

---

# **Software Requirements Specification**

**for**

# **RecipeBuddy**

**Version 1.2**

**Prepared by Maxwell Barnaby, Alex Cohen, Carlo Desena, Michael  
Horick, Matthew Thompson**

**ACME**

**12/15/2021**

# Table of Contents

<b>1. Introduction</b>	<b>4</b>
1.1 Purpose	4
1.2 Document Conventions	4
1.3 Intended Audience and Reading Suggestions	4
1.4 Project Scope	4
1.5 References	4
<b>2. Overall Description</b>	<b>4</b>
2.1 Product Perspective	4
2.2 Product Features	5
2.3 User Classes and Characteristics	5
2.4 Operating Environment	5
2.5 Design and Implementation Constraints	5
2.6 User Documentation	6
2.7 Assumptions and Dependencies	6
<b>3. System Features</b>	<b>7</b>
3.1 Recipe Creation	7
3.2 Recipe Deletion	7
3.3 Recipe Edit	8
3.4 Recipe Preparation Requirements Validation	8
3.5 Interactive Recipe Execution	8
3.6 Measurement System Toggle	9
3.7 Timers	9
3.8 Ratings and Comments	9
3.9 Recipe Search Bar	10
3.10 Recipe Search Filter	10
3.11 Initial Pantry Audit	10
3.12 Adding and Removing Ingredients	11
3.13 Initial Appliance Audit	11
3.14 Adding and Removing Appliances	11
3.15 Users Pantry Page	12
3.16 Account Privacy	12
3.17 Account Creation	12
3.18 Account Login	13
<b>4. External Interface Requirements</b>	<b>15</b>
4.1 User Interfaces	15
4.2 Hardware Interfaces	17
4.3 Software Interfaces	17

4.4	Communications Interfaces	18
<b>5.</b>	<b>Other Nonfunctional Requirements</b>	<b>19</b>
5.1	Performance Requirements	19
5.2	Safety Requirements	19
5.3	Security Requirements	19
5.4	Software Quality Attributes	19
5.5	Business Rules	19
<b>6.</b>	<b>Key Milestones</b>	<b>20</b>
<b>7.</b>	<b>Key Resource Requirements</b>	<b>21</b>
<b>8.</b>	<b>Other Requirements</b>	<b>22</b>
<b>9.</b>	<b>Requirement Change Management</b>	<b>23</b>
<b>10.</b>	<b>Restrictions, Limitations, and Constraints</b>	<b>24</b>

**Revision History**

\*note: Alex Cohen left the group on 10/25

<b>Name</b>	<b>Date</b>	<b>Reason For Changes</b>	<b>Version</b>
Maxwell Barnaby	10/13	Initial document additions	1.0
Michael Horick	10/14	Touch ups	1.0.1
Michael Horick	10/15	Section 2	1.0.2
Michael Horick	10/16	Section 5	1.0.3
Maxwell Barnaby	10/16	System Features 3.1-3.3	1.0.4
Maxwell Barnaby	10/17	Miscellaneous	1.0.5
Alex Cohen	10/17	System Feature 3.4	1.0.6
All members	10/17	Finish Rough Draft All Sections	1.1
All members	12/15	Final Submission Changes	1.2

# 1. Introduction

## Purpose

*Recipe Buddy 1.0 is a standalone desktop application focused on allowing multiple users to save, create, share and use their own and others recipes. The frontend will be focused on providing an intuitive and user friendly UI. The software will also include an integrated database that directly connects to the rest of the program. This SRS document will provide all encompassing documentation on the totality of the standalone app. That includes all requirements needed for the UI design, user actions, and backend database.*

## Document Conventions

*Indented Arial Font is the standard typeface for this document. Headings are denoted by bolded Times New Roman font. Priorities for higher-level requirements are assumed to be inherited by detailed requirements unless otherwise stated. The idea of recipe creation and recipe making are treated as two specific different actions in this documentation. Recipe creation is the act of generating or inputting a recipe new to the system, into the system. Preparing a recipe is the act of following the instructions of a recipe within the system.*

## Intended Audience and Reading Suggestions

*This document is intended to be used as either a reference for developers/testers or as an in depth guide for a new developer, end-user, or software reviewer. Sections such as revision history, system features and requirements, key resource requirements, and restrictions, limitations, and constraints, can be used as quick references and guides for further development and ensuring proper testing. The rest of the sections, especially the introduction and overall description, will allow those who have not yet become familiar with the project to quickly get up to speed with where the project is currently at, and where it is going in the near and far future. If you are reading this document for the first time, the suggested way of reading is simply in each section's proper numerical order.*

## Project Scope

*RecipeBuddy is a native application that allows a user to make an account for creating, executing, and sharing recipes. It provides a homepage interface and a recipe interface. The homepage interface shows the user different recipes, while the recipe interface helps the user prepare recipes.*

## References

N/A

## 2. Overall Description

### Product Perspective

*RecipeBuddy is a new, self-contained product that is intended to allow the end user to create and manage recipes, ingredients. as well as discover and share recipes. with an advanced database backend for simplicity and speed on the end user's side.*

### Product Features

*RecipeBuddy includes:*

- *Account creation to store kitchen utensils and supplies*
- *Recipe creation and sharing*
- *Recipe searching*
- *Recipe making UI for preparing a meal*
- *Ratings and Comments for recipes*
- *Homepage for instant recommendations and trending page*

### User Classes and Characteristics

*This project is meant to be used by a user class that consists of people who want to:*

- *Organize all of their recipes in one place*
- *Change up their diet*
- *See trending recipes from other users, or try new recipes*
- *Have a simple, easy to use interface for making food for a first time user*

### Operating Environment

*RecipeBuddy is a native Windows GUI application. As such, it will only be tested on Microsoft Windows.*

### Design and Implementation Constraints

*Concerning out of spec. constraints, there are not many. There are no specific requirements given for: regulation, hardware, interfaces with non first party software, use of specific technologies, dev tools, database technologies, system topology, language, protocols. or specific security. nor specific design conventions other than specific UI requests outlined in the specifications.*

*However there are voluntary internal requirements set by the team for the team to create a product that will not be a memory hog, or use more processing time than is needed, and be a good citizen of its platform. as well as generate a code base that is well documented for both internal and external use.*

## **User Documentation**

*User Documentation will be included within the application itself under the “Help” section of the Homepage.*

## **Assumptions and Dependencies**

*RecipeBuddy uses Python as a base, Python 3.7 is recommended.*

## 3. System Features

Features are grouped in order by their similarity. Each feature is also given a priority level 1-5. Each feature is given a volatility level of low, medium or high. Functional requirements are listed in order of general requirement dependency with dependent requirements listed after their dependencies.

### 3.1 Recipe Creation

#### 3.1.1 Description and Priority

*Description:* Accessed on the main page or from edit button within a user's account page, this wizard will allow the user to specify the following for their recipe in this order: A title, a photo, required appliances, required ingredients and their corresponding measurements, each step of their recipe with optional interactive timers, and tags.

*Anticipated Error Correction and Validation:*

- A title and at least one appliance, ingredient, and step will be required for each recipe creation to ensure the recipe at least contains some substance.
- If the recipe is not saved and the creation page is left, it will be discarded or reverted to its last save.

*Priority:* 5

*Volatility:* High

### 3.2 Recipe Deletion

#### 3.2.1 Description and Priority

*Description:* Recipes are deleted from the recipe management area. Once a recipe is deleted it should no longer be shown in the recipe management area or from any search or main page listing. A verification window will be shown to verify the user actually wants to delete the recipe.

*Anticipated Error Correction and Validation:*

- The verification window is important to avoid a misclick and accidental data erasure.
- Once the user decides to delete a recipe it must be fully erased from the system.

*Priority:* 5

*Volatility:* Low



### 3.3 Recipe Edit

#### 3.3.1 Description and Priority

*Description: When a recipe is chosen to be edited the user will simply return to the recipe creation wizard where they last saved the recipe.*

*Anticipated Error Correction and Validation:*

- Deleted or edited recipes must affect the database properly to ensure changes are seen from the user who created the recipe and from other users who access the recipe from the search or home screen.

*Priority: 4*

*Volatility: Low*

### 3.4 Recipe Preparation Requirements Validation

#### 3.4.1 Description and Priority

*Description: Before the recipe can be made, the system must check whether the user has the required ingredients and appliances in their pantry/ appliance list. This will be checked in the background before the user can start the making process. If and only if there is a missing requirement found then the user will be prompted with a special menu showing what they are missing (and if applicable how much) and allow them to instantly add the missing items to their pantry/appliance list.*

*Anticipated Error Correction and Validation:*

- If a discrepancy is found and the user decides that they no longer can or want to prepare the recipe, they should be able to cleanly leave the recipe making process.
- If the user does not choose to add the items to their pantry/appliance list there is no way for them to continue to the interactive recipe making.

*Priority: 3*

*Volatility: Low*

### 3.5 Interactive Recipe Execution

#### 3.5.1 Description and Priority

*Description: Initiated from a listed recipe, the user follows a step by step process in which they indicate when they have finished a step. Steps not yet initiated will be greyed out. When the user finishes the recipe by pressing a button revealed after the final step is initiated, all used ingredients are removed from their pantry.*

*Anticipated Error Correction and Validation:*

- Although there will not be a way to undo the initiation of the next step, previous steps will still be visible and interactable.

Priority: 5  
Volatility: High

## 3.6 Measurement System Toggle

### 3.6.1 Description and Priority

*Description: Whether the recipe was created with metric or imperial measurements, the user when making the recipe can toggle between both systems of measurement which is calculated automatically. This is done simply by a button located next to the ingredients list.*

*Anticipated Error Correction and Validation:*

- *All possible measurement types for each system must be anticipated to allow for the conversions to be calculated*

Priority: 4  
Volatility: Low

## 3.7 Timers

### 3.7.1 Description and Priority

*Description: The recipe creator may choose for each step whether to include a timer. If a step includes a timer it will be displayed below the corresponding step. The timer will be fully interactable by the user where they can start, stop or reset the timer. The timer will display minutes and seconds.*

*Anticipated Error Correction and Validation:*

- *Timers from the current and all previously initiated steps should be fully interactable.*

Priority: 3  
Volatility: Low

## 3.8 Ratings and Comments

### 3.7.1 Description and Priority

*Description: Once a recipe is declared finished by the user, they are given an option to leave a rating and comment. The rating is 1-5 stars and the comment is a small text box allowing the user to type out a review. When a user views a recipe they will be able to see any comments or reviews left by other users. If a user does not wish to leave a rating and comment, they can simply leave the inputs blank to finish the recipe making process.*

*Anticipated Error Correction and Validation:*

- *A user can leave a rating with no comment but not a comment with no rating.*
- *Comments will be limited to a TBD size.*

Priority: 3  
Volatility: Low

### 3.9 Recipe Search Bar

#### 3.9.1 Description and Priority

*Description:* The search bar will be a way to allow users to search through recipes to find what they want. The search bar first checks if what is searched is an ingredient. If it is then it will show all recipes that contain that ingredient. If the search is not an ingredient then it will show results for full or partial title matches. All searches are ordered by most executed.

*Anticipated Error Correction and Validation:*

- This feature will need to effectively search and sort through the database of recipes.
- Multiple words will add to the complexity of the sorting process

Priority: 5  
Volatility: High

### 3.10 Recipe Search Filter

#### 3.10.1 Description and Priority

*Description:* Filters give the user the ability to refine their search more precisely. These filters will include time, ingredients, dietary restrictions, etc.

*Anticipated Error Correction and Validation:*

- This feature will need to effectively search and sort through the database of recipes.
- Multiple filters will add to the complexity of the sorting process

Priority: 2  
Volatility: Medium

### 3.11 Initial Pantry Audit

3.11.1 On account creation, users will be able to manually initialize their pantry. Here the user can select on a drop down menu which ingredients they have. They can then enter in how much of those ingredients that they have at the time of account creation.

*Anticipated Error Correction and Validation:*

- It is possible that users do not add anything to their list on account creation but this is ok because they cannot create any recipes without the requirements and can choose to add to their pantry and appliances later on their account page

Priority: 4  
Volatility: Low

## 3.12 Adding and Removing Ingredients to Pantry

### 3.12.1 Description and Priority

*Description:* A list will be shown to the user with all of the hard coded ingredients listed. Ingredients will be listed in order of most to least amount owned. Along with the ingredients name and amount, the proper unit for that ingredient will be shown. Therefore, when adding or removing ingredients, the user types the amount corresponding to the displayed unit they wish to add or remove. To add or remove the user must first type the amount and then click the “add” or “remove” button. For example, if a user wished to add butter, they could see butter was counted by the tablespoon, type “8” and click add to enter 8 tbsps or 1 cup of butter into their pantry.

*Anticipated Error Correction and Validation:*

- Users will be able to always see the unit of measurement currently being used to each ingredient so they know the proper amount to enter.

*Priority:* 4

*Volatility:* Low

## 3.13 Initial Appliance Audit

*3.13.1 On account creation, users will be able to manually initialize which appliances they own. Here the user can select on a drop down menu which appliances they have.*

*Anticipated Error Correction and Validation:*

- It is possible that users do not add anything to their list on account creation but this is ok because they cannot create any recipes without the requirements and can choose to add to their appliances later on their account page

*Priority:* 4

*Volatility:* Low

## 3.14 Adding and Removing Appliances

### 3.14.1 Description and Priority

*Description:* A list will be shown to the user with all of the hard coded appliances listed. Appliances will be divided with owned appliances shown before unowned, with each section in alphabetical order. Appliances are either owned or unowned.

*Anticipated Error Correction and Validation:*

- Users will be able to always see the unit of measurement currently being used to each ingredient so they know the proper amount to enter.

*Priority:* 5

*Volatility:* Low

### 3.15 Users Pantry and Appliance Page

#### 3.15.1 Description and Priority

*Description: At any time users can add or remove items from their pantry within their account page. The process of which is described within 3.12.*

*Anticipated Error Correction and Validation:*

- *Each measurement type will have a maximum to ensure the user can add too much and cause system instability.*

*Priority: 4*

*Volatility: Medium*

### 3.16 Account Privacy

#### 3.16.1 Description and Priority

*Description: At any time users can toggle between being a private or public account. Any recipes made by a private account cannot be seen by any other account except the creator. The recipe will not come up in any searches. Recipes made by a public account can be seen by anyone, and will show up in the results for searches. If an account is public but changes to private then any recipe previously created will become unable to be seen by anyone. If an account is private but changes to public then any recipe previously created will become viewable by anyone.*

*Priority: 4*

*Volatility: Low*

### 3.17 Account Creation

#### 3.17.1 Description and Priority

*Description: By pressing the Create Account button a user can begin to create their account. The user will enter in a username and password. If the username is the same as someone else's then the user will be prompted to make a different username, as that one is taken.*

*Priority: 5*

*Volatility: Low*

## 3.18 Account Login

### 3.18.1 Description and Priority

*Description: By entering in the username and password of a user at the login screen, the user is able to enter their account and see their recipes. If the user enters in the username or password incorrectly then they are prompted to re enter their credentials as they have incorrectly entered them in the previous attempt.*

*Priority: 5*

*Volatility: Low*

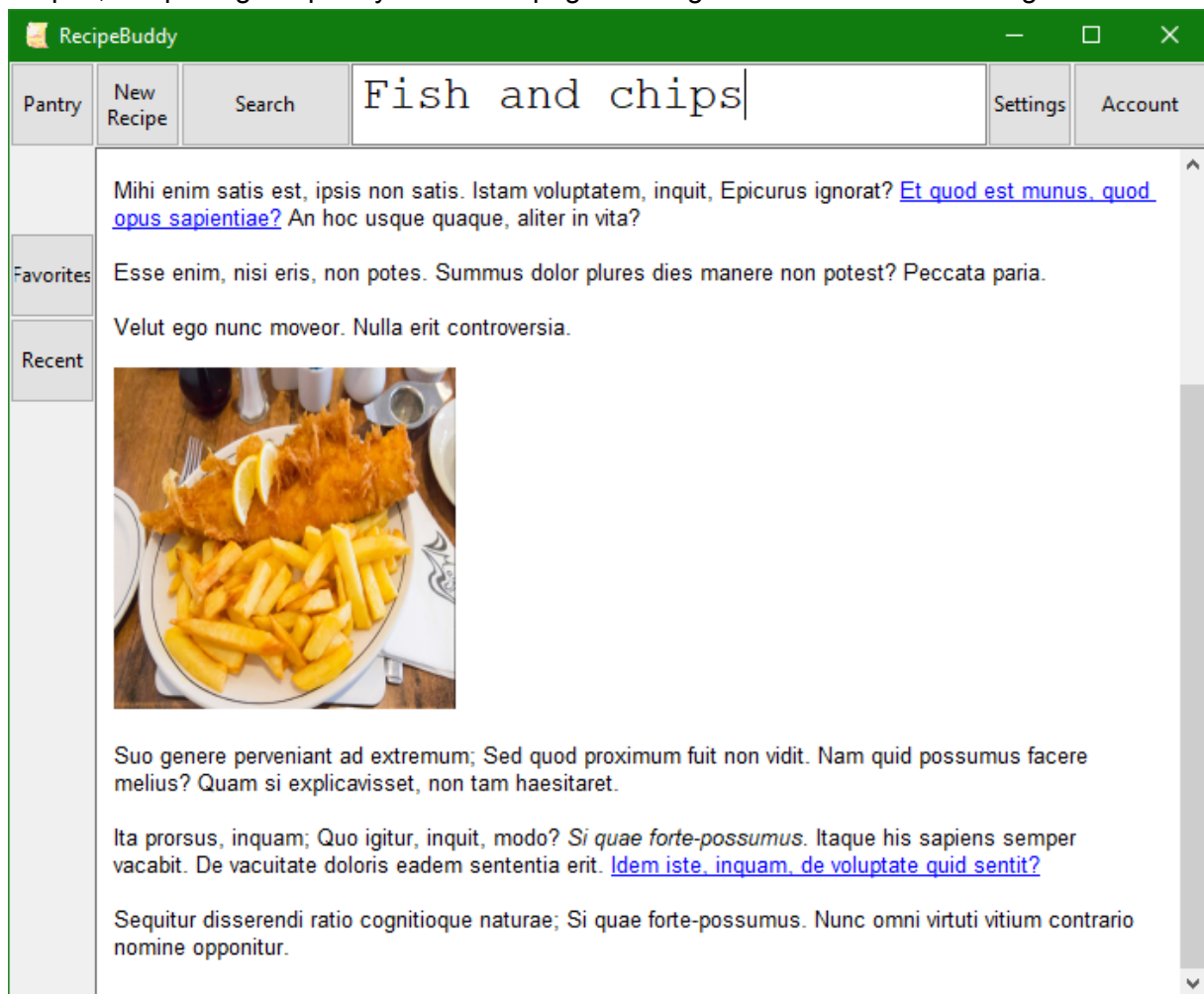
## 4. External Interface Requirements

### User Interfaces

User interfaces are an important design element of the overall system, as they are the main method of interaction for the majority of users. For our set of interfaces, we will try to use system standard styles and layouts where possible, with some common web application technologies to assist familiarity. For all screens, an attempt will be made to not show errors, but when appropriate, an error display will be shown, or given as feedback for an action.

#### Main Menu (homepage)

The main menu, or homepage, is the primary screen that will be seen by the user, it has a large viewing section to view content, either search results, or the main recipe view. It also contains a search bar, and quick access buttons for commonly used items such as opening the favorite recipes, or opening the pantry. The homepage is designed to be clean and straightforward.



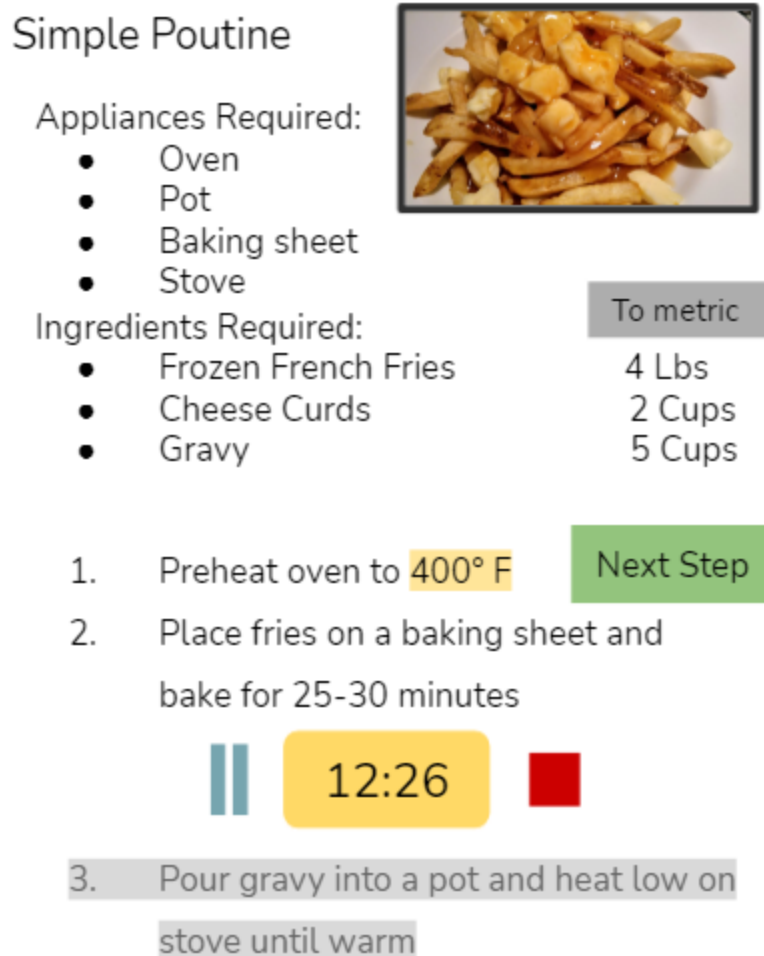
A mockup of the homepage

## Recipe Creation

The Recipe Creation screen will basically be a large form fill, with clearly marked areas for name, short description, ingredients, time to cook, how many people, tools, and then finally a place to put the steps required, as well as a timer creation tool.

## Recipe Execution

For Preparing a Recipe, the user will have a window with instructions, which will walk the user through the steps of the recipe, onscreen will list the tools required for the step, as well as the ingredients, both of which will also be displayed at the beginning of the recipe when it is selected. Additionally if a step calls for a timer, an onscreen item will be displayed for which a timer preset to the required time will be available to press.



A mockup of the recipe interface

## Pantry

The pantry screen will be updated in the background when users select / prepare recipes. When opening the pantry screen users will see a list of ingredients for the recipes they have selected, with information about what recipes that ingredient is required for. critically the pantry screen can generate a shopping list.



## Account / Settings

For the Account screen users will be able to login / logout, and toggle some user settings, such as privacy settings, personal preferences, etc..

## Keyboard shortcuts

some keyboard shortcuts that would ideally be included with the software are:

- Ctrl-S for search
- Ctrl-F (find) also search, might be more context specific,
- Ctrl-C and Ctrl-V, copy and paste, all text copyable and text entry fields are pastable, this is handled by the OS not the UI software though.
- Ctrl-Z (undo) will have different uses depending on context
- tab (+/- shift)(cycle through selected items) used for fast maneuvering through menus, pressing tab or shift tab will highlight a UI element, and pressing enter will interact with it. useful for if you don't want to touch the mouse.

The full range of keyboard shortcuts will be developed when the software is ready for user testing, to determine what would be useful, and what is clunky and would benefit from a new shortcut.

## Hardware Interfaces

N/A

## Software Interfaces

Being written primarily to work on the Microsoft Windows operating system, that is the first requirement, since Windows 11 has only just started its life cycle, and software support is still being determined by many vendors, and Windows 10 is mature in its life cycle. The primary version will be Windows 10.

While other Operating systems are planned as a stretch feature, Windows 10 is the primary OS target for the MVP. If the goals are met the program will have full functionality on Apple MacOS, versions unclear for now. and finally, Linux, specific distros and versions are unclear for now, as they are not a priority or a requirement.

The second software requirement is python, the newest version of python is python 3. Since Python 2 has reached EOL, Python 3 will be the primary development target. The inbuilt python package manager PIP is included in Python 3 as a software package.

The software components are Python UX library wxPython. The newest version is version 4, therefore it is the development target, it is installed via PIP. The wxPython Library uses the operating system's default styles and colors to make windows look like any native application. On the backend there will be a version 8 MySQL database, and code written in C++ version 20.

## **Communications Interfaces**

N/A

## 4. Other Nonfunctional Requirements

### Performance Requirements

*RecipeBuddy is designed to be lightweight and run on a variety of machines. The response time of the program is direct and the application is considered real time.*

### Safety Requirements

*RecipeBuddy should not interact directly with files stored on the user's computer aside from files that RecipeBuddy creates. RecipeBuddy will only interact with files stored on the local database used for storing account information.*

### Security Requirements

*RecipeBuddy does not store any personal information beyond kitchen utensils and what recipes you choose to prepare. However, making a private account will allow a user to utilize RecipeBuddy without fear of anyone being able to see their account activity.*

### Software Quality Attributes

*This application provides a simple and user-friendly interface with simple functionality. Any user should be able to pick up and use RecipeBuddy with nothing more than some basic computer knowledge. Users can simply create an account and prepare recipes with a step by step interface.*

### Business Rules

*Recipe Buddy's feature set will be accessible to all with a user account. Anyone using recipe buddy can simply create an account. The only constraints exist when a user first uses recipe buddy without an account, where they will be considered a guest. Guests can only view and search recipes but cannot prepare or create them. All actions taken as a guest are temporary.*

## 5. Key Milestones

#	Milestone	Target Completion Date	Comments
1.	Database of accounts	10/31/21	
2.	Account front end	11/7/21	
3.	Database of recipes	11/21/21	
4.	Recipe front end	11/28/21	
5.	Main page	12/10/21	

## 7. Key Resource Requirements

[illegible]

## **8. Other Requirements**

*The database of this application will have a schema that will maximize the efficiency of searching for certain elements. The schema will consist of a table for account and recipe. The table for 'account' will have the following attributes: name, password, and number of recipes made. The table for 'recipe' will have the following attributes: ingredients, course, serving size, time needed to prepare the meal, and a difficulty rating.*

## **9. Requirement Change Management**

*Requirements can be simply changed by any request given by the customer, formal or informal. They can be dictated after a review of this document, viewing a prototype or mockup, during implementation of a requirement, or even if the customer changes their mind. Before a change or addition can be established however, the assigned developer or developers working on the feature must discuss and consider the feasibility. Only after the implementation of the change or addition is decided to be feasible, or it is compromised to make it feasible, then it can be fully documented and implemented.*

## **10. Restrictions, Limitations, and Constraints**

- For development purposes the program will be limited to a fully integrated database with no network synchronization for accounts between clients.