



JUNE 30, 2020

CEF412: HUMAN COMPUTER INTERFACE  
PERVASIVE FRIDGE FRIEND: A FINAL REPORT

MAFANY TANDE MYLES BILONG  
FE17A037

Software Engineering  
Department of Computer Engineering

***Instructor(s)***

**Hughes M. Kamdjou, M.Eng**



## TABLE OF CONTENTS

CHAPTER ONE.....	1
Abstract (Project goal): .....	1
Acknowledgements .....	2
Dedication.....	2
Tools and technologies used.....	3
CHAPTER TWO .....	4
Design Process.....	4
Project features: .....	8
Original Project Design:.....	9
File Structure:.....	10
CHAPTER THREE .....	11
The Design: .....	11
The App: .....	12
Challenges .....	14
Link to the Source Code of the App: .....	14

## CHAPTER ONE

### **Abstract (Project goal):**

My app's purpose is to help users organize and easily visualize their groceries, by using a Universal Product Code (UPC) supplied database. Users will be able to barcode scan their groceries as they unpack them, adding them to their current list with a single button press. Tags such as expiration date and food type are automatically added, so users can easily sort their grocery list by what to eat (or throw away) next.

## **Acknowledgements**

I would like to acknowledge all the people who helped me to develop this project. Firstly, I would like to thank my instructor, Mr. Hughes M. Kamdjou, M.Eng, for giving me a chance to critically think of a solution to some of mankind's project using Human-Computer Interaction.

I want to express profound gratitude to Usmaila Abdoul Moumini who spared no effort in lending me a hand and his vast expertise. Not forgetting Mbino Ruth Atem and Bassah Roselyn Nahjela who provided me with resources when I ran out.

## **Dedication**

I dedicate this work to the advancement of Sustainable Development Goals 1 and 2 - No Poverty and Zero Hunger respectively. This project will go a long way towards eliminating poverty and hunger worldwide if implemented on a large scale.

## Tools and technologies used

### 1. Android Studio

Android Studio provides code editing, debugging, and testing tools all within an easy-to-use drag-and-drop interface. It is free to download and is supported not only by Google, but also by a large and actively engaged community of Android developers.

### 2. Java

This object-oriented programming language is the official language for Android development. This language is easy to handle and many open source libraries are made available for users to choose from.

### 3. Gradle

Back in 2013, Google endorsed [Gradle](#) as a build system for Android apps. Based on Apache Maven and Apache Ant, Gradle is one of the most popular development tools for creating large-scale applications involving Java.

Developers like using Gradle in conjunction with Android Studio because it's very easy to add external libraries using a single line of code.

### 4. XML

Extensible Markup Language provides a standard method to access information, making it easier for applications and devices of all kinds to **use**, store, transmit, and display data

## CHAPTER TWO

### Design Process

Considering the constraints, I planned out three stages of the design project: 1. **research**, 2. **design sprint**, and 3. **iterations with user feedback**. In research stage, I mostly did literature review to set a clear design direction. In design sprint stage, I worked on a prototype within a few weeks using UI and paper designs, with design methods such as wireframing.

#### 1. Research

First, I clarified the challenge question and defined the problem space. I did literature review on FAO (Food and Agricultural Organization of UN) statistics and other UN reports to obtain more information related food waste in less developed countries, like Cameroon in this case. What I learned was that food waste occurring at the consumer level was considerable. People may discard food because they buy more food than their needs, miss timing to consume the perishables, or simply follow the expiration dates.

According to the information, I focused on *changing behaviors of consumers*. I decided that *helping consumers to waste food less would be the fundamental way* to make them being more conscious about food waste problem, as an HCI student.

To change the consumer behavior, I referred two psychology theories to guide the general direction of design. The first theory is *Cognitive Miser model*, a tendency to find a “shortcut” to process information. Consumers would choose a solution which helps them efficiently manage their stored foods with *minimal or no efforts*.

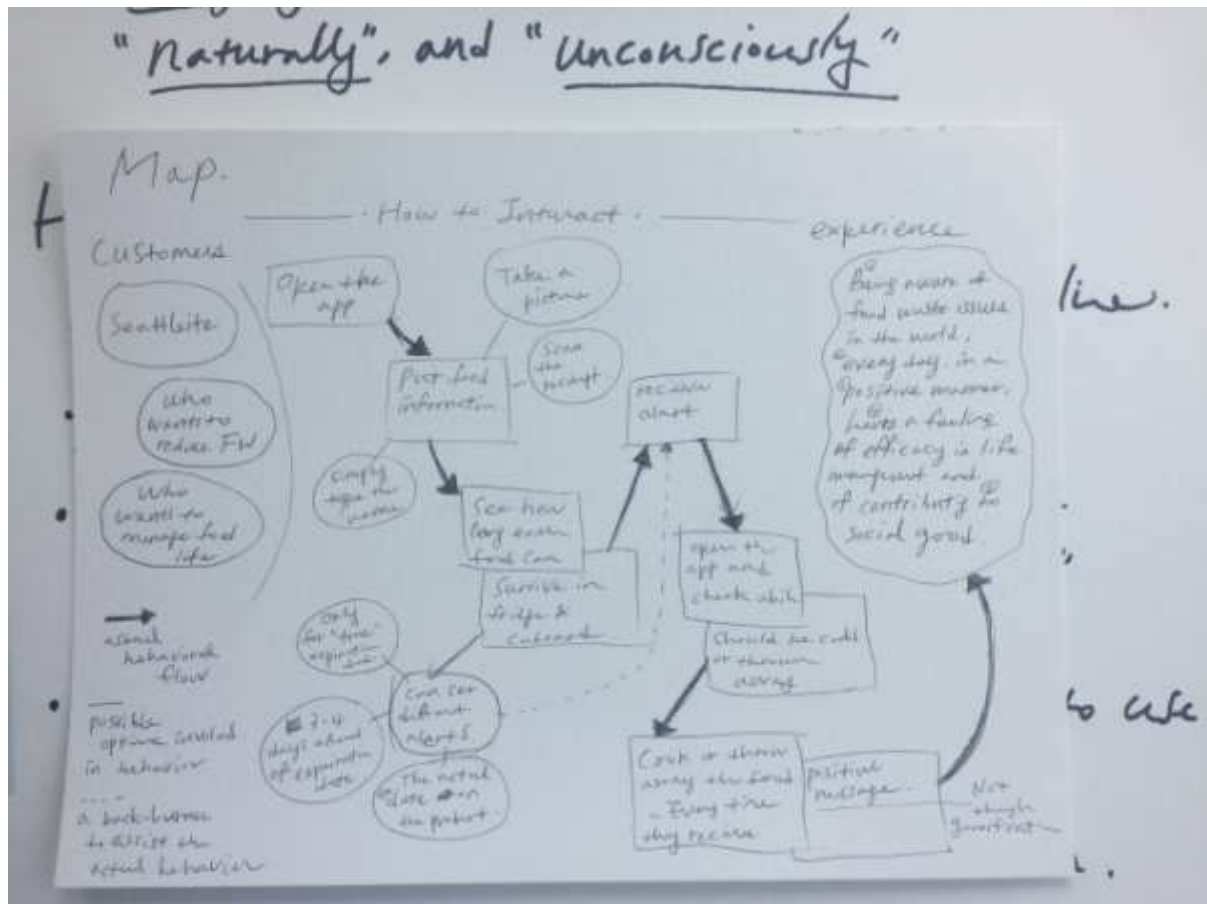
The second is *Preference for Immediate Rewards* including sociomotional ones. Consumers would choose a solution that compensates them with *immediate positive feelings* such as efficiently managing the part of life and contributing for the good of the greater community. The design should aim to create *cognitively smooth* and *socioemotionally positive* user experience.

## **2. Design Sprint**

After setting up the direction of the research, I estimated the time and resources I had for the actual design process. Based on the limitation, *Design sprint* (<https://www.gv.com/sprint/>) seemed to be a pertinent solution to deliver a prototype within five days (although I used more time than that).

For the platform, I decided to *design a mobile app* considering its high ubiquity for users. The app should aim to help users manage their food daily, anytime and anywhere.

Firstly, I set big goals and measured risks regarding the initial direction from the research. Then I created a *map of challenge* to draw out target users and the four stages of user experience: (1) Inserting and (2) evaluating the food information, (3) making a decision, and (4) taking an action.



Next, based on the map, I created a *persona* representing the target audience, a single, highly educated female city-dweller living a busy life but wanting to maintain healthy life style. I moved to *comparative analysis* and *Sketching* stages. For the analysis, I reviewed *Google Tasks* to learn about pros and cons in their design.

Based on the analysis, I set four features to be implemented in my app: **Simple, visually straightforward, cognitively efficient, and socioemotionally positive.** To make it simple, the storage sections and type of interactions were limited to three, considering the attention and memory span of human. To make it visually straightforward, the home screen contained all the information that the app could provide. To make it cognitively efficient, details (i.e., expiration date, type of item, ...) should be easily recognized and not too diversified.





**Project features:**

- Scan barcodes from camera
- Fetch barcode UPC info using Computer Vision API
- Store UPC Items in a database
- Sort UPC Items by food category or expiration date

Original Project Design:



Figure 2: Home Page



Figure 3:Results after Barcode Scan

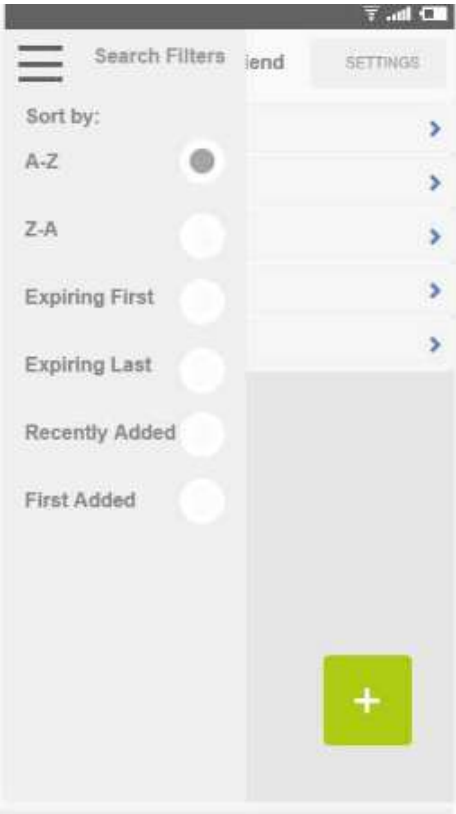


Figure 4:Menu with Sorting Options

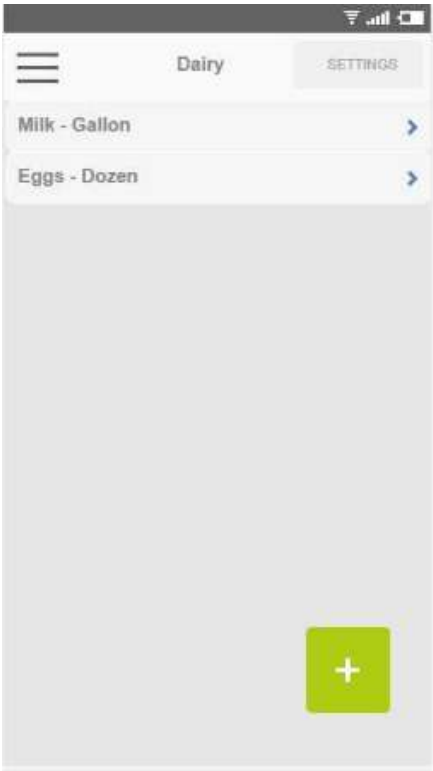


Figure 1: Sorted by Category

## **File Structure:**

### UI Files:

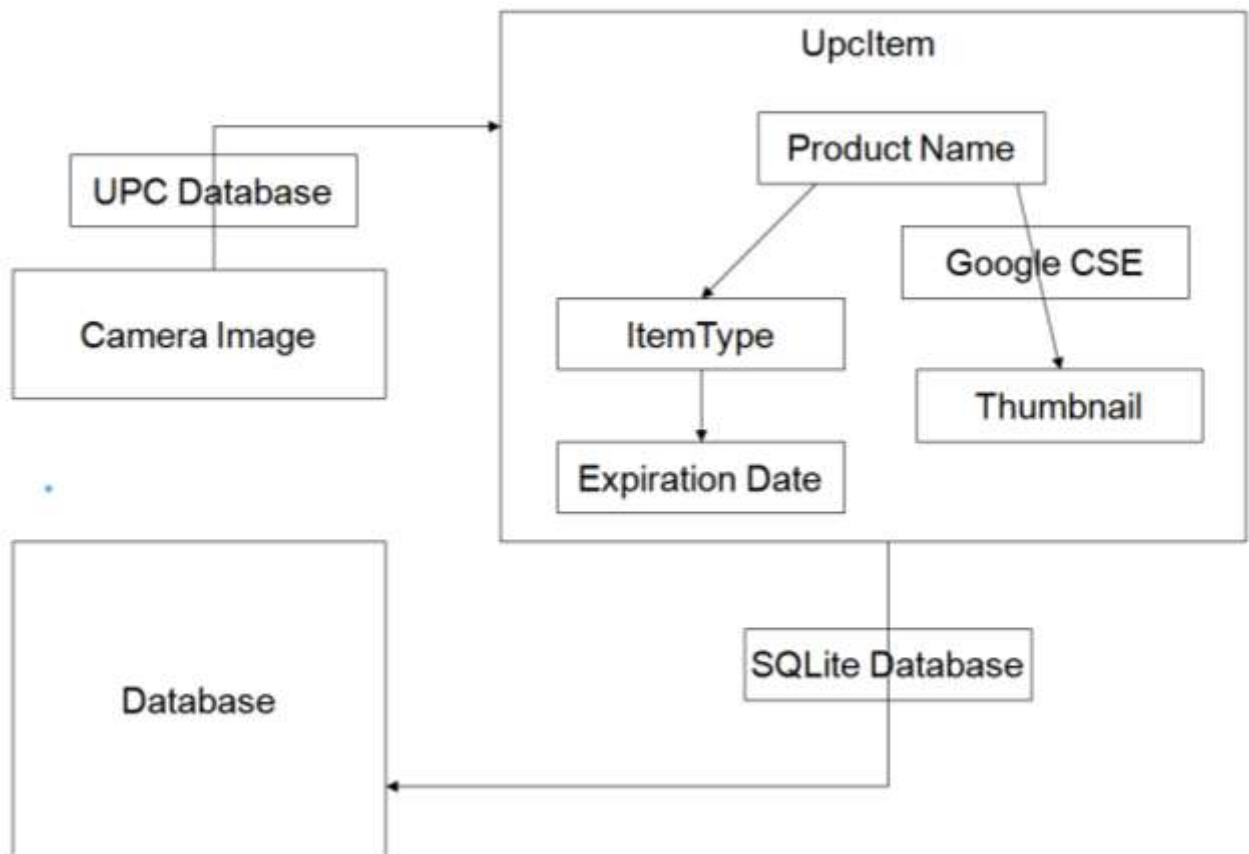
- Activity\_Main: homepage with photo button
- Activity\_Results: Results screen after barcode scan; shows product name and info, with Add button
- content\_main:
- drawer\_list\_item:
- list\_item: individual xml for each list item

### Java Files:

- DBHelper: Creates the SQLite Database for storing food items by UPC type including expiry date, type of food, name, etc.
- ItemType: associates common words in UPC item name with appropriate food category, defines item typing
- MainActivity: Displays the homepage with current items in database, along with button to add new via photo
- NetworkUtilis: Handles network operations of looking up barcodes
- Results\_Activity: Looks up barcode scan and retrieves UPC Item with description & photo, adds to the database
- UpclItem: Defines a UPC Item including name, item type, and shelf life, to be added to database
- UpclItemLoader: Asynchronously loads UpclItem in background

## CHAPTER THREE

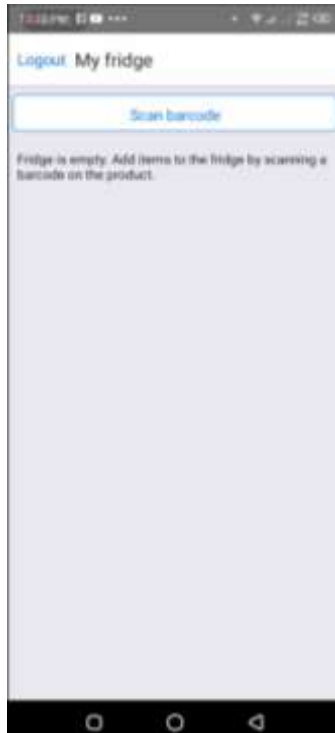
### The Design:



## The App:

### ➤ Upc Item

- A product with a barcode ID, category (dairy, meat, etc.),



and thumbnail image

### ➤ Item Type

- Assigns a shelf life to products based on their categories

### ➤ Network Utilities

- Calls UpcDatabase API to retrieve a

product name from a UPC

- Uses Google CSE to fetch a thumbnail image

### ➤ Main Activity

- Opens a camera to



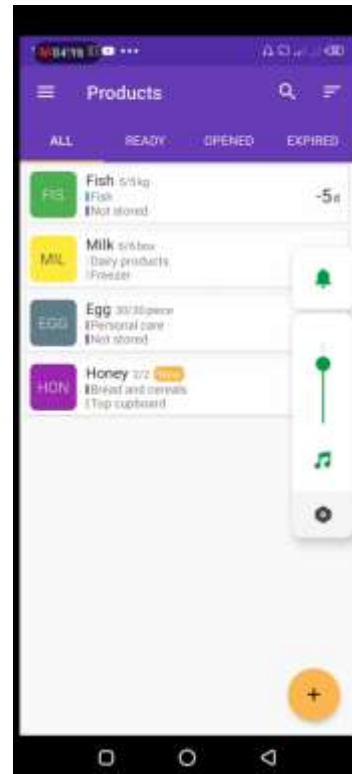
capture a barcode image

### ➤ Results Activity

- Converts images to barcodes using the Google Play Services Barcode API

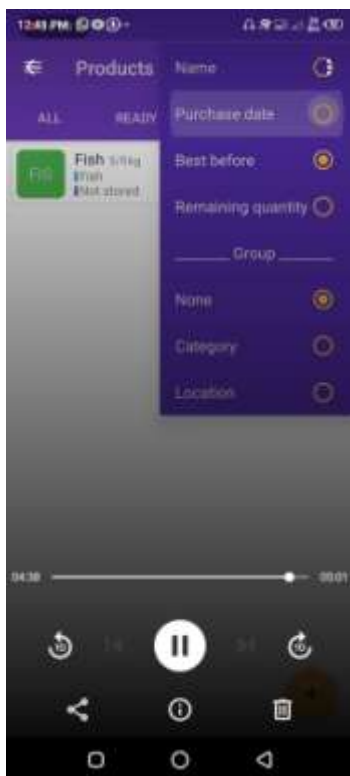


- Sort by product name, date added, and expiration date



### ➤ Settings

- Filter products by category



### ➤ Database Helper

- Manages insertions, deletions, and queries for a SQLite database

○

## **Challenges**

I encountered two major challenges. The first was to reshape the high-level question I had chosen to answer into a solvable problem in the form of my project. As the question did not indicate specific constraints or conditions, I had almost unlimited level of freedom to approach the problem; well-defined strategy was the key.

On the contrary, the second was limited time and resources. Although I was given about two months to complete the project, the amount of time I actually could spend was at least thrice as much to finish the project and I was the only workforce.

### **Link to the Source Code of the App:**

[https://www.github.com/MafanyTandeMyles/fridge\\_friend](https://www.github.com/MafanyTandeMyles/fridge_friend)