PickaBook
Find your next muse.

Sprint 1 Planning Document

Team 22

Logesh Roshan Ramadoss

Myeongsu Kim

Piyush Juneja

Shobhit Makhija

Sprint Overview

The primary goal of our first sprint is to establish and develop the skeleton of our application, i.e. the fundamentals services of our web application. With regard to the front end, we'd like to design and implement an interface that allows users to login/signup on the website, fill in their personal information and be able to view their user profile. An interface for resetting password or updating information will also be implemented through the course of this sprint. Finally, the user will be able to view any book in the database. The backend tasks, on the other hand, will be focused on designing and implementing the database logic and providing the RESTful API for the front end. Services like encryption of password and maintaining user session will be taken care of in the backend.

Scrum Master: Logesh Roshan Ramadoss

Meeting Schedule: *Saturday 2pm & Tuesday 11:30 am. Another meeting may be decided upon depending on requirement.*

Risks & Challenges:

Since all the team members are new to both the language frameworks being used here, like django, javascript, react, etc, and the entire software development cycle, we expect to run into a good amount of challenges pertaining to these topics. The initial sprint might be the hardest in terms of producing content, since we all need to learn the languages and frameworks for the application. Also, it is highly likely that we might run into issues with

syncing and incorporating the frontend and the backend, since both will be developed concurrently, neither team will have the finished product of the other team for development. This will cause a few issues in testing and will need to dealt with appropriately. Finally, since none of us have worked with each other, we need to ensure that everyone pulls their weight and works efficiently and follows the sprint.

Current Sprint Detail

Login Tasks

User Story #1

As a user, I would like to be able to create an account on the website.

#	Task Description	Time Estimate	Team	Owner
1	Create a sign-up form that allows user to enter their login information.	4	Front End	Shobhit
2	Ensure that the user information is valid and the password meets the set criteria.	2	Front End	Shobhit
3	Create a POST call to the backend Rest API to submit login details.	2	Front End	Shobhit
4	Encrypt user password and insert into database.	3	Back End	Logesh
5	Create a GET call to the backend to verify login status.	1	Front End	Shobhit
6	Redirect new user to enter personal information.	1	Front End	Shobhit
6	Send POST request to backend and insert information into database.	2	Back End	Logesh

Acceptance criteria:

- ❏ Given the signup form has been implemented, when the user tries to signup, then the form should be easy to navigate and complete.
- ❏ Given the signup form has been implemented, when wrong entries, such as invalid emails or short passwords, then the account will not be accepted and the form will be reloaded.
- ❏ Given the signup is implemented, when the password writes their password, then the password field should be hidden.

User Story #2

As a user, I would like the server to authenticate and verify my account during login.

#	Task Description	Time Estimate	Team	Owner
1	Create a database to store username and password.	3	Back End	Logesh
2	Create a REST API to obtain user info for login.	2	Back End	Logesh
3	Authenticate login by matching with database.	2	Back End	Logesh
4	Fetch the profile information related to the user and send it to frontend after logging in.	2	Back End	Logesh

Acceptance criteria:

- ❏ Given the user inputs the correct username and password, the app should look-up the username in the database and check for accurate credentials.
- ❏ GIven the user inputs accurate credentials, users should be granted access to their own profile.
- ❏ GIven we provide access to the user profile, user's profile information must be sent to the front end.

User Story #3
As a user, I would like to be able to login to the website with a created account.

#	Task Description	Time Estimate	Team	Owner
1	Create a login form that hides password with styling.	2	Front End	Myeongsu
2	Create a landing page for and after logging in.	4	Front End	Piyush & Myeongsu
3	Create a POST call to the backend Rest API to submit login details.	1	Front End	Myeongsu
4	Verify Login in backend	1	Back End	Logesh
5	Create a GET call to the backend to verify login status.	1	Front End	Myeongsu
6	Route to Profile Page after login	2	Front End	Myeongsu
7	Maintain session across the site	3	Front End	Logesh

Acceptance criteria:

- ❏ Given the backend is implemented, when the user enters their password, only their own password should be able to match the database.
- ❏ Given the frontend is robust, when the user enters a wrong password, then an “invalid password” message will be displayed.
- ❏ Given the users enter the correct information, then when they log in, they should be automatically redirected to the landing page.
- ❏ Given that the user successfully logs in, when the user goes to a new page, their authentication remains valid across all pages.
- ❏ Given the post request is implemented, when the user logs in, then only their data should be retrieved and no else’s.

User Story #4

As a user, i would like to be able to reset my password through an email.

#	Task Description	Time Estimate	Team	Owner
1	Write backend API to deal with forgotten password requests.	2	Back End	Logesh
2	Cross reference user's email for validity.	1	Back End	Logesh
3	Create a new password for the user, replace password in database and email new password.	3	Back End	Logesh
4	Add "forgot password" link to the login page.	2	Front End	Piyush
5	Send POST request to the backend with email id.	1	Front End	Piyush

Acceptance criteria:

- ❏ Given the password reset feature has been added, the user should receive an email with instructions on how to reset their password.
- ❏ Given the reset password is implemented, when the user opts to reset their password, they should be given a new password that is strong and is updated in the database. redirected to a web page where they set-up a new password.
- ❏ Given that the user enters a new passwords and saves it, user credentials should be updated in the database table.

User Story #5

As a user, I would like to be able to update my personal information.

#	Task Description	Time Estimate	Team	Owner
1	Display an update option on the user profile page to allow users to update their information.	2	Front End	Piyush
2	Create a GET call to the backend to acquire the user's old information from the database.	2	Back End	Shobhit
3	Create and display an 'Update Information' form and pre-populate it with the user's existing information in the database.	4	Front End	Shobhit
4	Create a POST call to the backend to send new information and redirect to user profile page.	2	Front End	Shobhit
5.	Use POST information in backend to update the user's information in the database.	2	Back End	Shobhit

Acceptance criteria:

- ❏ Given there is an update option on the user profile page, the user should be redirected to a update profile form.
- ❏ Given the user is redirected to the update profile page, the profile page must be pre-populated with existing information in the database.
- ❏ Given the update profile page is loaded with pre-existing information, the user should be able to update their information.
- ❏ Given the user updates their information, the changes should be visible in the database tables.

Book Information

User Story #6
As a user, I would like to be able to keep up with new books on the market.

#	Task Description	Time Estimate	Team	Owner
1	Create a web crawler to retrieve information on books from web.	3	Back End	Myeongu
2	Add parser to parse the retrieved information and split it into each book's info.	3	Back End	Myeongu
3	Clean the data for each book, add rating scores and insert it into the database.	2	Back End	Myeongu
4	Implement the crawler to function asynchronously in the backend.	3	Back End	Myeongu

Acceptance criteria:

- ❏ Given that the crawler is implemented, when the crawler is executed, new books should be added to the database.
- ❏ Given that the parser is implemented, then when the crawler produces data, the parser should clean and produce useful information, otherwise the database will be filled with incoherent information.
- ❏ Given that the crawler executes asynchronously, when a certain time has passed, the crawler should be called and run without halting responses to the client.
- ❏ Given that the parser is implemented, when the crawler gets a book already in the database, then the parser will throw it away.

User Story #7

As a user, I would like to be able to view the relevant details of a book, including the title, author, publication date, ratings and a blurb/synopsis.

#	Task Description	Time Estimate	Team	Owner
1	Create a database table for storing book information - title, author, publication date and blurb/synopsis.	3	Back End	Piyush
2	Create a Book Information page for displaying the relevant information for the user.	4	Front End	Piyush
3	Create a GET call to the backend to retrieve information for the book.	2	Front End	Piyush
4	Display information of book. Add links to the author and genre.	3	Front End	Piyush

Acceptance criteria:

- ❏ Given that the database and parser is implemented correctly, a book should not have multiple entries in the database.
- ❏ Given that the front end is implemented, when a user lands on a book page, then the user should easily be able to navigate through its details.
- ❏ Given that the page is implemented correctly, when a user clicks a link to the author or genre, then they should be redirected to said page, none of which should be broken.

User Story #8

As a user, I would like to be able to read the reviews as well as relevant questions & answers for each book.

#	Task Description	Time Estimate	Team	Owner
1	Create database tables for storing book reviews, and the Q&A.	3	Back End	Myeongsu
2	Add a 'Reviews' section to the Book Information page.	3	Front End	Piyush
3	Add a 'Q&A' section to the Book Information page.	3	Front End	Piyush
4	Add review and Q&A information to the GET request of the book information, through REST API serialization.	4	Back End	Myeongsu
5	Display the author of review/ Q&A and the number of likes and dislikes	2	Front End	Piyush

Acceptance criteria:

- ❏ Given that the database is implemented correctly, when a user looks at a review, then they should be able to see who wrote it and how well it is received (likes & dislikes).
- ❏ Given that the review system works, when a user tries to post more than one review on a book, then the application should deny the second review.
- ❏ Given that the page is implemented, when the user scrolls through it, then they will be able to see the top reviews and top Q&A

Author Information

User Story #9
As a user, I would like to get more details about the author from the book, and check out more books from the same author.

#	Task Description	Time Estimate	Team	Owner
1	Create a database table for storing author information - Background Information like Name, Date of Birth, Age, Genre, Books Released, Average User Review	3	Back End	Shobhit
2	Creating an Author Page displaying relevant Author Information like Personal Background, Genre and List of Books	2	Front End	Logesh
3	Create a GET call to the backend to retrieve Author Information	2	Front End	Logesh
4	Send Author information for backend	2	Back End	Shobhit

Acceptance criteria:

- ❏ Given the Author's name, in the book details, the user should be able to click on the Author's name and get redirected to the Author's Profile Page.
- ❏ Given the user can redirect to the Author's Profile Page, the page must be loaded with Author information from the database tables.
- ❏ Given the Author information is displayed, the user should be able to see user ratings and lists of author's other works.
- ❏ Given the database is implemented correctly, when the user looks at an Author's page, then the page must display other works by the author.

Lookup Information

User Story #10:

As a user, I would like to be able to search for books and authors.

#	Task Description	Time Estimate	Team	Owner
1	Add a navbar header to all pages.	3	Front End	Logesh
2	Add a search form to navbar	2	Front End	Logesh
3	Send POST request from search and redirect to author/book page.	3	Front End	Logesh
4	Lookup database for book/author and respond to request with book/author information if available.	3	Back End	Shobhit

Acceptance criteria:

- ❏ Given that the navbar is implemented, when the user scrolls through a page, the navbar remains at the top of the screen.
- ❏ Given the search bar is implemented, when the user enters a book name, then they should be redirected to the book's page.
- ❏ Given the frontend is implemented, when the user enters an invalid entry in the search form, the user will be redirected to a 404 page with the book not found.

Remaining Backlog

Functional Requirements

1. User account.

   As a user,

   ~~a. I would like to be able to Register and Create a new account.~~
   b. I would like to be able to link my Google or Facebook account.
   ~~c. I would like to be able to reset my password.~~
   ~~d. I would like to login and logout from my account.~~
   ~~e. I would like to set a profile picture.~~
   f. I would like to be able to set my profile's privacy to control if other users can see my profile.

2. User Profile

   As a user,

   ~~a. I would like to keep track of my profile.~~
   b. I would like to see a collection of my favorite books.
   ~~c. I would like to see my favorite genres.~~
   d. I would like to see all the authors I'm following.
   ~~e. I would like to see my profile and information.~~
   f. I would like to view any public users' profile.
   g. I would like to be able to set my profile to private.
   h. I would like to be given recommendations on what to read in the future.

3. Viewing Content

   As a user,

   ~~a. I would like to search for any book.~~

   ~~b. I would like to see a page with the information and ratings + reviews for the book.~~

   c. I would like to add a book to my favorite collection.

   d. I would like to search for an author.

   ~~e. I would like to get the author of a book.~~

   f. I would like to be able to follow the author to get notifications for any new releases.

   ~~g. I would like to see all the releases from an author.~~

4. Book Reviews

   As a user,

   a. I would like to be able to write reviews for books.

   ~~b. I would like to see all the reviews of a book.~~

   c. I would like to give ratings for books.

   d. I would like to see the ratings of a book.

   e. I would like to see all the reviews I have given.

   f. I would like to like or dislike reviews given by other people.

5. Exploration System

   As a user,

   a. I would like to be able to explore different titles.

   b. I would like to browse through new releases.

   c. I would like to browse through the top rated books.

   d. I would like to filter by the year of publication.

   e. I would like to filter by genre of books.

6. Back End

   As a developer,

   a. ~~(If time allows) I would like the server to crawl the web for new books.~~
   b. ~~(If time allows) I would like to use the crawler to parse new books and update the database automatically~~
   c. I would like to keep track of user preferences in the database.
   d. I would like to recommend new books based on user preferences.

Non Functional Requirements

1. The backend must follow a RESTful API development process.
2. The login information must be stored securely.
3. The web application must be easy to navigate.
4. The application must be secure from issues like SQL injections and cross site scripting.
5. The design must be scalable.