



CSC 431

WeCook

Software Requirements Specification (SRS)

Team 04

Avani Choudhary	
Alessandro D'Urso	
Daniel Moran	

Version History

Version	Date	Author(s)	Change Comments
1	2/23/21	Daniel Moran, Alessandro D'Urso, Avani Choudhary	first draft
2	3/9/21	Daniel Moran, Alessandro D'Urso, Avani Choudhary	second draft

Table of Contents

1.	System Requirements	6
1.1	Functional Requirements	6
1.1.1	Sign-up for account	6
1.1.2	Log-in to existing account	6
1.1.3	Account & Settings Modification	6
1.1.4	List Cookbooks	7
1.1.5	View Recipe	7
1.1.6	View Recipe	7
1.1.7	Search & Explore Page	8
1.1.8	Liking/Saving Recipes	8
1.1.8	Commenting functionality	9
1.1.9	Messaging functionality	9
1.1.10	Notification Badges	9
1.1.11	Password Recovery	10
1.1.12	Login/Account Failure	10
1.1.13	Recipe Sharing Failure	10
1.2	Non-Functional Requirements	11
1.2.1	Database input sanitization	11
1.2.2	Message encryption	11
1.2.3	Automatic password reset requests	11
1.2.4	Search result time	11
1.2.5	Uptime	12
1.2.6	Cookbook Storage	12
2.	System Constraints	13
2.1	Tool Constraints	13
2.1.1	Mobile Application Framework Constraint	13
2.2	Language Constraints	13
2.2.1	Backend Framework	13
2.3	Platform Constraints	13
2.3.1	Mobile Application Platform	13
2.4	Hardware Constraints	13
2.4.1	Storage Constraints	13
2.4.2	Computation Constraints	14
2.5	Network Constraints	14
2.5.1	Access Database	14
2.5.2	Uploading	14
2.5.3	Messaging	14
2.6	Deployment Constraints	14
2.6.1	Amazon Lightsail Deployment	14
2.7	Transition & Support Constraints	15
2.7.1	End of Life	15
2.8	Budget & Schedule Constraints	15
2.8.1	Time Constraints	15
2.8.2	Budget Constraints	15

3.	Requirements Modeling	16
3.1.1	System Model	16
4.	Evolutionary Requirements	17
4.1	Functional Requirements	17
4.1.1	Requirement Title	17
4.2	Non-Functional Requirements	17
4.2.1	Requirement Title	17

Table of Tables

<Generate table here>

Table of Figures

<Generate table here>

1. System Requirements

1.1 Functional Requirements

1.1.1 Sign-up for account

Title	Sign-up
Description	Form that allows users to create an account
Priority	0
Precondition(s)	Open up the application after downloading
Basic Flow	User clicks create account Form opens, user enters details including name, username, email and password User submits information and the account is created and saved to database User is directed to login page
Postconditions(s)	User account is created and they are directed to login, can access all features of the app
Use Case Diagram	<Link or number, if present>

1.1.2 Log-in to existing account

Title	Log-in
Description	Existing users must be able to sign into their accounts
Priority	0
Precondition(s)	User should have already signed up for an account
Basic Flow	User enters username/email User enters password User presses "Sign-in" and validation occurs
Postconditions(s)	User is redirected to the home page Repeat if incorrect username/password
Use Case Diagram	<Link or number, if present>

1.1.3 Account & Settings Modification

Title	Account & Settings
Description	A user must be able to edit their account and application settings
Priority	0
Precondition(s)	User should have already signed into the application, and clicked account page button from home page
Basic Flow	User can edit/delete/add: <ul style="list-style-type: none">- Name, Username, Password, Linked Phone #, Friends list, Profile photo User can delete account User can block other users

	User can change dark/light theme User can modify push-notification settings User can view Technical support information
Postconditions(s)	User account information is modified in backend database, user settings are stored locally
Use Case Diagram	<Link or number, if present>

1.1.4 List Cookbooks

Title	Cookbook List
Description	A user must be able to view a list of their cookbooks, and select them interactively
Priority	0
Precondition(s)	User should have signed in to the application and clicked "Cookbooks" icon to be redirected
Basic Flow	User can select a Cookbook User can redirect to public Cookbook search User can join a Private Cookbook
Postconditions(s)	New Recipes added/removed Cookbook settings changed Cookbook availability changed
Use Case Diagram	<Link or number, if present>

1.1.5 View Cookbook

Title	Cookbook View
Description	A user must be able to individually view public or private cookbooks they are members of
Priority	3
Precondition(s)	User should have already signed in to the application and selected a Cookbook from the Cookbook List
Basic Flow	User can add/delete/edit recipe If user owns Cookbook, they can change public/private If user owns Cookbook, they can invite users to Cookbook If user owns Cookbook, they can change permissions User can add/remove members from a private cookbook
Postconditions(s)	New Recipes added/removed Cookbook settings changed Cookbook availability changed
Use Case Diagram	<Link or number, if present>

1.1.6 View Recipe

Title	Recipe View
Description	A user must be able to view an individual recipe using the recipe icon in their cookbook next to a particular entry
Priority	1
Precondition(s)	User should have already signed in to the application and selected a recipe from a public or private Cookbook
Basic Flow	User can create/delete recipe

	User can add ingredients User can add instructions User can add comments to other users' recipes User can add a photo
Postconditions(s)	New Recipes added/removed Recipe modified Recipe published and saved to database
Use Case Diagram	<Link or number, if present>

1.1.7 Search & Explore

Title	Search & Explore Page
Description	A user must be able to search for their recipes, as well as view recipes similar to those they already like or have viewed recently
Priority	2
Precondition(s)	Sign into application, navigate to search & explore page from home page
Basic Flow	User can either search or explore For search: <ul style="list-style-type: none"> - User clicks on search bar - Enters a few words based on what recipe/cookbook/user they are looking for - Application provides results based on what user enters/public Cookbook availability For explore: <ul style="list-style-type: none"> - User clicks on one of the categories provided by the page, including types of cuisines and meals - Given the category they click on, they are provided with a feed of recipes and Cookbooks saved under that category
Postconditions(s)	Recipes saved in search are recorded by database, search history is also saved by database
Use Case Diagram	<Link or number, if present>

1.1.8 Liking/Saving Recipes

Title	Like/Save
Description	A user must be able to like or save a recipe
Priority	4
Precondition(s)	Sign into application, navigate to recipe
Basic Flow	Liking a Recipe: <ul style="list-style-type: none"> - User selects like icon underneath recipe - Creator is notified that their recipe has been liked - Liked recipe is added to database, so user can locate it later - User may unlike a recipe Saving a Recipe <ul style="list-style-type: none"> - User selects save icon below the recipe

	<ul style="list-style-type: none"> - Saved recipe is added to the User's "Saved Recipes" cookbook - User may unsave a recipe
Postconditions(s)	Recipes are added to "saved recipes" cookbook, or to the database for retrieval by user if liked
Use Case Diagram	<Link or number, if present>

1.1.9 Commenting Functionality

Title	Comment
Description	A user must be able to comment on recipes
Priority	5
Precondition(s)	Sign into application, navigate to recipe
Basic Flow	<ul style="list-style-type: none"> - User selects "add a comment" - User writes comment - Creator of recipe is notified - Creator, or other users may reply to individual comments - Commenting may be turned off by the creator
Postconditions(s)	Comment is added to the recipe
Use Case Diagram	<Link or number, if present>

1.1.10 Messaging Functionality

Title	Messaging
Description	A user must be able to send private messages to another user
Priority	4
Precondition(s)	Select recipe, or "friend" and select messaging icon
Basic Flow	<ul style="list-style-type: none"> - User selects messaging icon - Message is sent/delivered to recipient - Recipient is notified - Recipient may reply - Recipients, if not already friends, may reject a messaging request
Postconditions(s)	Message thread is created or updated
Use Case Diagram	<Link or number, if present>

1.1.11 Notification Badges

Title	Notifications
Description	A user must be notified when their recipes are liked, cookbooks are followed, or comments/messages are received
Priority	4
Precondition(s)	User must have an account
Basic Flow	<ul style="list-style-type: none"> - One of above actions (like, message, comment, friend request) occurs - User is notified - User may respond appropriately
Postconditions(s)	User is notified
Use Case Diagram	<Link or number, if present>

1.1.12 Password Recovery

Title	Password Recovery
Description	A user must be able to recover their password, should the forget it or have their account compromised (who hacks a cookbook app? Seriously.)
Priority	0
Precondition(s)	User must have an account
Basic Flow	<ul style="list-style-type: none">- User has forgotten password, or account has been locked for security reasons- User submits a request for a password reset- User receives email with password reset instructions- User is redirected to an in-app password reset tool from unique link
Postconditions(s)	User password is modified, password is updated in database
Use Case Diagram	<Link or number, if present>

1.1.13 Login/Account Creation Failure Notification

Title	Account Failure
Description	A user must be notified when account creation or login has failed
Priority	1
Precondition(s)	User must be at account creation or user login page
Basic Flow	<ul style="list-style-type: none">- User attempts to create an account<ul style="list-style-type: none">- User is notified if password is too weak- User is notified if username/email is already in use- User attempts to login to an existing account<ul style="list-style-type: none">- User is notified if password/username combination is incorrect- After 10 failed attempts, user password is reset (per 1.1.12)- User is redirected to account creation/login page depending on precondition
Postconditions(s)	User is redirected to repeat process, or eventually, reset password
Use Case Diagram	<Link or number, if present>

1.1.14 Recipe Sharing Failure

Title	Recipe sharing Failure
Description	A user must be notified when recipe sharing has failed
Priority	5
Precondition(s)	User must select "share recipe" on a non-public recipe

Basic Flow	<ul style="list-style-type: none"> - User attempts to share nonpublic recipe on the platform <ul style="list-style-type: none"> - User is notified immediately that sharing has failed with a splash message - User attempts to view a shared recipe from outside the platform <ul style="list-style-type: none"> - User is notified that they do not have permission to view the recipe - User may request to join the cookbook (per 1.1.4)
Postconditions(s)	User is redirected to repeat process, or eventually, reset password
Use Case Diagram	<Link or number, if present>

1.2 Non-Functional Requirements

1.1.1 Database input sanitization

Title	Database input sanitization
Description	Any time inputs are entered into database, they must be sanitized to prevent errors and attacks
Priority	0
Applicable FR(s)	1.1.1(,3,5,6,7,8)

1.1.2 Message encryption

Title	Message encryption
Description	Messages between users must be encrypted. This will prevent bad actors from intercepting messages
Priority	3
Applicable FR(s)	1.1.10

1.1.3 Automatic password reset requests

Title	Automatic password reset requests
Description	Emails for password resets must be sent automatically
Priority	0
Applicable FR(s)	1.1.2, 1.1.12

1.1.4 Search result time

Title	Search result time
Description	Search results must be returned in less than 500ms
Priority	0
Applicable FR(s)	1.1.2, 1.1.12

1.1.5 Uptime

Title	Uptime
Description	The application must be down for no longer than 15 minutes each month
Priority	3
Applicable FR(s)	N/A

1.1.6 Cookbook Storage

Title	Cookbook Storage
Description	Database must be able to support dynamic storage resource allocation to adequately store the users' data
Priority	0
Applicable FR(s)	N/A

System Constraints

61.2 Tool Constraints

< List all tool constraints in the following example format >

1.2.1 Mobile Application Framework Constraint

Title	Mobile Application Framework Constraint
Description	We will be using Apache Cordova as the framework for the application. This allows for the creator to wrap a "web application" into a mobile app
Priority	Mandatory: 0

61.3 Language Constraints

< List all language constraints in the following example format >

1.3.1 Backend Framework

Title	Backend Framework
Description	As we are using Apache Cordova, HTML5, CSS3 and JavaScript are a requirement. The entire backend framework will be written in these languages
Priority	Mandatory: 0

61.4 Platform Constraints

< List all platform constraints in the following example format >

1.4.1 Mobile Application Platform

Title	Mobile Application Platform
Description	Apache Cordova is platform independent and allows for cross-platform mobile application creation.
Priority	Lowest: 5

61.5 Hardware Constraints

< List all hardware constraints in the following example format >

1.5.1 Storage Constraints

Title	Storage Constraints
Description	Will be set by Amazon Lightsail, but user information will be saved in the database, so this is of minimal importance.
Priority	Lowest: 5

1.5.2 Computation Constraints

Title	Computation Constraints
Description	Set by Amazon Lightsail, sufficient for the purposes of this version of the application and for this course. Simple computations are necessary for this application, such as the ranking of search results.
Priority	Medium: 3

Network Constraints

< List all network constraints in the following example format >

1.5.3 Access Database

Title	Access Database
Description	Must have access to and query the MySQL database in order to get account, user, and recipe information.
Priority	Mandatory: 0

1.5.4 Uploading

Title	Uploading
Description	Users must be able to upload and share recipes to their accounts, which must be saved to the MySQL database
Priority	Mandatory: 0

1.5.5 Messaging

Title	Messaging
Description	Users must be able to send and receive messages, and share posts
Priority	Mandatory: 0

61.6 Deployment Constraints

< List all deployment constraints in the following example format >

1.6.1 Amazon Lightsail Deployment

Title	Amazon Lightsail Deployment
Description	This mobile application will be deployed by Lightsail. Lightsail provides a free month of usage which will be sufficient for the purposes of this course.
Priority	Medium: 3

61.7 Transition & Support Constraints

< List all transition & support constraints in the following example format >

1.7.1 End of Life

Title	End of Life
Description	As this project will end at the termination of this course in May, another team will have to take over in order to develop the mobile application.
Priority	Lowest: 5

61.8 Budget & Schedule Constraints

< List all budget & schedule constraints in the following example format >

1.8.1 Time Constraints

Title	Time Constraints
Description	Application must be completely designed before the end of the semester (May 2021). A final project packet with a Requirements Document and Design Document must be completed by then.
Priority	Mandatory: 0

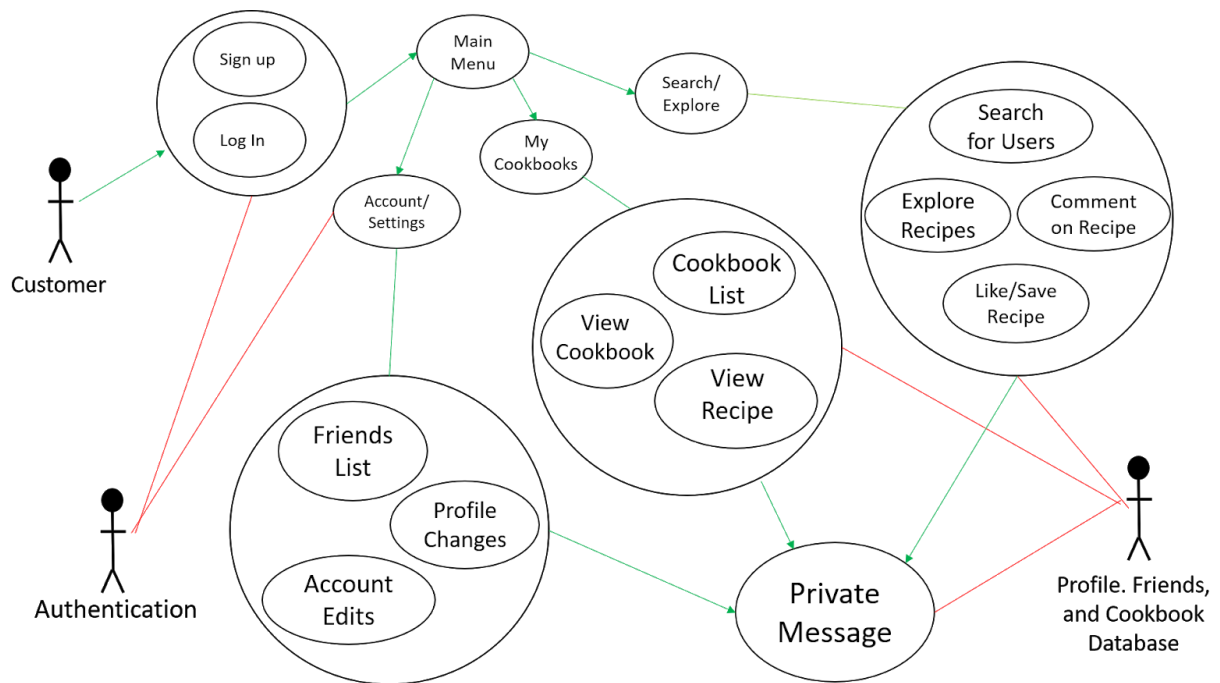
1.8.2 Budget Constraints

Title	Budget Constraints
Description	There are no funds available for this project, therefore there is no budget.
Priority	Lowest: 5

62. Requirements Modeling

< List all Use-case diagrams for the functional requirements in the following format >

2.1.1 System Model



63. Evolutionary Requirements

63.1 Functional Requirements

< List all functional requirements in the following example format >

3.1.1 Requirement Title

Title	<Insert title>
Description	<A one or two sentence description>
Priority	<Priority from 0 (highest) – 5 (lowest)>
Precondition(s)	<What needs to happen before>
Postconditions(s)	<What happens as a result>
Use Case Diagram	<Link or number, if present>

63.2 Non-Functional Requirements

< List all non-functional requirements in the following example format >

3.2.1 Requirement Title

Title	<Insert title>
Description	<A one or two sentence description>
Priority	<Priority from 0 (highest) – 5 (lowest)>
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>