System Test Plan

for

Glucose Genie

Version – 3.0

June 4, 2025

Prepared by

Francisco Cruz-Urbanc

Krisi Hristova

Carson Ford

Jared Jackson

Thomas Capro

CI 493 – Senior Project

Filippos Vokolos

Table of Contents

1 Introduction

1.1 Purpose

This System Test Plan document defines the testing objectives, resources, testing plans, approaches, schedule, exit criteria and associated risks for the Glucose Genie software application. It ensures that the system is thoroughly evaluated and meets its intended requirements before release.

1.2 Definitions and Acronyms

- API – Application Programming Interface
- AWS – Amazon Web Services
- UI – User Interface
- SRS – System Requirements Specification

1.3 Scope of Testing

1.3.1 In Scope

The scope of this System Test Plan includes functional and nonfunctional testing of both the UI and backend/API components of the Glucose Genie mobile application. The focus will be on verifying functionality and integration of the following features:

- Account creation
- Recipe search
- Individual recipe display
- Weekly meal plan creation
- Nutrient tracker
- Language switching (English and Spanish)
- Grocery store list generation and display
- Grocery store map locator based on user location

The UI of these features will be tested through functional and integration testing. The API and backend implementations of these features will be verified through functional and integration testing.

1.3.2 Out of Scope

The System Test Plan considers unit testing out of scope and thus will not be documented. Additionally, the team will **not** test the following features due to time constraints and lower priorities in regard to the Glucose Genie SRS:

- Notification options
- Grocery store searching based on price ranges

1.4 Intended Audience

This document is intended for use by developers, QA testers, project managers and stakeholders involved with the creation and release of the Glucose Genie application. Developers will use it as reference to how their code will be tested, QA testers will use it to design and implement tests for the system, project managers will use it to ensure the testing plans are executed as per schedule and finally stakeholders will use it to confirm that features are being tested in accordance with their requirements and get insight into the readiness of the system for deployment.

2 References

- SRS - https://github.com/fjcu256/glucose-genie/tree/main/docs/Requirements/SoftwareRequirementsSpec

3 Testing Approach

3.1 UI Testing

The UI testing will fall under two categories: functional testing and integration (non-functional) testing.

3.1.1 Functional Testing

The functional tests for our UI will include two main scenarios: if the correct information is displayed properly on each view component and if the expected result is displayed when the user interacts with the component.

In order to separate integration concerns from the functional features, we will use test data, which will include a set of mock recipes and nutritional facts. This set of test data will be stored locally on each of our machines and integrated through our version control system. This will ensure that all functional tests of the UI will focus solely on how data is displayed and will not rely on any external data sources or APIs.

Since most of our UI displays lists of continuous data, we will create two categories of test data for functional testing: no data, and more than one unit of data. The set of no data will include a test where there is no data given to our UI. This will test our expectations of how the UI should look in the event that no data is given. The other set of test data for UI functional testing will include a list of data that has more than one unit of data. This will test the most common condition of having more than one item of data displayed on the different UI pages. For our purposes, a unit of test data means a single recipe; this includes a name, image, description, nutritional data, etc. So, each set of test data should be able to be used for multiple different functional tests to be most efficient.

3.1.2 Integration Testing

The integration tests for our UI will evaluate the interactions between the user and the frontend that result in database changes in the backend. Some of these tests will be able to be performed on the development version of the application; for example, testing that when a user creates an account it is stored in the AWS database. Most of the tests, however, will have to be performed on the production version of the application because they require API calls and responses. These will be manually tested and checked on our backend database.

3.2 API Testing

The API testing will fall under two categories: functional testing and integration (non-functional) testing.

3.2.1 Functional Testing

The functional tests for the APIs will include testing all valid responses as well as expected error responses from our APIs. This will ensure that queries and input data that is requested from the APIs is in the correct format.

We will create mock queries to the APIs to test that they return either the correct format of data, or an error if that is what should be expected. We are making the assumption that the format in which the data is returned in the response does not change. If this were to change, we would need to make edits to our unit tests and to the application itself.

3.2.2 Integration Testing

There will be no formal API integration tests performed for our application that test that the APIs work correctly. Since the entire application, and thus our other integration and unit tests, depend on the functionality of the APIs, the performance of the APIs will be validated through the other integration tests that are performed successfully. So, all of our integration tests regarding the APIs will be testing the performance, security, and scalability of the of the application. For example, this will include timing the speed of an API request starting from an interaction with the UI to when the correct information is displayed on the screen.

4 Testing Environment

The Glucose Genie application will be tested manually by running a simulator (from Xcode itself or on a device). Time permitted the application will also be tested locally using several integrated parts. The tools are described below. We will use XCTest as the core testing framework to write

and run both unit and UI tests. Glucose Genie is built in test configuration and runs inside the iOS simulator during testing. For API testing, we will inject API clients instead of making real server requests. For UI testing we will use XCUITest along with the possibility of ViewInspector for more specific testing. Our app is configured to use temporary memory in the simulator so that no real user data is affected.

5 Testing Tools

- XCTest
    - Apple's first-party framework for testing. Contains sub-frameworks for specific testing such as mocking APIs.

- XCUITest
    - Sub-framework of XCTest for specifically UI automation tests. This will require no changes in the application.

- ViewInspector
    - Third-party package that works with XCUITest that allows us to inspect the runtime state of views and perform actions like pressing buttons.

6 Test Cases

Below is a listing of all of the test cases associated with the Glucose Genie system in the scope of this document. Each test case has an associated Test ID, Test Name, Description, Pre-Conditions, list of Steps to Take, Post-Conditions, Test Status, a test Creator and Tester. After initial testing failures and their resolutions will be documented in a Defect Report. Each test case also has an assigned date for which the test must be completed by and in be in a SUCCESSFUL test state by. For reference a test status key is also provided below.

Test Status Key:

UNTESTED	The test has not yet been executed.
IN_PROGRESS	Testing is in progress.
SUCCESSFUL	All steps and post-conditions have been met successfully as described in the test case.
FAILED	The test case has not passed. The issue must be documented with a description of which step failed in the test case and what the results are. Bug fixing and retesting is necessary.

7.1 UI Test Cases

Tests for Account Creation

Test ID	GLUC-004-001	**Test Name**	User prompted to Login or Create Account
Description	Opening the Glucose Genie application prompts the user to either login or create an account.		
Pre-Conditions	User is connected to the internet and on the home page of their device.		
Steps	1. Click the “Glucose Genie” application from the home screen of the device. 2. Observe the application open.		
Post-Conditions	User should be prompted to login to AWS or to make an account.		
Test Status	SUCCESSFUL		
Creator	Krisi Hristova	**Date Created**	4/27/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/16/2025

Tests for Main Page

Test ID	GLUC-011-001	**Test Name**	Main Page Navigation Buttons
Description	Verify that each button on the Main Page navigates correctly to the corresponding page and can navigate back to the Main Page.		
Pre-Conditions	-User is logged into their account. -User is on the Main Page.		
Steps	1. Ensure that all of the following buttons are clearly visible on the Main Page: - “Recipes” - “Saved Recipes” - “Weekly Meal Plan” - “Grocery List” - “Find Grocery Store” - “Track Nutrients” - Settings gear icon 2. For each of the buttons: 2.1 Click the button. 2.2 Verify that the button opens the corresponding page. 2.3 Navigate back to the Main Page using the backwards arrow at the top of the screen.		
Post-Conditions	Each button from the main page navigates to the appropriate page and back.		

Test Status	SUCCESSFUL		
Creator	Krisi Hristova	**Date Created**	4/27/2025
Tester	Krisi Hristova	**Date to Test by**	5/19/2025

Tests for Recipe Search

Test ID	GLUC-013-001	**Test Name**	Initial Recipe Search Display with Data
Description	Opening the recipe search page displays a scrollable list of all recipes to the user's screen. Each recipe will be marked in a box with the following information: -Recipe Name -Recipe Image if it exists, otherwise a placeholder image -Calories -Carbohydrates		
Pre-Conditions	-User is connected to the internet. -User is logged in to their account. -User is on the Main Page		
Steps	1. Click "Recipe Search" button on the Main Page. 2. Observe the Recipe Search page open and load all recipes. 3. Scroll to the bottom of the screen to see loaded recipes ensuring each recipe has the 4 required pieces of information.		
Post-Conditions	Recipe Search displays a scrollable list of unfiltered recipes. Each recipe box should display the recipe name, image/placeholder, calorie and carbohydrate information.		
Test Status	SUCCESSFUL		
Creator	Krisi Hristova	**Date Created**	4/27/2025
Tester	Jared Jackson	**Date to Test by**	5/17/2025

Test ID	GLUC-013-002	**Test Name**	Initial Recipe Search Display without Data
Description	Opening the recipe search page displays a blank page with a message that there are no recipes to display.		
Pre-Conditions	-User is on the Main Page		
Steps	1. Click "Recipe Search" button on the Main Page. 2. Observe the Recipe Search page open and present a message that there are "No recipes available" to load.		
Post-Conditions	Recipe Search displays no recipes to the screen and a message that there are no recipes.		
Test Status	SUCCESSFUL		
Creator	Krisi Hristova	**Date Created**	4/27/2025
Tester	Jared Jackson	**Date to Test by**	05/25/2025

Test ID	GLUC-013-003	**Test Name**	Click Recipe Box to View Individual Recipe Page	
Description		Clicking on a recipe opens that particular recipe's individual recipe display page.		
Pre-Conditions		-User is connected to the internet. -User is logged into their account. -User is on the Recipe Search Page.		
Steps		1. Locate a random recipe from the recipe search page. 2. Click on the recipe image. 3. Observe that the selected recipe's individual display page opens.		
Post-Conditions		Individual recipe display page is opened for the selected recipe.		
Test Status		SUCCESSFUL		
Creator		Krisi Hristova	**Date Created**	4/27/2025
Tester		Francisco Cruz-Urbanc	**Date to Test by**	5/20/2025

Test ID	GLUC-013-004	**Test Name**	Recipe Search by Keyword	
Description		Searching for a recipe using keywords filters the display to recipes correctly.		
Pre-Conditions		-User is connected to the internet. -User is logged into their account. -User is on the Recipe Search Page.		
Steps		1. Click on the search bar and type in a keyword (ex. "chicken", "salad", etc). 2. Click search. 3. Wait until the new recipes are loaded and observe the results. 4. Ensure that each recipe that is displayed after the search has some affiliation with the entered keyword. If it is not clear based on recipe name and image, click the recipe to see its detailed view and extra information. 5. Repeat steps 1-4 with different keywords at least 3 times.		
Post-Conditions		Only recipes matching the search keyword are displayed.		
Test Status		SUCCESSFUL		
Creator		Krisi Hristova	**Date Created**	4/27/2025
Tester		Francisco Cruz-Urbanc	**Date to Test by**	5/17/2025

Tests for Recipe Individual Display

Test ID	GLUC-003-001	**Test Name**	Display Recipe Information Correctly
Description	When a user clicks on a recipe from the search results, the individual recipe page should display the recipe's full information.		
Pre-Conditions	User is logged in and has app open on search page with available recipes.		
Steps	1. Search for a recipe 2. Tap on a recipe image from results 3. Observe the individual recipe page that loads.		
Post-Conditions	The page displays recipe name, image (or placeholder), meal type, ingredients list, nutritional information and a link to the full recipe online.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/15/2025

Test ID	GLUC-003-002	**Test Name**	Recipe Page Navigation
Description	User can navigate back to the search results page after viewing an individual recipe page.		
Pre-Conditions	User is on the individual recipe page after selecting a recipe.		
Steps	1. Click the back button on the recipe page.		
Post-Conditions	User is returned to the search results page without app crash or loss of search results.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/27/2025

Test ID	GLUC-003-003	**Test Name**	UI Responsiveness of Recipe Page
Description	Verify that the individual recipe page displays properly on different device sizes (e.g., iPhone SE, iPhone 14 Pro Max).		
Pre-Conditions	App running in iOS Simulator with different device profiles.		
Steps	1. Open the individual recipe page on a small device. 2. Verify no overlapping/cut-off elements. 3. Open and repeat the same page on a large device.		
Post-Conditions	All elements are visible, readable, and not distorted across devices.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/27/2025

Test ID	GLUC-003-004	**Test Name**	Scroll Through Recipe Page
Description	Users can scroll through a long recipe without glitches.		
Pre-Conditions	Recipe page is open with a long recipe.		
Steps	1. Open a recipe with long instructions or many ingredients. 2. Scroll from top to bottom.		
Post-Conditions	All content scrolls smoothly and is accessible.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/15/2025

Test for Recipe Saving

Test ID	GLUC-010-006	**Test Name**	Tap Save Button on Recipe Page
Description	Verify that tapping the “Save” button on a recipe triggers the save action visually.		
Pre-Conditions	User is logged in and viewing an individual recipe page.		
Steps	1. Tap on a recipe from the search page. 2. Tap the “Save” button. 3. Look for any immediate UI change		
Post-Conditions	UI reflects that the save action has been triggered (before backend confirmation).		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/17/2025

Tests for Weekly Meal Planner Page

Test ID	GLUC-014-001	**Test Name**	Weekly Meal Planner Dates
Description	When a user opens the weekly meal planner page, the dates on the page should match the current week.		
Pre-Conditions	User is logged in and has the app open on the home screen		
Steps	1. The user clicks on the “Weekly Meal Planner” button to open the weekly meal planner 2. When the page opens, the current week’s month and dates should appear along with their corresponding days of the week		
Post-Conditions	-		
Test Status	SUCCESSFUL		

Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Thomas Capro	**Date to Test by**	05/27/2025

Test ID	GLUC-014-002	**Test Name**	Weekly Meal Planner Current Date
Description	When a user opens the weekly meal planner page, the current date should be highlighted differently compared to the other days of the week (currently blue font and sun icon).		
Pre-Conditions	User is logged in and has the app open on the home screen		
Steps	1. The user clicks on the "Weekly Meal Planner" button to open the weekly meal planner 2. When the page opens, the current week's month and dates should appear along with their corresponding days of the week 3. When the user scrolls to the current date, the date's section title should be highlighted		
Post-Conditions	-		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/25/2025

Test ID	GLUC-014-003	**Test Name**	Weekly Meal Planner Add Meals
Description	When a user has the weekly meal planner page open, they should be able to add meal recipes to the planner. They should be able to add recipes to any day of the current week for any meal time.		
Pre-Conditions	User is logged in and has the app open on weekly meal planner page		
Steps	1. The user clicks on an empty meal time for one of the days of the week 2. The user then goes through the process of selecting a recipe to add to the meal planner 3. After the user selects a recipe, it should appear on the weekly meal planner page at the desired date and meal time		
Post-Conditions	The selected meal recipe is visible on the weekly meal planner page in the desired date and time slot		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Krisi Hristova	**Date to Test by**	05/14/2025

Test ID	GLUC-014-004	**Test Name**	Weekly Meal Planner Remove Meals
Description	When a user has the weekly meal planner page open, they should be able to remove meal recipes from the planner. They should be able to remove recipes from any day of the current week for any meal time.		
Pre-Conditions	User is logged in and has the app open on weekly meal planner page		
Steps	1. The user clicks and holds on an occupied meal time for one of the days of the week 2. A menu should pop up giving the user the option to remove the meal recipe from their planner 3. After the user selects the remove recipe option, the recipe should be removed from the weekly meal planner page		
Post-Conditions	The selected meal recipe is no longer on the weekly meal planner page at the removed date and time slot		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Krisi Hristova	**Date to Test by**	05/14/2025

Test ID	GLUC-014-005	**Test Name**	Weekly Meal Planner Saves and Loads Plan
Description	When a user makes a change to their weekly meal plan, the plan should save automatically to their device and / or a database.		
Pre-Conditions	User is logged in and has the app open on weekly meal planner page		
Steps	1. The user makes one or more changes on the weekly meal planner page 2. The user then goes to the home page of the app 3. From there, the user goes back into the weekly meal planner page 4. The user then makes one or more changes again to their meal plan 5. Afterwards, the user force closes the app and opens the app again 6. The user then navigates back to the weekly meal planner page		
Post-Conditions	The weekly meal planner page should reflect the user's recent changes to their meal plan		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Carson Ford	**Date to Test by**	05/16/2025

Test ID	GLUC-014-006	**Test Name**	Weekly Meal Planner Resets After New Week
Description	When a week passes, the user's meal plan should reset to an empty meal plan.		
Pre-Conditions	User is logged in and has the app open on home page		
Steps	1. The user clicks on the weekly meal planner button 2. When the page loads, the meal plan should be empty and the days of the week should reflect the current week		
Post-Conditions	-		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/19/2025

Tests for Grocery List Page

Test ID	GLUC-009-001	**Test Name**	Grocery List Displays Message When Empty
Description	When a user opens the grocery list page without any recipes in their meal plan, the grocery list should display an "empty" message		
Pre-Conditions	User is logged in and has the app open on home page		
Steps	1. The user clicks on the grocery list button 2. When the page loads, the grocery list should have no items in it. It should inform the user that they have no recipes in their meal plan		
Post-Conditions	-		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Carson Ford	**Date to Test by**	05/23/2025

Test ID	GLUC-009-002	**Test Name**	Grocery List Displays Grocery Items
Description	When a user opens the grocery list page, it should display a list of grocery items needed to make the recipes in the user's meal plan		
Pre-Conditions	User is logged in and has the app open on home page		
Steps	1. The user clicks on the grocery list button 2. When the page loads, the grocery list should have items in it. The items should reflect the ingredients needed to make the recipes saved in the user's meal plan		
Post-Conditions	-		
Test Status	SUCCESSFUL		

Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Thomas Capro	**Date to Test by**	05/23/2025

Test ID	GLUC-009-003	**Test Name**	Grocery List Updates Grocery Items
Description	When a user opens the grocery list page, it should display a list of grocery items needed to make the recipes in the user's meal plan. If the user then makes a change to their meal plan and opens the grocery list page, the grocery list should be updated to reflect the current meal plan.		
Pre-Conditions	User is logged in and has the app open on home page		
Steps	1. The user clicks on the grocery list button 2. When the page loads, the grocery list should have items in it. The items should reflect the ingredients needed to make the recipes saved in the user's meal plan 3. The user then navigates back to the weekly meal planner page 4. The user makes one or more changes to their meal plan 5. Afterwards, the user navigates back to the grocery list page		
Post-Conditions	The grocery list page should be updated to reflect the user's current meal plan		
Test Status	SUCCESSFUL		
Creator	Jared Jackson	**Date Created**	04/26/2025
Tester	Thomas Capro	**Date to Test by**	05/18/2025

Tests for Grocery Store Map page:

Test ID	GLUC-006-001	**Test Name**	User location is seen on Grocery Store Map Page
Description	The Grocery Store Map page should load and display the user's location correctly.		
Pre-Conditions	User is logged in and is on home screen. Location services should already be allowed by user.		
Steps	1. Click Grocery Store Map page. 2. User Annotation (blue dot) is seen on map.		
Post-Conditions	User annotation corresponds with the user's location		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/25/2025

Test ID	GLUC-006-002	**Test Name**	Location services window appears
Description	The Grocery Store Map page asks for the user's location on first appearance		
Pre-Conditions	User is logged in and is on home screen. Location services permission should be unset		
Steps	1. Click Grocery Store Map page.		
Post-Conditions	Apple alert window should pop up saying "Glucose Genie want to access your location to show nearby grocery stores".		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/24/2025

Test ID	GLUC-006-003	**Test Name**	Location Services access – allow once
Description	The Grocery Store Map page should refresh with user's location after giving access to location services with allow once button		
Pre-Conditions	User is logged in and is on home screen. Location services permission should be unset		
Steps	1. Click Grocery Store Map page. 2. When location services window appears, press "Allow once"		
Post-Conditions	Apple alert window should disappear and map should refresh to show user's location.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Jared Jackson	**Date to Test by**	05/25/2025

Test ID	GLUC-006-004	**Test Name**	Location Services access – allow while using app
Description	The Grocery Store Map page should refresh with user's location after giving access to location services with allow while using app button		
Pre-Conditions	User is logged in and is on home screen. Location services permission should be unset		
Steps	3. Click Grocery Store Map page. 4. When location services window appears, press "Allow while using app"		
Post-Conditions	Apple alert window should disappear and map should refresh to show user's location.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025

Tester	Jared Jackson	**Date to Test by**	05/25/2025

Test ID	GLUC-006-005	**Test Name**	Location Services access – don't allow
Description	The Grocery Store Map page should show default map after pressing “don’t allow” button for location services prompt		
Pre-Conditions	User is logged in and is on home screen. Location services permission should be unset		
Steps	5. Click Grocery Store Map page. 6. When location services window appears, press “Don't Allow”		
Post-Conditions	Apple alert window should disappear and map should be default. (Expected result of this should be double checked before testing)		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Jared Jackson	**Date to Test by**	05/25/2025

Test ID	GLUC-015-001	**Test Name**	Grocery Store Map page is rotatable
Description	The user is able to access the rotation map control		
Pre-Conditions	The user is logged in and has allowed location services		
Steps	1. Open Grocery Store Map page from home screen. 2. Rotate map		
Post-Conditions	Map has been rotated		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/25/2025

Test ID	GLUC-015-002	**Test Name**	Grocery Store Map page is zoomable
Description	The user is able to access the zoom map control		
Pre-Conditions	The user is logged in and has allowed location services		
Steps	1. Open Grocery Store Map page from home screen. 2. Zoom into or out of map		
Post-Conditions	Map zoom has been changed		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/14/2025

Test ID	GLUC-015-003	**Test Name**	Grocery Store Map page is pannable	
Description		The user is able to access the pan map control		
Pre-Conditions		The user is logged in and has allowed location services		
Steps		1. Open Grocery Store Map page from home screen. 2. Pan away from initial map location		
Post-Conditions		Map view has been changed		
Test Status		SUCCESSFUL		
Creator		Carson Ford	**Date Created**	04/27/2025
Tester		Carson Ford	**Date to Test by**	05/14/2025

Test ID	GLUC-015-004	**Test Name**	Grocery Store Map page "Search for Stores" button functions properly	
Description		The user should be able to press the "Search for Stores" button and automatically have stores searched for them. Markers should appear on the map to indicate stores' locations.		
Pre-Conditions		The user is logged in and has allowed location services		
Steps		1. Open Grocery Store Map page from home screen. 2. Wait for map to load current location. 3. Click the "Search for Stores" button. 4. Wait 2 seconds.		
Post-Conditions		Markers should appear for nearby grocery stores.		
Test Status		SUCCESSFUL		
Creator		Carson Ford	**Date Created**	04/27/2025
Tester		Krisi Hristova	**Date to Test by**	05/24/2025

Test ID	GLUC-015-005	**Test Name**	Grocery Store Map page - store details	
Description		The user should be able to press any grocery store marker to see details about the location		
Pre-Conditions		The user is logged in, has allowed location services, and has searched for stores on the grocery store map page.		
Steps		1. Click a marker.		
Post-Conditions		The store details half-page should appear for that store.		
Test Status		SUCCESSFUL		
Creator		Carson Ford	**Date Created**	04/27/2025
Tester		Francisco Cruz-Urbanc	**Date to Test by**	05/25/2025

Test ID	GLUC-015-006	**Test Name**	Grocery Store Map page - store details – open in maps
Description	The user should be able to view a details page for any store marker and can open that location in the Apple Maps app.		
Pre-Conditions	The user is logged in, has allowed location services, and has searched for stores on the grocery store map page. Device should have been through the initial set up of the Maps app.		
Steps	1. Click a marker. 2. Click the “Open in Maps” button.		
Post-Conditions	Maps app should be opened and the selected location should appear. May have to be tested manually.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/25/2025

Test ID	GLUC-015-007	**Test Name**	Grocery Store Map page - store details – street view appears
Description	The user should be able to view a details page for any store marker and can open that location's street view.		
Pre-Conditions	The user is logged in, has allowed location services, and has searched for stores on the grocery store map page.		
Steps	1. Click a marker. 2. Click the Street View window.		
Post-Conditions	A view should appear of the street view of the selected location		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/25/2025

Test ID	GLUC-015-008	**Test Name**	Grocery Store Map page - store details – street view is interactable
Description	The user should be able to interact with the street view for any nearby store location		
Pre-Conditions	The user is logged in, has allowed location services, and has searched for stores on the grocery store map page.		
Steps	1. Click a marker. 2. Click the Street View window. 3. Drag the screen to look around. 4. Tap a spot on the street to move around.		

Post-Conditions	The user's location in the 3D space of the street view should have moved and the camera perspective should have changed from looking around.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/25/2025

Test ID	GLUC-015-009	**Test Name**	Exit Grocery Store Map page
Description	The user should be able to leave the grocery store map page with back button		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Grocery Store Map" page 2. Click "Back" button.		
Post-Conditions	The grocery store map page should be exited and the user should be taken back to the home page.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Jared Jackson	**Date to Test by**	05/25/2025

Tests for Nutrient Tracker page:

Test ID	GLUC-012-001	**Test Name**	Nutrient Tracker page can be accessed
Description	The user should be able to visit the nutrient tracker page		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Nutrient Tracker" page		
Post-Conditions	The nutrient tracker page appears.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/29/2025

Test ID	GLUC-012-002	**Test Name**	Logs appear on Nutrient Tracker page
Description	A log of the user's nutrients should appear after entering any value of nutrients.		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Nutrient Tracker" page 2. Type "1" into protein entry. 3. Press enter button.		

Post-Conditions	Log window pops up with a log at the top with the current date and a value of 1 for protein, 0 for carbs, and 0 for fiber.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/25/2025

Test ID	GLUC-012-003	**Test Name**	Nutrient tracker tracks a running total for nutrient in a day.
Description	A running total of a user's nutrient intake should be calculated.		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Nutrient Tracker" page 2. Type "1" into protein/carb/fiber entry. 3. Press enter button. 4. Validate log. 5. Type "1" into protein/carb/fiber entry. 6. Press enter button.		
Post-Conditions	Log window shows a running total of the protein/carb/fiber count by showing "2g" for total intake.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/28/2025

Test ID	GLUC-012-004	**Test Name**	Exit Nutrient Tracker page
Description	User should be able to exit the nutrient tracker page		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Nutrient Tracker" page 2. Click "Back" button		
Post-Conditions	User should be taken back to home page.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/28/2025

Test ID	GLUC-012-005	**Test Name**	Nutrient Tracker page alphabetical characters
Description	User should not be able to enter alphabetical characters into nutrient tracker		
Pre-Conditions	The user is logged in.		
Steps	1. Click "Nutrient Tracker" page		

	2. Click any field. 3. Enter “abc” 4. Press enter button.		
Post-Conditions	Log should not be corrupted at all. Invalid input should be handled gracefully.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/29/2025

Test ID	GLUC-012-006	**Test Name**	Nutrient Tracker page special characters
Description	User should not be able to enter special characters into nutrient tracker		
Pre-Conditions	The user is logged in.		
Steps	1. Click “Nutrient Tracker” page 2. Click any field. 3. Enter “%$&” 4. Press enter button.		
Post-Conditions	Log should not be corrupted at all. Invalid input should be handled gracefully.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Jared Jackson	**Date to Test by**	05/25/2025

Test ID	GLUC-012-007	**Test Name**	Nutrient Tracker page whitespace
Description	User should not be able to enter whitespace into nutrient tracker		
Pre-Conditions	The user is logged in.		
Steps	1. Click “Nutrient Tracker” page 2. Click any field. 3. Enter “ “. 4. Press enter button.		
Post-Conditions	Log should not be corrupted at all. Invalid input should be handled gracefully.		
Test Status	SUCCESSFUL		
Creator	Carson Ford	**Date Created**	04/27/2025
Tester	Thomas Capro	**Date to Test by**	05/28/2025

Tests for language settings:

Test ID	GLUC-007-001	**Test Name**	Spanish Version of Application	
Description		If the user's phone is set to Spanish, then the application should automatically be in Spanish when it is opened.		
Pre-Conditions		The user's phone has its primary language set to Spanish.		
Steps		1. Open the application. 2. View every custom screen (for example, main page, recipes page, etc.) 3. Check that all page names, recipe descriptors, pop-messages, nutrient descriptors and other content not including actual recipe content is translated into Spanish.		
Post-Conditions		None.		
Test Status		SUCCESSFUL		
Creator		Francisco Cruz-Urbanc	**Date Created**	04/27/2025
Tester		Krisi Hristova	**Date to Test by**	05/14/2025

Test ID	GLUC-007-002	**Test Name**	Spanish Version of Authentication	
Description		If the user's phone is set to Spanish, then the AWS-hosted authentication page should be in Spanish.		
Pre-Conditions		The user's phone has its primary language set to Spanish.		
Steps		1. Open the application (sign out if needed) 2. Sign in 3. Check that sign in page is in Spanish		
Post-Conditions		None.		
Test Status		FAILED		
Creator		Francisco Cruz-Urbanc	**Date Created**	04/27/2025
Tester		Francisco Cruz-Urbanc	**Date to Test by**	05/25/2025

Test ID	GLUC-007-003	**Test Name**	Spanish Version of API data
Description		If the user's phone is set to Spanish, then the application should automatically make API calls to the Spanish version of the API and display recipes translated to Spanish.	
Pre-Conditions		The user's phone has it's primary language set to Spanish.	
Steps		1. Open the application 2. Click on the Recipes Search button	

	3. Check that the recipes that have been fetched are in Spanish instead of English.		
Post-Conditions	None.		
Test Status	FAILED		
Creator	Francisco Cruz-Urbanc	**Date Created**	04/27/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/22/2025

7.2 API Test Cases

Tests for Recipe Saving

Test ID	GLUC-010-001	**Test Name**	Save Recipe to User Profile
Description	Ensure that when a recipe is saved, it is correctly stored in the user's profile in the database.		
Pre-Conditions	User is logged in and has not saved this recipe before.		
Steps	1. Save a new recipe from the individual recipe page. 2. Query user's saved recipes backend or use Saved Recipes page.		
Post-Conditions	The saved recipe exists in the user's profile data (server-side/database).		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/15/2025

Test ID	GLUC-010-002	**Test Name**	Prevent Duplicate Saves
Description	Test that saving a recipe already saved does not create duplicate entries in the backend.		
Pre-Conditions	User already saved the recipe once.		
Steps	1. Attempt to save the same recipe again. 2. Check backend database for recipe entries under the user.		
Post-Conditions	No duplicate recipe is added. Either the server rejects the request gracefully or ignores it.		
Test Status	FAILED		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Krisi Hristova	**Date to Test by**	05/27/2025

Test ID	GLUC-010-003	**Test Name**	Fetch and Display Saved Recipes

Description	Ensure that saved recipes can be fetched correctly and displayed in the “Saved Recipes” section.		
Pre-Conditions	User has one or more recipes saved.		
Steps	1. Login. 2. Navigate to the Saved Recipes page. 3. Verify list content against backend data.		
Post-Conditions	Displayed recipes match backend data (correct names, images, etc.).		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Francisco Cruz-Urbanc	**Date to Test by**	05/17/2025

Test ID	GLUC-010-004	**Test Name**	Saved Recipes Persist After Logout
Description	Verify that recipes saved to a user’s account persist after logging out and logging back in.		
Pre-Conditions	User has saved one or more recipes.		
Steps	1. Save a recipe. 2. Log out. 3. Log back in. 4. Navigate to Saved Recipes page.		
Post-Conditions	Previously saved recipes are still present and accessible.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Carson Ford	**Date to Test by**	05/22/2025

Test ID	GLUC-010-005	**Test Name**	Handle Save Failure
Description	If saving a recipe fails (network issue, API error), the app should notify the user and not falsely show it as saved.		
Pre-Conditions	Network disabled or simulate API failure.		
Steps	1. Try saving a recipe while offline or simulate API 500 error.		
Post-Conditions	Save operation fails gracefully (e.g., error message shown). No incorrect entries saved.		
Test Status	SUCCESSFUL		
Creator	Thomas Capro	**Date Created**	04/27/2025
Tester	Jared Jackson	**Date to Test by**	05/27/2025

7 Exit Criteria

The Glucose Genie system will be considered ready for release when all planned functional and nonfunctional test cases have been tested and passed successfully. All arising issues and critical defects must be identified during testing, resolved, retested, and closed. Stable performance under expected load conditions, no outstanding security vulnerabilities and verification that no new issues have been introduced through bug fixing is also required. Additionally, all system documentation including SRS and user manuals must be updated to portray the current state of the application. Finally, approval for release should be given by the stakeholders before deployment to the App Store.

8 Risks

8.1 API Access

There is the risk that the APIs that we use in this project might not always be available. For example, this project uses the Edamam API for recipe searching and recipe information. If the Edamam API is offline for maintenance or some other issue, features such as recipe searching and recipe details will not function properly. Additionally, there is the risk that access to the different APIs that we use can be revoked. This can most likely occur if access to the APIs is not renewed.

8.2 Time Constraints

As we are approaching the end of the spring quarter, there is a limited amount of time left to dedicate to coding, testing, and documenting this project. Testing time could be reduced if certain features need extra development time or if project documentation needs to be updated.

8.3 Cloud Service Interruptions

There is a risk that cloud services such as Amazon EC2 (server hosting), RDS (database), and Cognito (user authentication) could experience outages, degraded performance, or security breaches during development or testing. This could lead to partial or full system unavailability, which would impact feature testing, user login functionality, and data storage operations.

8.4 Cloud Service Interruptions

There is a risk that certain UI or feature behaviors could differ between different iPhone models or iOS versions despite successful simulator testing. Issues could include layout glitches, slower performance, or feature malfunction.

8.5 Assumptions

- All required third-party APIs (Edamam Recipe API, Map APIs) will remain available and functional throughout testing.
- Development and testing environments (macOS, Xcode, Swift) will remain stable and compatible with the application throughout testing.
- iOS simulator testing accurately reflects behavior on actual iOS devices.
- Internet access will be available during testing sessions to interact with backend servers and APIs.
- Test data used during API and UI testing will be representative of real-world scenarios.
- Testers will have access to the most recent builds.
- Team members assigned to testing tasks will be available and complete testing on schedule.

9 Appendix

Appendix A: Version History

Version 1 – The initial version of this document.

Version 2 – Correct duplicated test ID's and update document title to Glucose Genie

Version 3 – Update test assignments, due dates and test results from testing