

# Greenlist

## Development Document

CS-4850 Section I

Spring 2026

Professor Perry

February 1st 2026



Tate Swidorski  
email: tatecswid@gmail.com



Kevin Gomes  
email: kevgom11@gmail.com



Chris Heyward  
email: chrisghrobot@gmail.com

# Table of Contents

<b>Table of Contents</b> .....	<b>1</b>
<b>Development Process</b> .....	<b>2</b>
i - UI and Prototyping.....	2
ii - Frontend Development Tools.....	2
iii - Authentication System Implementation.....	2
iv - Grocery List Management.....	3
<b>Database Connection</b> .....	<b>3</b>
<b>Project Setup</b> .....	<b>4</b>
<b>Development Summary</b> .....	<b>4</b>

# Development Process

## ***i - UI and Prototyping***

The first stage of development for Greenlist has focused primarily on designing the User Interface and planning the user experience. The team has used Figma Make and v0 to create UI mockups for the applications. These designs include screens for login, signup, viewing all of the users grocery lists, and inside of the grocery lists themselves.

Using Figma and v0 allows the team to define the structure and layout of the application before implementing it in code. The mockups are used as references when building the React Native interface. Changes to layout, navigation flow, and screen organization are first tested in the prototypes and then translated into React Native components during development.

## ***ii - Frontend Development Tools***

The mobile application is being implemented using React Native. The user interface is built using React Native components such as View, Text, TextInput, Button, and FlatList.

Each screen of the application, such as the login screen, list overview screen, and grocery list detail screen, is implemented as a separate component, typically managed through a navigation library such as React Navigation.

To help create the frontend interface, v0 is used to generate UI layout ideas and component structures. These generated layouts are then translated into React Native components and integrated into the application code. Components are organized into screens that allow users to navigate between viewing their grocery lists and editing list items.

## ***iii - Authentication System Implementation***

The project implements user authentication using Firebase Authentication. This system allows users to create accounts and log in using their email and a password OR using google.

The authentication workflow includes the following development steps:

1. Creating login and registration screens with help of Figma and v0.
2. Connecting the application to Firebase Authentication.
3. Validating user login credentials through firebase.
4. Maintaining user session state within the application.

Once authentication is fully implemented, each Greenlist user account will be associated with the grocery lists that the user owns or has been invited to collaborate on.

#### ***iv - Grocery List Management***

The project implements grocery list management using the React Native frontend and Firebase Realtime Database. This functionality allows users to create grocery lists and manage the items within those lists.

The grocery list workflow includes the following development steps:

1. Creating screens that display the user's grocery lists with help of React Native and v0.
2. Connecting the application to the Firebase Database system.
3. Allowing users to create new grocery lists through input fields in the application.
4. Allowing users to add, edit, and delete grocery lists and list items.
5. Updating the Firebase Database whenever a list or items is changed.
6. Show the change in real-time.

Once grocery list management is fully implemented, users will be able to organize multiple grocery lists and manage the items within each list directly through the Greenlist application.

## **Database Connection**

The Greenlist application connects to the Firebase Database to store and retrieve grocery lists and grocery list items. The database connection process includes the following development steps:

1. Creating a database on firebase.
2. Adding Firebase to the React Native project using the Firebase configuration files.
3. Connecting the React Native application to Firebase Realtime Database.

4. Reading grocery list and item data from the database.
5. Writing new lists and item updates to the database when changes are made.

Once the database connection is fully implemented, grocery lists and items will be stored in Firebase and retrieved by the application when users access their lists

## Project Setup

In the event that a developer wants to download this project, the following steps are required to run it:

1. Install Node.js and a package manager (npm or yarn).
2. Install React Native Expo
3. Clone the repository from GitHub.
4. Set up a Firebase project through the Firebase console.
5. Enable Firebase Authentication and Firebase Database.
6. Add the Firebase configuration files to the React Native project.
7. Run the application on an emulator or connected mobile device.

To test this if the application is working, log in on two different accounts and try creating lists, editing items, collaborating, etc.

## Development Summary

At the current stage of development, the Greenlist project has focused primarily on establishing the foundation of the application. The user interface layout and screen structure have been designed using Figma and v0, and the React Native project structure has been set up for frontend development. Authentication and Firebase integration have now been implemented and integrated into the application.

The development process has prioritized building the core components required for the application, including the login and signup interfaces, grocery list viewing screens, and the structure for managing grocery list items. These components are now connected to Firebase services to support user authentication and data storage.

Some features are still in progress, including full implementation of grocery list collaboration and final database structure. Additional testing will be required as authentication and database connections continue to be refined.

At this stage, the project has established the core structure of the system and is continuing development toward completing the remaining functionality.