

# Testing Plan For PrepPal

## Change Log

Version	Change Date	By	Description
1.0.0	27/02/2024	Kenny Li	Initial testing plan
2.0.0	28/03/2024	Kenny Li	Update feature requirements and project roles

# 1) Introduction

## 1.1 Scope

Functional and non-functional requirements of the software that **will be** tested.

1. Account Manager
  - a. Log in as a user
  - b. Sign up as a user
  - c. Edit profile
2. Create/View Recipe
  - a. Create own recipes
  - b. View my own created recipes
3. Recipe collection
  - a. Save other users' recipes
  - b. View saved recipes
4. Search/Explore
  - a. View random list of public recipes
  - b. View list of public recipes based off of search requirements
    - i. Title
    - ii. Author
    - iii. Ingredients
    - iv. Description
    - v. Cooking time
5. Meal Planner
  - a. View recipes planned in a given day/month
  - b. Add/remove a recipe for a specific day on the calendar
6. Comment/Rating
  - a. Add a review and rating on recipe by user
  - b. View comments/rating made on a recipe

Non-functional requirement:

- can respond to 20 users with a total of 200 requests per minute concurrently

## 1.2 Roles and Responsibilities

Name	Net ID	GitHub username	Role
Kenny Li	lik5	lik5	Developer, Test Manager
Ryan Loeppky	loepk10	Rloepkky	Developer
Henry Wong	umwongh	hwbit	DevOps, Back-end developer
Frieda Bi	bif	bfrieda	Developer
Izan Cuetara Diez	cuetarai	algorizan	Developer

## 2) Test Methodology

### 2.1 Test Levels

#### Feature 1: Account Manager

Unit tests:

- Create an account with a valid username and password.
- Attempt to create an account with a username already in use.
- Attempt to create an account with a blank username.
- Attempt to create an account with a blank password.
- Update a user with valid information.
- Attempt to update a user with invalid user ID/username/password.
- Look up a registered user.
- Look up a user that does not exist.

Integration tests:

- Create a new account and see profile/collection/new recipe/calendar/shopping list in the navigation bar.
- Sign in and be able to edit profile

Acceptance tests:

1. As a new user, I want to be able to create an account.
  - a. I open the PrepPal website.
  - b. I click on the 'Login' button at the top of the webpage.
  - c. I click on the 'Sign up' button at the bottom of the webpage.
  - d. I fill out my desired username in the Username text area.
  - e. I fill out my desired password in the Password text area.
  - f. I click on 'Sign up' to complete the signup process.
  - g. If I get a message saying that the username is already taken, I repeat step (d), and (f).
2. As a registered user, I want to be able to log into my account.
  - a. I open the PrepPal website.
  - b. I click on the 'Login' button at the top of the webpage.
  - c. I fill out my username in the Username text area.
  - d. I fill out my password in the Password text area.
  - e. I click on the 'Login' button to complete the login.
  - f. If I get a message saying that I have the wrong username or password. I repeat Steps (c), (d), and (e).

#### Feature 2: Create/View Recipes

Unit tests:

- Create a recipe with valid fields.
- Attempt to create a recipe with invalid author/title/ingredients/instructions/serving size/prep time/cooking time.

Integration tests:

- Sign in and create a new recipe

#### Acceptance test:

1. As a user, I want to be able to create a recipe.
  - a. I open the Preppal application.
    - i. I log in if I am not already logged in
  - b. I click on the 'New Recipe' button at the top of the webpage.
  - c. I enter the name of the recipe in the field labeled 'Title'.
  - d. I enter a description of the recipe in the field labeled 'Description'.
  - e. I fill out the first ingredient for my recipe.
    - i. If I have another ingredient, I click the 'Add Ingredient' button and repeat this step.
  - f. I fill out the first step instructions for my recipe.
    - i. If I have another step, I click the 'Add Instruction' button and repeat this step.
  - g. I fill out the number of people this recipe can serve under 'Servings'.
  - h. I fill out the amount of time in minutes required to prepare the meal in the text area labeled 'Prep Time'.
  - i. I fill out the amount of time in minutes required to cook the meal in the text area labeled 'Cooking Time'.
  - j. I click 'Submit' to complete my recipe.
2. As a user, I want to be able to view a recipe
  - a. I open the Preppal application.
  - b. I click on the logo at the top left of the webpage.
  - c. I click on one of the recipes I want to view
  - d. View recipe I selected

### **Feature 3: Recipe Collection**

#### Unit tests:

- Get all recipes of a user.
- Get all favorited recipes.
- Save another user's recipe.
- Unsave a favorited recipe.

#### Integration tests:

- Sign in and view own recipe collection
- Sign in and view favorited recipe collection

#### Acceptance test:

3. As a user, I want to be able to view my own recipes
  - a. I open the Preppal application.
    - i. I log in if I am not already logged in
  - b. I click on 'Collections' at the top right of the webpage.
  - c. I click "Own Recipes" tab
  - d. I click on one of the recipes I want to view
  - e. View recipe I selected
4. As a user, I want to be able to view a saved recipe

- a. I open the Preppal application.
  - i. I log in if I am not already logged in
- b. I click on 'Collections' at the top right of the webpage.
- c. I click "Favourites" tab
- d. I click on one of the recipes I want to view
- e. View recipe I selected

#### **Feature 4: Search and Explore Recipe**

Unit tests:

- Search for a recipe by name.
- Search for valid recipes that share the same name.
- Search for recipes based on tags.
- Search recipes based on prep time.
- Search recipes based on cook time.
- Save/favourite recipes.
- Get a list of user saved recipes.
- Get a list of recipes created from an author.
- Search for a recipe and get results back for the searched recipe.
- Look up a recipe with a recipe ID number.
- Attempt to get a recipe with an invalid recipe ID/author/recipe.
- Look up an author's recipe.
- Search for a recipe with a valid name.

Integration tests:

- Sign in to save a recipe
- Get other user's created recipes

Acceptance tests:

1. As a user, I want to be able to search for a recipe.
  - a. I open the Preppal application.
  - b. At the top of the page in the search bar, I type in the recipe name I want to find
  - c. I click on the recipe that I want
  - d. I view the recipe
2. As a registered user, I want to be able to save a recipe.
  - a. I open the Preppal application.
  - b. At the top of the page in the search bar, I type in the recipe name I want to find
  - c. I click on the recipe that I want to save
  - d. I click the heart button to save
  - e. I click on "Collections" button in the top right
  - f. I click on the "Favourites" tab
  - g. I see the recipe I saved

#### **Feature 5: Meal Planner**

Unit tests:

- Attach a recipe(s) to a day.
- Attach a recipe(s) to a week.

- Get planned recipes for a single day.
- Get planned recipes for a week.
- Get planned recipes for a month.
- Remove a recipe from a day.
- Attempt to remove a recipe that is not attached the specified day
- Attempt to remove a recipe that is invalid
- Attempt to attach a recipe that is invalid

Integration tests:

- Can add a recipe to a calendar when the user is logged in
- Can remove a recipe from a calendar when the user is logged in

Acceptance tests:

1. As a user, I want to be able to add a recipe to day.
  - a. I open the Preppal application.
  - b. I want to click on “Calendar” in the top right
  - c. I click on the day that I want to add a recipe
  - d. I click the “Add recipe” button
  - e. Select the recipe I want to add
  - f. View the recipe in the calendar
2. As a user, I want to be able to remove a recipe from a day.
  - a. I open the Preppal application.
  - b. I want to click on “Calendar” in the top right
  - c. I click on a day that I want to remove a recipe from
  - d. I click the “x” button next to the recipe I want to remove
  - e. View the calendar without removed recipe

## **Feature 6: Commenting, Reviewing, and Rating Recipes**

Unit tests:

- Add a comment to a recipe.
- Add a reply to a comment.
- Add a ‘like’ to a comment.
- Get the total number of comments attached to a recipe.
- Rate a recipe.
- Get the number of ratings for a recipe.
- Get the average rating for a recipe.
- Get the number of people that saved the recipe.
- Attempt to add a comment to an invalid recipe
- Attempt to add a rating to an invalid recipe
- Rating should be 0 when no ratings exist

Integration tests:

- A comment can only be made when a user is logged in
- A rating can only be made when a user is logged in

Acceptance tests:

1. As a user, I want to be able to leave a comment/review on a recipe.

- a. I open the Preppal application and arrive at the home page.
  - b. I click on a recipe that I want to leave a comment/review on.
  - c. I scroll to the bottom of the recipe.
  - d. I click on the “comment” button.
  - e. I type in the comment I want to leave.
  - f. I click on the “submit” button.
2. As a user, I want to be able to leave a rating on a recipe.
  - a. I open the Preppal application and arrive at the home page.
  - b. I click on a recipe that I want to leave a rating on.
  - c. I click on the “leave rating” button.
  - d. I type in the rating out of 5 I want to leave.
  - e. I click on the “submit” button

## **2.2 Test Completeness**

Criteria for test completeness:

- Automated testing is executed
- All bugs should be reported and addressed
- Tests should be created to ensure previous issues and bugs do not occur again
- Regression testing must pass before committing and merging into dev and main branch
- 100% back-end code coverage testing
- 10 unit tests per core feature
- 10 integration tests in total to cover the core features
- 1 acceptance test per core feature

## **3) Resource & Environment Needs**

### **3.1 Testing Tools**

The following tools will be used for testing:

- Automation and Regression: GitHub Actions
- Backend test: Jest, Supertest
- Frontend test: Jest, React-Testing
- Issue/Bug-tracking: GitHub Issues
- Load testing: Jmeter
- Other: Postman

### **3.2 Test Environment**

- Windows 10 and above
- Jenkins
- Docker

## 4) Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface