The Boys of 103 - Team 19
mixbook: Mixology Application
Product Backlog

Alexander Coll, Nick Zenobi, Tyler Preston, Wyatt Dahlenburg

Problem Statement

No drink beats a well mixed cocktail at your favorite bar. However, what if you don't want to leave your house or pay $10 for a drink that you can make yourself for the fraction of the cost? You could mix your own drinks at home, but you're not a bartender: you don't know what drinks you can make with what you have in your kitchen. Our mixology app aims to solve that problem: give it a list of ingredients and Mixbook will give you detailed recipes using only those ingredients. Currently there are a multitude of outdated mixology apps that just solely list recipes. Our application serves to create a social media where users can share recipes and review drinks through a simple user interface.

Background information

For the longest time, drink recipes were in paper-bound books and usually only bartenders would have these books. With the advent of the internet, websites started cropping up, although shoddy, with some drink recipes here and there, though nowhere near complete or user friendly. As mobile applications came about, this technology shifted to become mobile app based, though none were anything more than static drink recipe books. Currently there is no existing mobile application that allows a user to find recipes based on the ingredients that they have as well as allow the sharing and reviewing of drink recipes in a user friendly manner. It is frustrating searching for hours to find drink recipes that are craftable with only the ingredients that you currently have, and this mobile application aims to change that.

Environment

- An Android application front end
- A MySQL database
- A Java based backend utilizing a three-tier Spring MVC based architecture
- AWS hosting of database and other web server

Functional Requirements

Backlog ID	Functional Requirement	Hours	Status
1	As a user, I would like to be able to edit a list of ingredients I own.	5	Planned for sprint 1

2	As a user, I would like the app to tell me what drinks I can make with those ingredients.	10	Planned for sprint 1
3	As a user, I would like to view different drink recipes.	5	Planned for sprint 1
4	As a user, I would like to rate and write reviews of drink recipes.	5	Planned for sprint 2
5	As a user, I would like to see ratings and reviews of drink recipes.	5	Planned for sprint 2
6	As a user, I would like to share custom recipes.	5	Planned for sprint 1
7	As a user, I would like the option of not needing to create an account to utilize the core services of the application.	5	Planned for sprint 1
8	As a user, I would like to be able to view recipes that include but are not limited to ingredients that I have.	5	Planned for sprint 2
9	As a user, I would like the option of being able to have my own account.	5	Planned for sprint 1
10	As a developer, I would like there to be minimal load times for all functionalities in the application.	10	Planned for sprint 2
11	As a user, I would like to be able to stay logged in.	10	Planned for sprint 2
12	As a user, I would like to be able to change my password.	5	Planned for sprint 1
13	As a user, I would like to be able to edit my own recipes.	3	Planned for sprint 2
14	As a user, I would like to be able to edit my own reviews.	2	Planned for sprint 2
	Total	**80**	

Non-Functional Requirements

1. Should be able to use this application on an Android device
2. Should use some sort of near-real-time or real-time solution for updates
3. Should be scalable to be user demand
4. Should meet standard security practices
5. Should have an interface that is simple enough so that the average user would be able to use the application without struggling
6. Should be able to have negligible downtime while allowing for continuous development.
7. Should be able to be high performing while maintaining proper security practices
8. Should have the ability to receive feedback in some from
9. Should design database schema independent of application so that it follows proper database schema design and can be ported for other uses
10. Should incorporate a three-tier architecture

Use Cases

Case: View/search for drink recipe

Action	**System Response**
User enters Recipes section	App shows recipe section
User taps Search	Search box appears
User types in query	Searches for drinks containing query
Tap show recipes	Recipe list appears

Case: See drink recipes possible with ingredients

Action	**System Response**
User enters Recipes section	App shows recipe section
User adds ingredients	Ingredients added to list
Tap show recipes	Recipe list appears

Case: Create custom recipe

Action	**System Response**
User enters Recipes section	App shows recipe section

User taps create	Create recipe from appears
User enters in ingredients, uploads picture	
User taps save	Recipe is submitted to recipe database

Case: Rate/review a recipe

Action	**System Response**
User taps on reciepe	App shows recipe details
User taps add review	Dialog prompts for star rating and text review
User taps submit	Review is submitted to database

Case: View recipe rating/reviews

Action	**System Response**
User enters Recipes section	App shows recipe list view, prompts for ingredients
User selects recipe	Recipe details displayed
Tap show reviews	Reviews are shown

Case: Logging in without an account

Action	**System Response**
Users is at login screen	
User taps continue without logging in	App goes to main screen, user is not logged in

Case: Logging in with account

Action	**System Response**
User is at login screen	
User taps login/create account	Login/create dialog shown

User logs in/creates an account	User is logged in, main screen is shown

Case: View recipes with ingredients I don’t have

Action	**System Response**
User enters Recipes section	App shows recipe view
User adds ingredients	Ingredients added to list
Tap show recipes	Recipe list appears, includes suggestions for drinks that require ingredients the user does not have

Case: Persistent logins

Action	**System Response**
User open the app after logging in a previous time	App starts at main screen with user already logged in

Case: Edit recipe rating/reviews

Action	**System Response**
User enters Recipes section	App shows recipe list view, prompts for ingredients
User selects recipe	Recipe details displayed
Tap show reviews	Reviews are shown
User taps edit on his review	Edit dialog is shown
User taps save	Review is updated.

Case: Change password

Action	**System Response**
User enters Settings section	Settings are shown
User taps change password	Change password dialog is displayed

User taps save	Password is updated in database

Case: Edit custom recipes

Action	**System Response**
User enters recipe section	Main recipe screen is shown
User enter the My Recipes section	User’s own custom recipes are shown
User taps edit on a recipe	Recipe ingredient from displayed
User taps save	Recipe is updated in database

Case: Minimal load time for loading all possible recipes

Action	**System Response**
User enters recipe section	Main recipe screen is shown
User adds ingredients	Ingredients added to list
Tap show recipes	Recipe list appears in less than a second

Case: Entering in ingredients

Action	**System Response**
Users enters recipe section	Main recipe screen is shown
User adds ingredients	Ingredients added to the list

Case: Removing ingredients

Action	**System Response**
Users enters recipe section	Main recipe screen is shown
User removes an ingredient	Ingredient is removed from the list