: 565 981 3276

Proposed Architecture Plan: We will use HTML to develop the structure and primary content of the website. This includes the individual books' pages (with descriptions and reviews), the main feed of recommendations, the user's account page, and more. Then, we will use CSS to format and stylize each of the webpages. In order to animate features of our website, such as button clicks and pop-ups, we will use JavaScript. Each of these are front-end technologies.

On the server side, we will use Node.js to create the server the website will be hosted on and develop the login. Registered users will be stored in a database, along with their preferences, ratings, previously read books/authors, and reviews. We will manage this database by integrating PostgreSQL into Node.js.

The application will be accessible on Heroku.

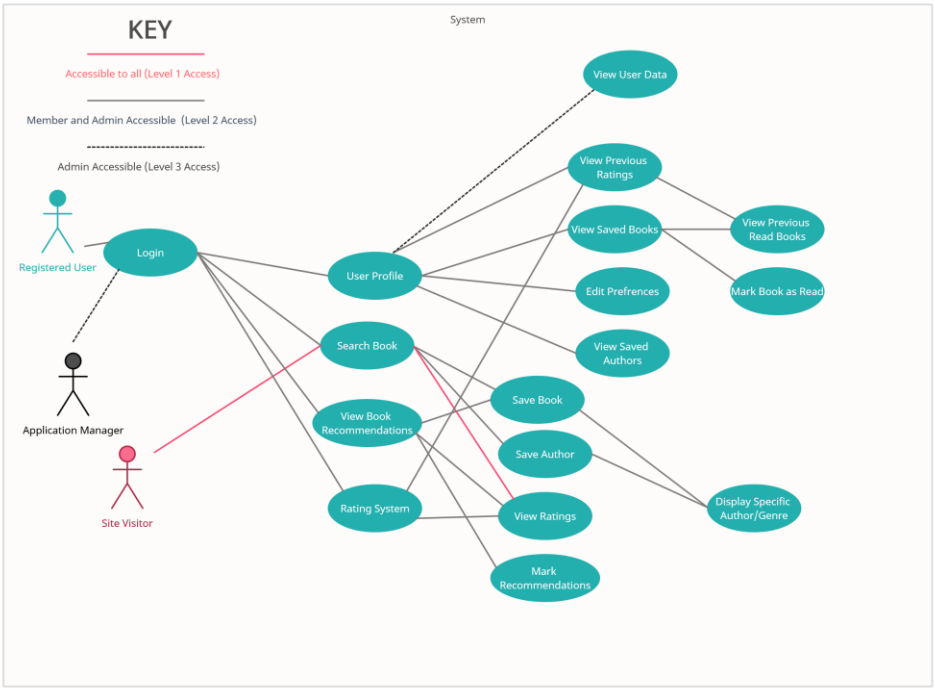
Architecture Diagram:
https://www.edrawmax.com/online/share.html?code=44f33c6a885f11ec968d0a54be41f961

Use Cases:

1. Create user account
2. Search for a specific book (by title, author, or genre)
3. Display a list of all books from a specific author or genre, sorted by most relevant to user
4. Write a rating for a specific book (to include a star-based rating and/or typed commentary)
5. View all ratings for a specific book provided by other users (to be sorted by date posted or star-based category)
6. Save book (for later viewing)
7. Save author (for later viewing)
8. Mark book as read
9. View list of saved books

10. View list of saved authors
11. View list of user's previous book ratings
12. View list of previously read books
13. View an endless list of book recommendations (the list continuously generates, sorting by most relevant)
14. Mark recommendation "interesting" or "not interesting"
15. Edit general preferences (such as preferred genres and authors) to immediately change the algorithm's results

Use Case Diagram:



Actors:

1. Site Visitor
2. Registered user
3. Admin