Mixbook

Mixology Application

Sprint 2 Planning Document

https://github.com/alexcoll/mixbook

Team 3:

Alexander Coll (acoll@purdue.edu), Matthew Fouts (foutsm@purdue.edu),

John Tyler Preston (prestoj@purdue.edu), Nicholas Zenobi (nzenobi@purdue.edu)

Sprint Overview: For second first sprint we will be working on the rating of reviews, filtering and sorting features, and UI design. By the end of the sprint users will be able to rate a review as helpful or not helpful. He or she will be able to view both pre-populated and user created recipes. Users will also have access to even more ingredients from which to choose. Users will experience an enhanced user interface and experience, such as pagination/pagination-like viewing of results, as well as general UI/UX cleanups. Users will also be able to lock their account if desired.

Scrum Master: John Tyler Preston

Meeting Schedule: Tue 4:30-5:30

Risks/Challenges: Designing an enhanced user interface to provide a better user experience will be challenging as none of the group members have much experience with UI/UX development. The bulk of this sprint will be revolving around designing an enhanced UI/UX while improving existing features and adding some new features, both on the frontend and the backend. Another difficulty that we will face will be adapting to changes to compensate for what did not go well in the first sprint. Overall, the second sprint will be a combination of overcoming our difficulties from the first sprint as well as adding new content for users on top of enhancing and adding new features.

User Story 1: As a user, I would like to choose from pre-populated recipes as well as user-added recipes.

#	Task Description	Estimated Time	Owner
1	Come up with and create pre-populated recipes	10 hours (2.5 hours per team member)	Alex, Matt, Tyler, and Nick
2	Write SQL creation script of the pre-populated recipes and associated data	10 hours	Tyler

Acceptance Criteria:
- ❏ Given the pre-populated recipe exists, when the user opens a pre-populated recipe, then the pre-populated recipe should look professional.
- ❏ Given the pre-populated recipe exists, when the user opens a pre-populated recipe, then the pre-populated recipe should be grammar error free.
- ❏ Given the pre-populated recipes exist, when the user opens up the list of recipes, then 100 pre-populated recipes should appear.
- ❏ Given the pre-populated recipes exist, when the user opens up the list of recipes, then the pre-populated recipes that appear should be varied in type and style.
- ❏ Given the pre-populated recipes exist, when the user opens any pre-populated recipe, then the same recipe author information should appear.

The frontend for this is mostly users entering/creating the recipes. The backend mostly deals with writing up the SQL to pre-populate the database with these new recipes and associated data.

User Story 2: As a developer, I would like for users to be able to lock their accounts.

#	Task Description	Estimated Time	Owner
1	Implement user	6.5 hours	Nick

	interface in app to request to lock user account		
2	Implement handling locking and unlocking a user's account on backend	5 hours	Tyler
3	Unit and usability testing	3 hours	Alex

Acceptance Criteria:
- ❏ Given the user exists, when the user locks their account, then no further authentication tokens will be granted for that user.
- ❏ Given the user has locked their account, when the user tries to login again, then they will be prompted with a message indicating the account is locked and that an email has been sent to the email associated with the account containing instructions to unlock the account.
- ❏ Given the user has tried to login after locking their account, when they click on the link in the email they received, then their account will become unlocked.
- ❏ Given the user has locked their account and the last authentication token associated with their account expires, when the user tries to make an authenticated request, then the authorization will fail and be denied access.
- ❏ Given the user has locked their account and the last authentication token associated with their account has not expired, when the user tries to make an authenticated request, then the authorization will succeed and be granted access.

The frontend for this is mostly to serve as the interface for a user to request to lock their account as well as the interface for the user to request to unlock their account by trying to login like normal. The backend for this is mostly dealing with processing the locking and unlocking of a user's account, as well as sending email notification of both locking the account as well as unlocking the account.

User Story 3: As a user, I would like to have pagination of results.

#	Task Description	Estimated Time	Owner
1	Update how lists are shown on the recipes page to one with pagination	3 hours	Nick
2	Unit testing and usability testing for recipes page	3 hours	Nick
3	Update how lists are shown on the mydrinks page to one with pagination	3 hours	Nick
4	Unit testing and usability testing for mydrinks page	3 hours	Nick
5	Update how lists are shown on the ingredients page to one with pagination	3 hours	Nick
6	Unit testing and usability testing for ingredients page	3 hours	Nick

Acceptance Criteria:
- ❏ Given that the number of results on the recipes page are higher than the pagination amount, when the page is rendered, then the list is displayed in pages the user can scroll through.
- ❏ Given that the number of results on the mydrinks page are higher than the pagination amount, when the page is rendered, then the list is displayed in pages the user can scroll through.
- ❏ Given that the number of results on the ingredients page are higher than the pagination amount, when the page is rendered, then the list is displayed in pages the user can scroll through.

- ❏ Given that the number of results on the a page with a list are higher than the pagination amount, when the user scrolls through the pages, then the displayed item will show up immediately.
- ❏ Given that the number of results on the ingredients page are higher than the pagination amount, when the user loads the page, then the user should be able to see how many items and pages there are.

The frontend is where all of the work will be for this issue. On every page where the user is shown a list, that list will now be paginated. This will show the user the next page of information.

User Story 4: As a user, I would like to have access to even more ingredients from which to choose.

#	Task Description	Estimated Time	Owner
1	Come up with more ingredients	4 hours (1 hour per team member)	Alex, Nick, Matt, Tyler
2	Add ingredients to database	8 hours	Alex
3	Verify ingredients are stored in the database	1 hour (0.5 hours per team member)	Alex, Tyler

Acceptance Criteria:
- ❏ Given new ingredients are added, when the ingredient list is loaded, then the newly added ingredients will be displayed.
- ❏ Given the new ingredients are in the database, when a user is creating a recipe, then the new ingredients will be available for them to add to the recipe.
- ❏ Given the new ingredients are in the database, when a user adds one of the new ingredients to their inventory, then the ingredient will show up in their inventory.
- ❏ Given the new ingredients are in the database, when a user loads the ingredient list, then the newly added ingredients will be typo-free.
- ❏ Given the new ingredients are in the database, when a user loads the ingredient list, then the newly added ingredients will be diverse from what already existed as ingredients.

The backend is where all of the work will be for this issue. The ingredients will be added to the database and added in the database schema creation script.

User Story 5: As a user, I would like to say if reviews are helpful or not.

#	Task Description	Estimated Time	Owner
1	Add thumbs up and thumbs down buttons to individual reviews	6 hours	Alex
2	Unit testing and usability testing for reviews page	3 hours	Alex
3	Update database to store helpful/unhelpful ratings	2.5 hours	Tyler
4	Implement server API to mark a review as helpful/unhelpful	2.5 hours	Tyler

Acceptance Criteria:
- ❏ Given a user is viewing reviews for a recipe, when they tap the helpful button, then the review is marked as helpful and the rating is stored in the database.
- ❏ Given a user is viewing reviews for a recipe, when they tap the unhelpful button, then the review is marked as unhelpful and the rating is stored in the database.
- ❏ Given a user is viewing a recipe, when they look at a particular review, then the count of helpful votes is displayed.
- ❏ Given a user is viewing a recipe, when they look at a particular review, then the count of unhelpful votes is displayed.
- ❏ Given a user is viewing a recipe, when they look at a particular review, then the counts of helpful and unhelpful votes will load in under a second.

The frontend will be updated to show thumbs up (helpful) and thumbs down (unhelpful) buttons next to each review. The ratings will be stored on the back end in the database.

User Story 6: As a user, I would like to filter the reviews based on rating.

#	Task Description	Estimated Time	Owner
1	Implement filtering functionality to frontend reviews page	6 hours	Alex
2	Add filter dropdown menu to the frontend interface	6 hours	Alex
3	Unit testing and usability testing for frontend filtering functionality	4 hours	Alex

Acceptance Criteria:
- ❏ Given the user is looking at the list of reviews, when they tap filter, then a drop down menu will allow them to input filter criteria.
- ❏ Given the user is looking at the list of reviews, when they tap filter and filter only helpful reviews, then the list will only show helpful reviews in descending order.
- ❏ Given the user is looking at the list of reviews, when they tap filter and filter only unhelpful reviews, then the list will only show unhelpful reviews in ascending order.
- ❏ Given a user is looking at the list of reviews, when they tap filter and filter by some criterion, then the list will update without freezing the user interface.
- ❏ Given a user is looking at the list of reviews, when they tap filter and filter by some criterion, then the list will update in less than a second.

All the work for this user story is on the frontend. The app will only show reviews that meet the filtering criteria and sort them accordingly.

User Story 7: As a user, I would like to have a profile rating that reflected the ratings of the recipes I have created.

#	Task Description	Estimated Time	Owner
1	Calculate and	4 hours	Nick

	display profile rating on own account page		
2	Unit testing and usability testing for account page	2 hours	Nick
3	Calculate and display profile rating when viewing other users profiles	4 hours	Nick
4	Unit testing and usability testing for other user's account page	2 hours	Nick
5	Implement handling profile rating on backend	5 hours	Tyler

Acceptance Criteria:
- ❏ Given that the user is logged in, when they go to their account page, then they should see their profile rating.
- ❏ Given that the user has created recipes, when they get rated by another user, then the user should have a profile rating.
- ❏ Given that the user is looking at another user's profile, when the profile is loaded, then the user should be able to see their profile rating.
- ❏ Given that the user is logged in, when another user rates the user's recipe, then the user's profile rating should update when reloaded.
- ❏ Given that the user is viewing another user's profile, when a recipe of the profile being viewed is rated, then that profile rating should be update when the profile is reloaded.

The bulk of the work for this issue will be on the frontend. Whenever the user looks at their own profile they should be able to see their rating and whenever they look at another user's profile they should be able to see their rating. The rating should also be updating whenever the page is loaded. The backend will handle storing and managing profile ratings.

User Story 8: As a user, I would like to be able to find and sort users by their profile ratings.

#	Task Description	Estimated Time	Owner
1	Create page on frontend that will display list of users and their ratings	8 hours	Matt
2	Create option that allows users to be sorted based on ratings	6 hours	Matt
3	Unit Testing/Usability Testing	4.5 hours	Matt
4	Implement a way of receiving users and their ratings from the backend	5 hours	Tyler

Acceptance Criteria:
- ❏ Given that a user is logged in, when they load this page, then they should be shown a list of all users.
- ❏ Given that the page is loaded, when the list of users is shown, then the user should see the correct rating should be shown for each user.
- ❏ Given that the correct user rating is shown, when the user sorts based on high rating, then then highest ratings should be on top.
- ❏ Given that the correct user rating is shown, when the user sorts based on high rating, then then lowest ratings should be on top.
- ❏ Given that the list is loaded, when the user clicks on a user, then they should see their profile.

The bulk of the work for this issue will be on the frontend. Whenever the user looks at the list of users they should be able to see the rating of each user and a list of all users. The ratings for each user should also be updating whenever the page is loaded. The backend will handle retrieving a user's ratings.

User Story 9: As a user, I would like to see recipes that I have made.

#	Task Description	Estimated Time	Owner
1	Create a page that allows user to see only the recipes that they have made	8 hours	Matt
2	Allows sorting of recipes the user has made based on rating	6 hours	Matt
3	Unit Testing/Usability Testing	4 hours	Matt
4	Implement a method to retrieve user recipes from backend	6 hours	Tyler

Acceptance Criteria:
- ❏ Given that a user has made recipes, when they load this page, then the user should be able to retrieve based on the individual user.
- ❏ Given that a user has loaded this page, when they look at the list, then the user should see the list of all recipes they have made.
- ❏ Given that a user has loaded this page, when they view the list, then they should see not any recipes that aren't their own.
- ❏ Given that the user has loaded this page, when they view the list, then they should be able to sort the recipes based on their ratings.
- ❏ Given that the user has loaded this page, when they click on a recipe, then they should see the details of the recipe.

The bulk of the work for this issue will be on the frontend. Whenever the user looks at the list of their recipes they should be able to see the rating of each recipe. The ratings for each recipe should also be updating whenever the page is loaded. The backend will handle retrieving recipes that a user has made.

Remaining Product Backlog:

Functional

1. **As a user, I would like to receive push notifications for a variety of events (if time allows).**
2. As a user, I would like to be able to recommend recipes to other users.
3. **As a user, I would like to see images of cocktails (if time allows).**
4. **As a user, I would like to add and remove images of cocktails (if time allows).**
5. As a developer, I would like to overhaul the current error handling platform.
6. As a developer, I would like to overhaul the interchange format of data to a more standardized version of JSON.
7. As a developer, I would like to remove the use of deprecated functions as much as possible.
8. As a developer, I would like overhaul the request handling to improve and make request handling more standardized.
9. As a developer, I would like redesign the add recipe page to be more user-friendly.

Non-Functional

1. Should use some sort of near-real-time or real-time solution for updates, utilizing AWS services where possible
2. Should have the ability to receive feedback in some from
3. Should thoroughly document all code base in order to enhance maintainability

Time Allocation of Work Per Group Member + Total:

Alex's Hours	Matt's Hours	Tyler's Hours	Nick's Hours	Total Hours
40	40	40	40	160