**ReciPlease**

AUTHORS:

Stephen R Ouellette

**Recipes that please with ease**

**[IMAGE OF PROTOTYPE]**

**Technical Field**

This project utilizes the Software Programming and Graphic Design fields

**Background Information**

My family has a plethora of recipes that we utilize daily. We create a two-week meal list of everything were going to cook, as well as create a shopping list from said meal list. Because of this, I wanted to design a program that could store all these recipes, make the meal lists, and create shopping lists from the meal lists. I also wanted the ability to share recipes with family members, so the idea for a sharing function was born.

**Prior Art**

It seems like there are a plethora of recipe programs available in the app store, but many of them lack in different categories. Most if not all the apps that I have found to date all require some sort of network connectivity for the user to search the online databases of recipes. The functions that will set my CIP apart from other projects is to give the user the ability to create their own recipes, saving these for later, allowing the user to select recipes to create a ‘meals list’ in which generates a shopping list the user can then utilize while shopping. These simple functions may seem trivial, but I believe they will be a large selling point, as many users do not always have connectivity to the internet. I have also yet to find an app that provides QR code recipe sharing, they do have recipe sharing via usernames and emails, but a simple QR scan that can add an entire recipe to another users list could be a great selling point.

**Project Description**

ReciPlease’s target user is the 18-60 year old male and female. Anyone who cooks at home and/or wants to store or share their recipes. This application will have the following functions: Create, Store, and Share Recipes, Create Meals lists, and finally Create Shopping lists.

**Innovation Claim**

ReciPlease is an innovation because of its ability to share and store a person’s recipes without the need for a network connection.

**Usage Scenario**

Users will be able to utilize all features of this program in a non-network environment. There will be no need for network connections to create, modify, share, and store recipes with this application.

**Evaluation Criteria**

The following questions are to be asked of ReciPlease to ensure its success.

Can you create a recipe?

Can you store a recipe?

Can you delete a recipe?

Can you share a recipe?

Can you create a meals list?

Can you delete a meals list?

Can you create a shopping list?

Can you delete a shopping list?

Do all functions work as intended?

Is the Graphic User Interface appealing and functional?

**Objectives and Tasks Associated with the Project**

1. Program Layout – Creation Phase
   1. Create Main Page
   2. Create Recipe Creation Page
   3. Create Recipe List Page
   4. Create Meals List Page
   5. Create Shopping List Page
   6. Create Recipe Share Page
2. Program Layout – Finalization Phase
   1. User testing for layout and usability
3. FUNCTION: Recipe (Create)
   1. Write code for the creation of recipes
   2. Test code
4. FUNCTION: Recipe (Delete)
   1. Write code for the Deletion of recipes
   2. Test code
5. FUNCTION: Recipe List (Create)
   1. Write code for the Creation of a recipe list
   2. Test code
6. FUNCTION: Recipe List (Add)
   1. Write code to add recipes to the list
   2. Test code
7. FUNCTION: Recipe List (Remove)
   1. Write code to remove recipes from the list
   2. Test code
8. FUNCTION: Recipe List (Delete)
   1. Write code to delete a recipe list
   2. Test code
9. FUNCTION: Meals List (Create)
   1. Write code to create a Meals list
   2. Test code
10. FUNCTION: Meals List (Add)
    1. Write code to add a meal to the Meals list
    2. Test code
11. FUNCTION: Meals List (Remove)
    1. Write code to remove a meal from a Meals list
    2. Test code
12. FUNCTION: Meals List (Delete)
    1. Write code to delete a Meals list
    2. Test code
13. FUNCTION: Shopping List (Create)
    1. Write code to create a Shopping list
    2. Test code
14. FUNCTION: Shopping List (Add)
    1. Write code to add an ingredient to the Shopping list
    2. Test code
15. FUNCTION: Shopping List (Remove)
    1. Write code to remove an ingredient from the Shopping list
    2. Test code
16. FUNCTION: Shopping List (Delete)
    1. Write code to delete a Shopping list
    2. Test code
17. FUNCTION: Recipe (Share)
    1. Write code to share a recipe
    2. Test code
18. FUNCTION: Create QR Code
    1. Write code to create a QR code for sharing a recipe
    2. Test code
19. FUNCTION: Scan QR Code
    1. Write code to scan and read a QR code
    2. Test code

**Description of Design Prototype**

Prototype will be created in the Unity Engine for the Android OS. After successful completion of the first iteration of the program, I will begin porting the application to the Apple OS. Currently once the program is downloaded onto your phone, the user needs only to click the icon, and the program will open to the main menu screen. From this point the user can check lists, create recipes, and modify recipes.

**Evaluation Plan**

In order to ensure success, the application will be distributed to family members and a few close friends for a closed beta testing iteration of the program. Users will be able to report bugs, and give important user feedback, that will ensure the program is working nominally. If the overall experience from the beta is a success and were able to appease the audience. Then the application will be considered a success. Any and all bugs and design quarrels will be addressed during this phase. If any changes are to be made, I will update the application during the beta process and once the application is running correctly and the majority of users are happy with the program, I will release to the general public.

**Project Completion Assessment**

Currently the following functions are working as intended: Create/Share/Delete recipes, Create/Modify Meals list, Create Shopping list. The overall process of getting these functions was pretty easy and straight forward. Halfway through the creation of these functions I decided to refactor and change the programming methods. I employed an event based system to ensure optimal memory and data usage. To Be Continued…

**Appendices**

TBC.