Software Requirements Specification

For 2204 – IT Development

Team – Recipe Gods

September, 2021

Version - 1

Team Members

(Names, Titles, and Roles of project team members)

Ian McConaghy - lead designer, technical expert, full-time member of Recipe App project. Ian will work on writing the technical documentation of the project, invent the bleeding-edge design of our application.

Michael Kashkov - lead back-end designer, inventor of app features, full-time member of Recipe App project. Michael will work on feature design and back-end app structure

1 INTRODUCTION

1.1 Project Overview

Our project team developed and conducted a survey among our classmates and their family members of each participant to reach the conclusion of which app has the best potential on the Canadian market and what services our local community needs that could be satisfied by features of our application. We concluded that Recipe App would be our perfect choice to practice Native React coding, meet the expectations of our code expert and teacher - Mr. Xing Liu and potential customers.

For our project we are choosing to code a mobile recipes app for users to be able to view, create and share their favorites with other users. Our application will allow users to create a library of all of their recipes in the convenience of their phones.

1.2 Goals and Objectives

Goal:

- Authorization with Google Authenticator
- View recipes from all user
- Full CRUD functionality for Recipes
- Help share different ethnic backgrounds with the art of food
- Write a review for other user's recipes
-

2 GENERAL DESIGN CONSTRAINTS

2.1 Software Environment

The front-end of our application is centered around React Native as an imminent software environment constraint of 2204.

As for the back-end of our application, we decided to go with Rails API.

Furthermore, for the coding environment, the best tool that we can think of is Visual Studio Code. It's lightweight with lots of extensions that would make the app development more comfortable and productive

2.2 Hardware Environment

For the duration of the 2204 course, we will be able to use anytime VCC labs that are set up with Windows computers. In addition, we are not restricted to code on our personal laptops. Michael Kashkov has a Lenovo Windows laptop and Ian McConaghy has a MacBook Pro. In order to test our application, we are going to use Ian's iPhone and Michael's Xiaomi Android phone

2.3 Other Constraints

Other constraints and conditions that are outside the control of the project that limit the design alternatives:

Schedule. We have time to complete this project till 2022. Three months total.

Budget. Our budget is limited to buying coffee to keep us awake.

Policy and Regulation. The team shall not share any code with other teams or use the code of competing teams.

3 NONFUNCTIONAL REQUIREMENTS

3.1 Operational Requirements

Supported operating systems for end users are Android and Apple's iOS.

Hosting for back-end is Heroku free tier 3600 SQL CRUD operations per hour.

Supported user devices are limited to phones that are no older than 5 years.

3.2 Performance Requirements

Our application will need a steady internet connection in order for the user to pull recipes from our database or upload one of their own.

3.3 Security Requirements

Our application will need a google account to authenticate for the application. Our users will use this login to store their recipes within our database. The emails used to sign up for this application will not be shared with any other users. It's up to the full discretion of the users to choose which information they make public to other users of this application (I.e. Name, Age, Email).

3.4 Other Requirements

Capacity:

All the user's data will be stored in the database, so they will not have to use any of their phone's storage. The only storage needed on the user's side is for downloading the application. In the future if we need to upgrade the database size, we may implement a paid subscription for our application. For the time being capacity does not seem to be an issue for our application.

Compatibility:

We will do our best to ensure our application works on a majority of android and IOS devices.

Reliability and Availability:

Our application will be available all hours of the day; however, if we do need to refresh our database, we would alert our patrons and do it early in the morning (When a majority of the patrons are not cooking).

Maintenance and Manageability:

If we need to restore/reset the database at any time we will have a downtime of 3 hours which will ensure we have enough time. This is based off the average time of restoration of a rails ruby using SQL (15-30 minutes average). Having a downtime greater than the restoration period will also allow us to make minor tweaks that users report to us.

Scalability:

In the first month of deployment, we will monitor the growth of our application and reliability of our database. As we monitor the situation, we will discuss the possibility of increasing our database capacity as well as other options.

Usability:

Throughout the creation of this application, we will have active prototypes that we will test with a closed group of testers. From this closed group of testers, we will make improvements to ensure we make this application as user-friendly as possible. When we are near the completion of the project, we will open this group of testers to a large audience to get feedback on the final prototype before deployment.
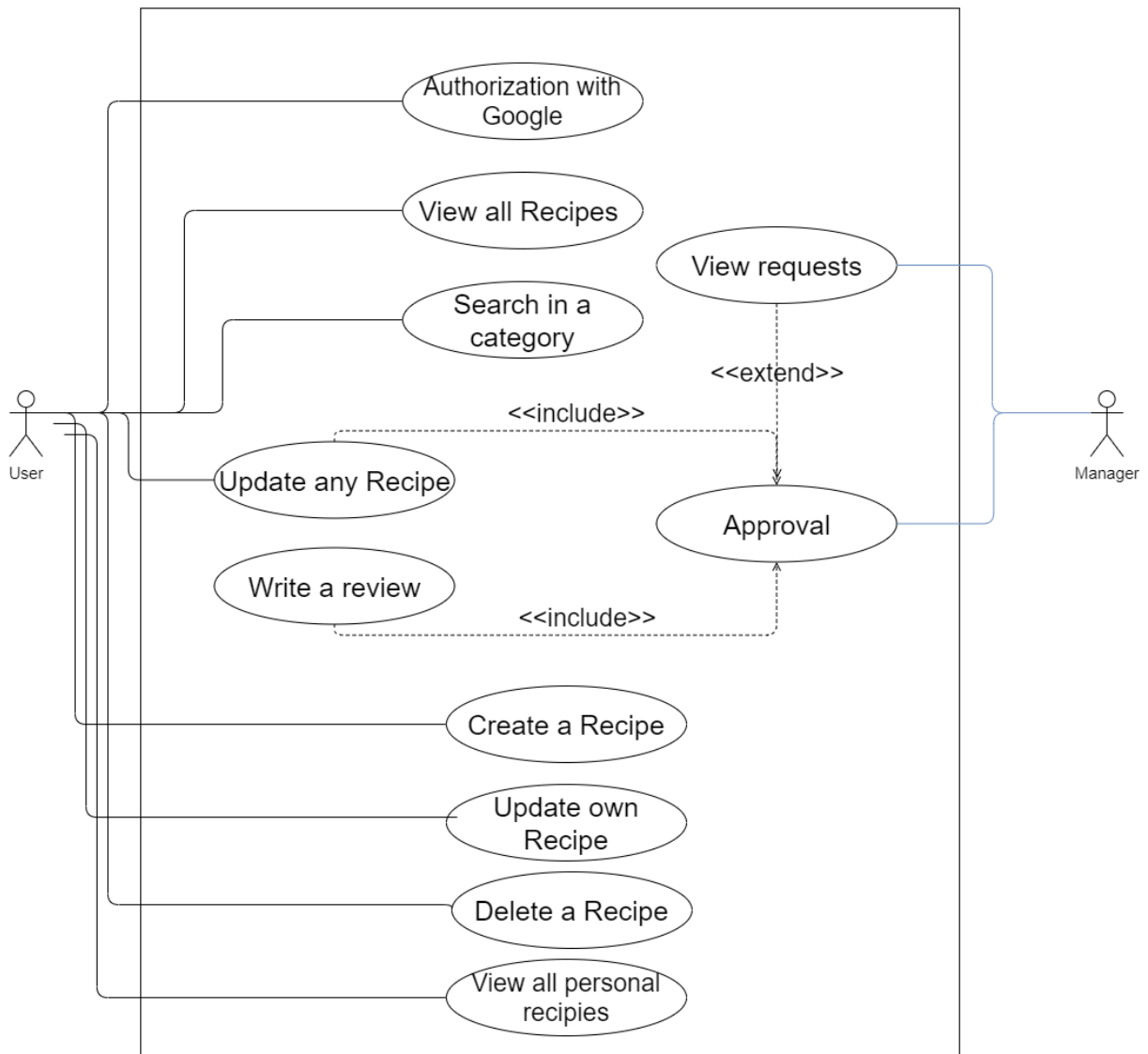
4 FUNCTIONAL REQUIREMENTS

(What the Software Can Do)

4.1 *Use Case 1 (description; use case diagram)*

UML Use Case Diagram

Actors

Use Cases

Actors

Revision History

Version	Date	Name	Description

1	Sep 21, 2021	Ian & Michael	Initial SRD Document
2			