



National College of Ireland

Computing

Software Development

Academic Year 2020/2021

Max Delaney

18100368

x18100368@student.ncirl.ie

GFree

Technical Report

Contents

Executive Summary	4
1.0 Introduction	5
1.1. Background	5
1.2. Aims.....	6
1.3. Technology.....	6
1.3.1. Android Studio	6
1.3.2. Firebase.....	6
1.3.3. Java.....	6
1.3.4. Google Machine Learning (Text Recognition).....	7
1.3.5. JUnit 5	7
1.3.6. Espresso (UI-Testing)	7
1.3.7. GitHub	7
1.4. Structure	8
2.0 System.....	8
2.1. Requirements.....	8
2.1.1. Functional Requirements.....	9
2.1.2. Non-Functional Requirements.....	9
2.1.3. Use Case Diagrams.....	10
2.1.3.1. Requirement 1: User scans packet	10
2.1.3.2. Description & Priority.....	10
2.1.3.3. Use Case	10
2.1.3.4. Requirement 2: User reviews.....	13
2.1.3.5. Description & Priority.....	13
2.1.3.6. Use Case	13
2.1.4. Data Requirements	15
2.1.5. User Requirements	15
2.1.6. Environmental Requirements	15
2.1.7. Usability Requirements.....	15
2.2. Design & Architecture	15
2.2.1. Android Activity Flow Diagram	15
2.2.2. Android Activity Structure Diagram	16
1.1. Implementation	17
1.1.1. Image Text Converter.....	17
1.1.2. Packet Scanner Activity.....	18
1.1.3. Register Activity	19

1.1.4.	Login Activity	20
1.1.5.	Natural Language Processor (Packet Scanner)	21
1.2.	Graphical User Interface (GUI)	23
	Low Fidelity	23
	Wireframes	25
	High Fidelity	27
1.3.	Testing	37
	<i>Trunk Test</i>	42
	Think Aloud	42
	<i>5 Second Test</i>	42
	<i>Click Test</i>	43
	<i>Heuristic Evaluation</i>	43
	System Usability Scale Questionnaire	43
2.0	Evaluation	44
	CPU	44
	RAM	46
	Network	47
	Energy Usage	48
3.0	Conclusions	49
4.0	Further Development or Research	50
5.0	References	50
6.0	Appendices	51
	Project Plan	51
	Contents	52
	Objectives	53
	Background	53
	Technical Approach	54
	Special Resources Required	54
	Project Plan	55
	Technical Details	56
	Evaluation	57
	Ethics Approval Application	58
	Introduction	58
	Sources of Data:	59
	CHECKLIST	60
	ETHICS CLEARANCE GUIDELINES WHEN HUMAN PARTICIPANTS ARE INVOLVED	60

Considerations in data collection.....	60
Informed consent.....	60
Ethics Appendix I.....	62
Ethics Appendix II.....	62
User Testing Results.....	64
Trunk Test.....	64
Think Aloud.....	69
5 Second Test.....	72
Click Test.....	75
Heuristic Evaluation.....	76
Visibility of system status.....	76
Match between system and real world.....	76
User control and freedom.....	76
Consistency and standards.....	76
Error prevention.....	77
Recognition rather than recall.....	77
Flexibility and efficiency of use.....	77
Aesthetic and minimalist design.....	77
Help users recognize, diagnose, and recover from errors.....	77
Help and documentation.....	78
6.1. Exit Survey.....	79
6.2. Other materials used.....	81
6.3. Reflective Journals.....	82
6.4. Testing Informed Consent Form.....	89
6.5. Focus Group Consent Form.....	90

Executive Summary

The number of individuals who suffer from celiacs disease in Ireland have a hard time adapting to their disease when first diagnosed. It is estimated that there are 50,000 people living in Ireland with celiacs disease and there is a further 400,000 who are gluten intolerant. (Coeliac Society Of Ireland, 2021)

More broadly worldwide the amount of celiacs being diagnosed in recent years has increased by 7.5%. It is estimated that 1.4% of the worldwide population spanning South America, North America, Africa, Asia, and Europe are affected by celiacs disease. When converted from percentages to population size this means 11 million of the world's 7.6 billion population size has the disease. (The Institute for Functional Medicine, 2021)

Celiacs have no clear-cut rules or guidelines to help them when eating out or shopping for themselves or family. Some food products packing has a clear and easy to see gluten free logo or stamp. However, many food brands are gluten free but are missing the stamp. Or some food products one year could be gluten free, but the following year contain gluten. This makes the process of shopping even harder for celiacs.

In 2020, many people of all age demographics have access to or personally own an Android device. With this I hope to create a solution to the problem many celiacs face daily using an Android Application.

Through using machine learning and the use of a natural language processor I want to offer a food packet scanner that can scan the ingredients section of packet which is not always easy to comprehend because of small text, or because certain ingredient names are puzzling to most people.

This captured information would then be passed into the natural language processor and this will make the data more useful to the end user by highlighting or flagging dangerous ingredients that are toxic to celiacs.

In addition to the packet scanner, I want to include a restaurant review system that will allow my users create new reviews, read existing reviews, update an existing owned review, and delate an existing owned review.

1.0 Introduction

1.1. Background

Celiacs disease is a digestive disorder that prevents individuals from consuming gluten which is a protein found in wheat, barley, rye, or oats. Any celiacs that consume gluten, once digested will feel immediately sick and can suffer from an array of symptoms ranging from bloating, abdominal pain, weight loss, malnutrition, or stunted growth. (Health Service Executive, 2011)

Celiacs disease is not directly known to be life threatening but if celiacs ignore their diet they can become anaemic, develop, or worsen osteoporosis which is the weakening of the bones or can even become infertile be the individual male or female. (Health Service Executive, 2011)

The idea of this project is quite a personal one. My own mother and grandmother have celiacs disease and ever since they were diagnosed with it 10 years ago the weekly shopping list and where they can eat out has been impacted massively.

When shopping, an individual newly diagnosed with celiacs disease typically will go to the gluten free section of a supermarket which from personal experience is 1 to 2 shelves of food which does not give a lot of choice. This fact alone can make a newly diagnosed celiac feel stressed and, in my opinion, leads to a lot of celiacs breaking their diet out of confusion as to what they can and cannot eat outside of the gluten free isle.

To rectify this, I have developed GFree which has an included packet scanner designed to be quick, effective, and clearly communicate what is safe, potentially dangerous, and dangerous to consume for celiacs. All the users must do is open the packet scanner and take a clear picture of the ingredients label, all the captured ingredients will be processed, and a list of ranked ingredients will be given back to the user, again following a colour coded safe, potentially dangerous, and dangerous rating.

As for restaurants and eating out, my mother has visited many restaurants over the years and quite a large margin of them do not cater for celiacs fully. Many restaurants that do, offer a tiny menu of "safe foods" which are dishes that are considered to be safe by restaurants but often they will be quite bland dishes like salads for example. I hope that my application GFree could change how celiacs view shopping and eating out.

1.2. Aims

The objective of my project GFree is to give individuals diagnosed with celiacs disease an application that they can use when wanting to shop, cook, or eat out with their celiac diet in mind.

A good example of the useful functionality offered by GFree is the packet scanner which will allow users to scan the side of food products ingredient labels to see if the food product contains ingredients that are toxic to celiacs. This like outlined earlier will follow a colour coded traffic light system with safe found ingredients in green, potentially dangerous found ingredients in yellow and toxic found ingredients in red.

GFree will also contain a review system for restaurants that will be compiled of reviews created by GFree users who ideally would celiacs themselves.

With this review system my user base will be able to tell immediately if a restaurant caters for celiacs disease or if it is best avoided.

The GFree application ideally could be used to see the extent a given restaurant caters to the disease, many restaurants boast they cater for celiacs disease, but some do not offer the full menu. With my application I want to make my users informed.

1.3. Technology

1.3.1. Android Studio

Android Studio is the integrated development environment used for Google's Android mobile operating system.

This is the software used to create Android mobile applications and can be considered the foundation of my project and any other Android application. Android Studio allows me to create code, debug errors, design unit tests and measure performance.

1.3.2. Firebase

Firebase is a database solution offered by Google. Databases are used to store information that needs to be saved or remembered like user's registration info or user created reviews.

Firebase is a cloud-based NoSQL database that allows for real time data synchronisation. This means if a user updates anything in the database it is instantly shown to other users of the application.

1.3.3. Java

Java is my programming language of choice which will be used to create my Android application, Kotlin as well as Java is a language supported by Android Studio however, I am more familiar with Java and as such Java was the obvious choice as it is backed by 6 years of knowledge.

1.3.4. Google Machine Learning (Text Recognition)

Google Machine Learning's Text Recognition API enables me as a developer to capture and process an image for text.

When tailored to my application GFree, this is what was used to capture ingredient labels. The processed packet information can then be used programmatically which in essence allows me as a developer to make the scanned information more useful or meaningful than the raw ingredient label that was scanned.

In recent months Google Machine Learning has separated from Firebase as they were once a single service.

Google Machine Learning still allows for an online based text recognition API but is now a metered, charged service.

The solution for this is to use their free offline version which must be packaged with the application therefore increasing the overall size of my app, but this side effect of using the service for free.

1.3.5. JUnit 5

JUnit 5 is a unit testing framework that is programmed in Java. Android Studio directly supports JUnit unit tests which allows me to test my applications functionality programmatically. This means I can test my application thoroughly and quickly once the tests are written. This type of automated testing can also remove potential human error if I were to manually test everything.

JUnit will allow me to test navigation, interaction with buttons and input fields as well as edge cases like "password to short" for example.

1.3.6. Espresso (UI-Testing)

Espresso is the testing suite that is included by default with Android Studio. Espresso is specifically a user interface testing program that allows me to create tests that mimic a user's interaction with the graphical user interface.

I personally have a number of tests that login, create reviews, read reviews, update reviews and delete reviews.

1.3.7. GitHub

GitHub is my version control of choice. GitHub allows me to track any changes I have made by remembering the history of every single file in my application. GitHub allows for the creation branches which supports a divide and conquer approach to development which is an excellent way of breaking a large project into smaller easier to work with tasks.

This naturally makes certain areas of the application less cumbersome and allows me to approach this in an flexible Agile like approach.

1.4. Structure

The layout of this document is as follows:

- 1.0: Reasons for creating the project and overview of technology used.
- 2.0 Requirements for the project both Functional and Non-Functional along with a visualisation of user's interaction with the system conveyed with aid of use case diagrams.
- 3.0 The conclusion of the project, both the positives and negatives of the finished product.
- 4.0 What I would do differently if more time were available.
- 5.0 References used in the document to improve the quality of statements.
- 6.0 Supplementary external documents that aid this one.

2.0 System

2.1. Requirements

I took the approach of creating a focus group of my identified stakeholders which were comprised of family members and friends who are affected by celiacs disease. The participants were given a consent form to fill in [see appendix 7.5](#). The signed versions of this template have been stored securely using a password/passphrase for protection. I tried to mimic a developer to stakeholder like relationship throughout the session the best I could due to Covid 19.

My method of eliciting the requirements was to sit down with my stakeholders and discuss the GFree project idea and some design choices such as the:

- Application layout.
- Accessibility / Ease of navigation.
- Functionality offered and its potential uses.

Using stakeholders who are affected by celiacs disease and showing them early prototypes such as wireframes was a massive benefit as this allowed me to get ideas and feedback directly from potential end users once my application is completed. Check section [2.4 Graphical User Interface](#) to see the wireframes distributed during the focus group.

From my focus group I discovered many insights into what my potential end users would like to see and surprisingly many of the needs and wants were of an aesthetic nature and so through generations of prototypes I found out that the application needs to be easy to use and navigate as many of my end users stated they would not use the application in a supermarket if it was a burden to use. As such I factored this feedback directly into my functional and non-functional requirements.

With ease of navigation now a priority, users should be able to understand the layout and structure of the application after their initial usage. It should take users 2-5 minutes to create an account and a further 5-10 minutes to run through all the features offered. Users should be able to use all system functionality after a total of 15 minutes running through the application as it is now designed with ease of navigation as a requirement.

2.1.1. Functional Requirements

1) Login and Register System.

A fully implemented and working database that is incorporated into GFree that allows for the reading and writing of information to and from the database.

The purpose of this database is to hold all user's login and registration information as well as restaurant reviews.

Google Firebase is the database of choice and is used to hold all information generated by the application.

The reason for having accounts at all is to link user accounts to reviews. Registration is optional and is not required if the user wishes to only use the packet scanner. **Highest**

2) Camera Packet Scanner.

A packet scanner activity is needed for GFree to scan food product ingredient labels. This will require an Android device with a functional camera. From this users will be able to scan for safe, potentially dangerous, and dangerous ingredients. Google's Machine Learning text recognition package will be the service used for this. **Highest**

3) Natural Language Processor (NLP)

A natural language processor will be necessary to process the raw text data generated from a user's image of an ingredient label. The ingredient label image data will be passed into the natural language processor to create more useful and concise information. **High**

4) Review System for Restaurants.

A user-based review system of restaurants that records the name, description and rating given by the user when reviewing a restaurant is a requirement. This requirement should allow users to create, read, update, and delete reviews which will allow users to easily find restaurants that are cater for celiacs disease. **High**

2.1.2. Non-Functional Requirements

1) Packet Scanner Performance

Performance of the packet scanner is an important factor of the application that must be considered as it is an important part of the project.

The packet scanner must not cause the camera viewport to lag and stutter otherwise the feature will go unused.

2) Fluid Interface

The interface needs to be easy to understand and navigate. The use of text in buttons alongside icons could help with navigating the application. The application should be learned after 5-15 minutes of use.

3) Availability

The availability of my application is important, the use of external services like Firebase which supports horizontal scaling and use of a local version of Google Machine Learning make my application solid when considering what could fail.

4) Scalability

The Scalability of my application when underload is an important factor to consider when designing an application to be used as a tool like GFree. The requests and responses will be from the login and logout activities as well as the restaurant reviews.

The use of Firebase here allows for horizontal scalability that will expand the number of servers used to accommodate the demand put on my applications services.

The likeliness of my applications online based services being bombarded is slim because of their nature. Not every user will be leaving a review at the same time for example and a user typically only registers and logs in once.

2.1.3. Use Case Diagrams

2.1.3.1. Requirement 1: User scans packet

2.1.3.2. Description & Priority

The User can scan packets to get information about their ingredients. This is a desired feature and therefore is high priority.

2.1.3.3. Use Case

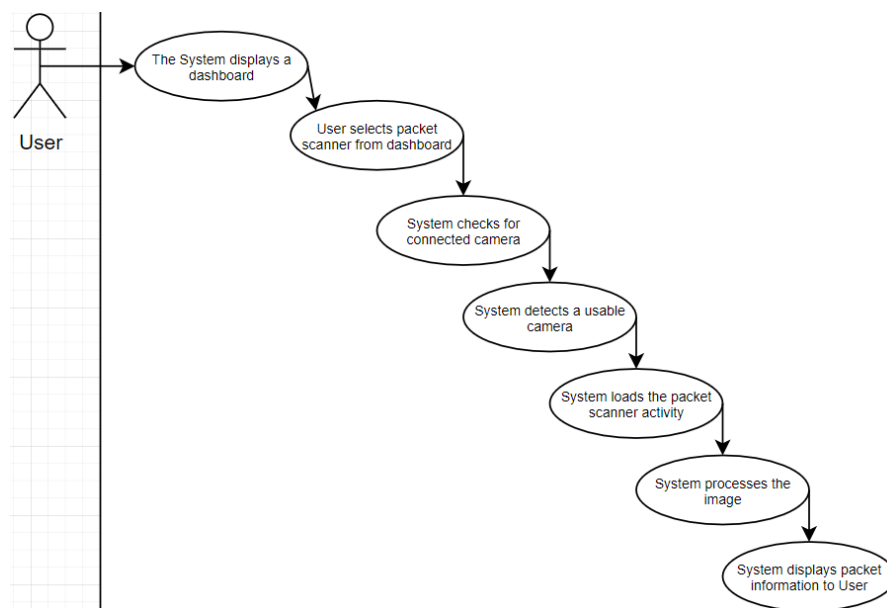
Scope

The scope of this use case is the interaction with the packet scanner of the application.

Description

This use case describes the scanning of packets during the Users use.

Use Case Diagram



Flow Description

Precondition

The application is installed on an Android device, the application is launched, the user has an account registered and is logged in.

Activation

This use case starts when a User taps the packet scanner from the application dashboard.

Main flow

1. The System displays a dashboard.
2. The User selects the packet scanner from the dashboard.
3. The System checks for a connected camera.
4. The System detects a usable camera. (See E1)
5. The System loads the packet scanner activity.
6. The User takes a photo of the packet they wish to scan.
7. The System processes the image. (See E2)
8. The System displays packet contents to the User.

Alternate flow

-

Exceptional flow

E1 : No camera found.

1. The System notifies the User that no camera was found.
2. The User is returned to the dashboard by the System.
3. The use case continues at position 6 of the main flow

E2 : Image too blurry/Unreadable.

1. The System notifies the User that the image is unusable.
2. The use case continues at position 1 of the main flow

Termination

The system presents the user with packet information.

Post condition

The System enters a wait state while showing the User packet information.

2.1.3.4. Requirement 2: User reviews

2.1.3.5. Description & Priority

The User can create, read, update, and delete their restaurant reviews. This is a core feature and should be the highest priority.

2.1.3.6. Use Case

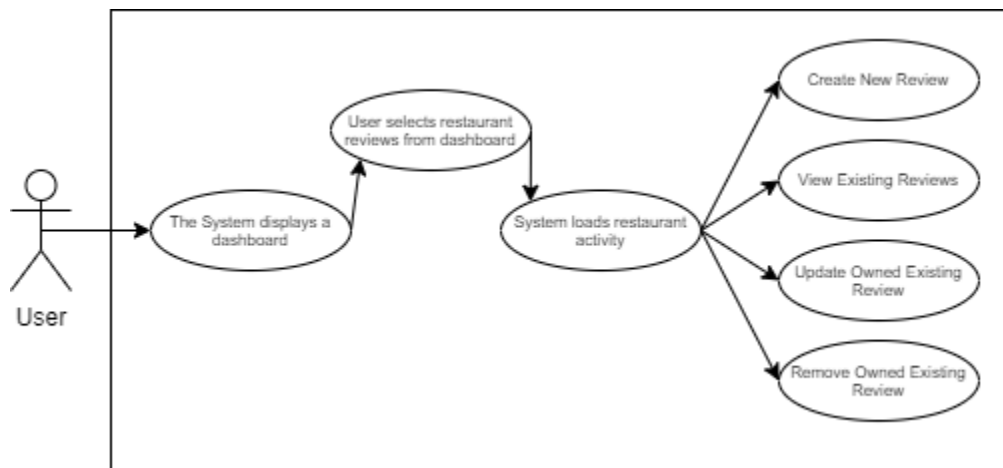
Scope

The scope of this use case is the interaction with the restaurant review section of the application.

Description

This use case describes the viewing of and/or manipulation of user reviews.

Use Case Diagram



Flow Description

Precondition

The application is installed on an Android device, the application is launched, the user has an account registered and is logged in.

Activation

This use case starts when a User taps the restaurant reviews from the application dashboard.

Main flow

1. The System displays a dashboard.
2. User selects restaurant reviews from dashboard.
3. The User is prompted to create a new review, read existing reviews, update an owned existing review, or delete an owned existing review.
4. The User selects create new review (See A1, A2, A3)
5. A window is displayed asking for the rating and review body text.
6. The User enters their review and rating.

7. The User selects submit review.
8. The review is submitted.
9. The User is notified of success.
10. The User is returned to the dashboard.

Alternate flow

A1 : Read reviews

1. The User selects read existing reviews.
2. All reviews are displayed.

A2 : Update existing review

1. The User selects update an existing review.
2. The rating and review text body is prefilled with the existing review.
3. The User edits the review body and/or rating.
4. The use case continues at position 7 of the main flow.

A3 : Delete existing review

1. The User selects delete an existing review.
2. A list of the User review is displayed.
3. The User selects the review of their choice.
4. The User selects delete.
5. The use case continues at position 9 of the main flow.

Exceptional flow

-

Termination

The system sends or receives a review.

Post condition

The System enters a wait state.

List further functional requirements here, using the same structure as for Requirement1.

2.1.4. Data Requirements

Users generated/created data in the form of user reviews will be stored on a Firebase database and will be linked to users using their email address.

Scanned images using the application.

2.1.5. User Requirements

Users must be connected to the internet via Wi-Fi or mobile data to use the application as a mandatory register and login is required to use the applications features.

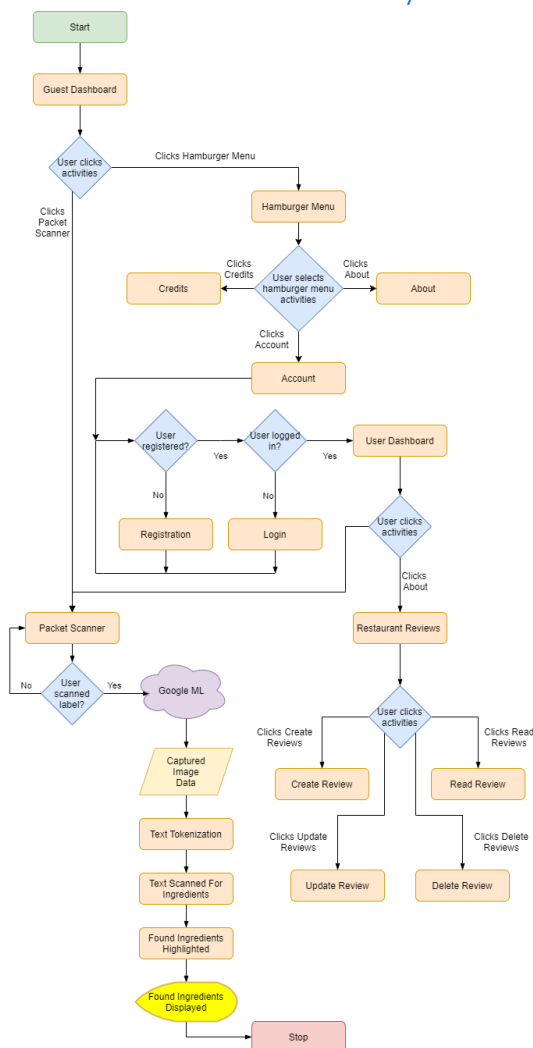
2.1.6. Environmental Requirements

2.1.7. Usability Requirements

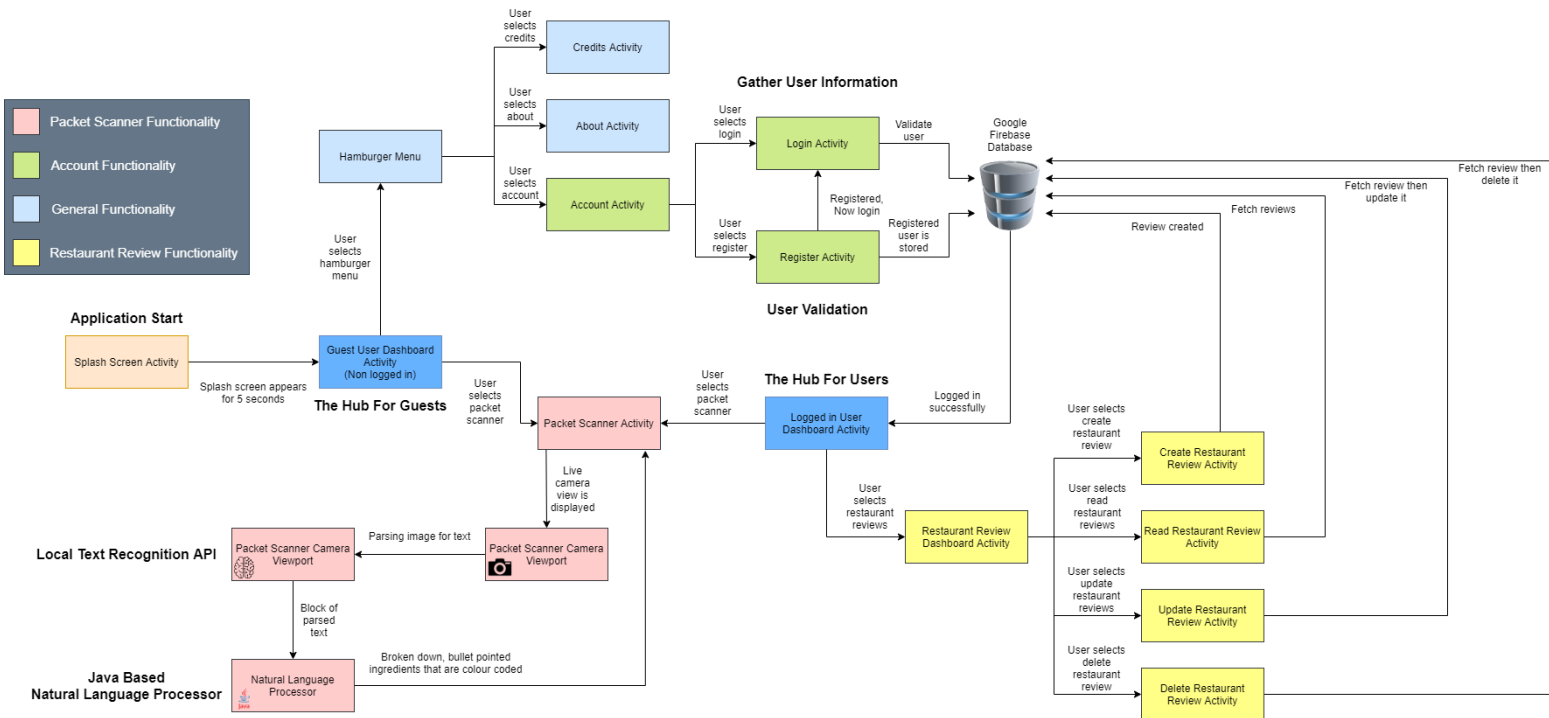
The interface must be easy to navigate and understand. Aesthetically pleasing colours must be used and must be easy to look at and read information from.

2.2. Design & Architecture

2.2.1. Android Activity Flow Diagram



2.2.2. Android Activity Structure Diagram



1.1. Implementation

1.1.1. Image Text Converter

The image text converter class is my programmed natural language processor (NLP) and is where my bag of words exists. Every word detected by the packet scanner will be compared against this classes 3 lists. The safe, potentially dangerous, and dangerous lists.

```
public void setDangerousIngredients() {
    dangerousIngredients.add("Gluten");
    dangerousIngredients.add("Corn"); //TEST
    dangerousIngredients.add("Barley Malt Extract");
    dangerousIngredients.add("Brewers Yeast");
    dangerousIngredients.add("Buckwheat");
    dangerousIngredients.add("Dextrin");
    dangerousIngredients.add("Ethanol");
    dangerousIngredients.add("Spelt");
    dangerousIngredients.add("Wheat");
    dangerousIngredients.add("Wheat Protein");
    dangerousIngredients.add("Hydrolyzed - Wheat Protein");
    dangerousIngredients.add("Malt");
    dangerousIngredients.add("Rye");
    dangerousIngredients.add("Guar Gum");
    dangerousIngredients.add("Seitan");
    dangerousIngredients.add("Soy Sauce");
    dangerousIngredients.add("Malt Vinegar");
    dangerousIngredients.add("Barley");
}

public void setWarningIngredients() {
    warningIngredients.add("Soya");
    warningIngredients.add("Oats");
    warningIngredients.add("Soy");
}

public void setSafeIngredients() {
    safeIngredients.add("Lecithin");
    safeIngredients.add("Maltodextrin"); //TEST
    safeIngredients.add("Millet");
    //safeIngredients.add("Soy");
    safeIngredients.add("Silicon");
    safeIngredients.add("Starch");
    safeIngredients.add("Tapioca");
    safeIngredients.add("Xanthan Gum");
    safeIngredients.add("Modified Food Starch");
    safeIngredients.add("Gluten Free");
}
```

This is an image of the 3 lists. Ingredients can be easily added to any of the lists at any time. The formatting of each ingredient does not matter as the casing is ignored, for example Malt Vinegar is the same as malt vinegar to the NLP.

1.1.2. Packet Scanner Activity

An important activity of my application is the Packet Scanner activity. Its purpose is to scan user captured images for text and upon finding text return it to the natural language processor.

This section of my project posed a significant challenge and completing it marked a large milestone in my project's progression. One of the most challenging issues I faced was how the data generated from the captured image was handled and how it could be programmatically used in the natural language processor.

A StringBuilder is what the Google ML text recognition outputs and this was the first time I have worked with this class and its methods.

```
public void receiveDetections(@NonNull Detector.Detections<TextBlock> detections) {
    captureScanButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            SparseArray<TextBlock> sparseArray = detections.getDetectedItems(); //Get array of items detected
            StringBuilder stringBuilder = new StringBuilder();

            for (int i=0; i<sparseArray.size(); i++){
                TextBlock textBlock = sparseArray.valueAt(i);
                if(textBlock != null && textBlock.getValue() != null){
                    stringBuilder.append(textBlock.getValue() + " "); //Convert what was detected into a StringBuilder.
                }
            }

            final String stringText = stringBuilder.toString();
            Handler handler = new Handler(Looper.getMainLooper());
            handler.post(new Runnable() {
                @Override
                public void run() {
                    stringScannedImageResult = stringText;
                    StringTokenizer st = new StringTokenizer(stringScannedImageResult.toLowerCase(), " ,");

                    String returnedIngredient = null;
                    while (st.hasMoreElements()){
                        String CurrentWordToCheck = st.nextToken();
                        try {
                            returnedIngredient = imgConverter.checkDangerousIngredients(CurrentWordToCheck, st.countTokens());
                            if ((!returnedIngredient.equalsIgnoreCase("anotherString: "")) || (!returnedIngredient.equalsIgnoreCase("anotherString: null"))){
                                scannedImageDangerousIngredients.add(returnedIngredient);
                            }
                            returnedIngredient = imgConverter.checkWarningIngredients(CurrentWordToCheck, st.countTokens());
                            if ((!returnedIngredient.equalsIgnoreCase("anotherString: "")) || (!returnedIngredient.equalsIgnoreCase("anotherString: null"))){
                                scannedImageWarningIngredients.add(returnedIngredient);
                            }
                            returnedIngredient = imgConverter.checkSafeIngredients(CurrentWordToCheck, st.countTokens());
                            if ((!returnedIngredient.equalsIgnoreCase("anotherString: "")) || (!returnedIngredient.equalsIgnoreCase("anotherString: null"))){
                                scannedImageSafeIngredients.add(returnedIngredient);
                            }
                        } catch (Exception e){
                            System.out.println(e);
                        }
                    }
                }
            });
        }
    });
}
```

This image shows the receiveDetections method which is run when the text recognition from the Google Machine Learning Kit detects words. You can see my comments and view how the array of detected words is converted to a StringBuilder and from there they are checked using the Image Text Converter class called imgConverter in this instance. The imgConverter is my NLP class and this is how it was incorporated into the Packet Scanner activity after much trial and error.

1.1.3. Register Activity

The registration activity is used to create users, users are required to link accounts to reviews. The registration has front end validation for things like the password policy, so users need a password over 7 characters in length.

The registration activity is used to create users for GFree. Accounts and a login system exist to begin with to allow user to create reviews as reviews behind the scenes in the backend are linked to email addresses.

The frontend user interface that is displayed to users has validation and prevents duplicate accounts from registering with the same email and has validation the prevents weak or empty password field etc.

```
private void registerUser(String email, String pass) {
    auth.createUserWithEmailAndPassword(email, pass).addOnCompleteListener( activity: RegisterActivity.this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            if (task.isSuccessful()){
                Toast.makeText( context: RegisterActivity.this, text: "Registration Completed!", Toast.LENGTH_SHORT).show();
                startActivity(new Intent( packageContext: RegisterActivity.this, LoginActivity.class));
                finish();
            }
            else if (!task.isSuccessful()){
                try
                {
                    throw task.getException();
                }
                // if user enters wrong email.
                catch (FirebaseAuthWeakPasswordException weakPassword)
                {
                    Log.d( tag: "Registration Check", msg: "onComplete: weak_password");
                    Toast.makeText( context: RegisterActivity.this, text: "Weak Password!", Toast.LENGTH_SHORT).show();
                    // TODO: take your actions!
                }
                // if user enters wrong password.
                catch (FirebaseAuthInvalidCredentialsException malformedEmail)
                {
                    Log.d( tag: "Registration Check", msg: "onComplete: malformed_email");
                    Toast.makeText( context: RegisterActivity.this, text: "Email is malformed! Double check it.", Toast.LENGTH_SHORT).show();
                    // TODO: Take your action
                }
                catch (FirebaseAuthUserCollisionException existEmail)
                {
                    Log.d( tag: "Registration Check", msg: "onComplete: exist_email");
                    Toast.makeText( context: RegisterActivity.this, text: "Email is already linked to an account!", Toast.LENGTH_SHORT).show();
                    // TODO: Take your action
                }
                catch (Exception e)
                {
                    Log.d( tag: "Registration Check", msg: "onComplete: " + e.getMessage());
                }
            }
            else {
                Toast.makeText( context: RegisterActivity.this, text: "Registration Failed!", Toast.LENGTH_SHORT).show();
            }
        }
    });
}
```

Visible here is the validation for the registration form. Inside the try catch statements we catch all the exceptions raised by Firebase should an error occur. The first exception is for a weak password, the second is for a malformed email address and the third is for if the email is already in use.

1.1.4. Login Activity

The login activity is used to authenticate users and grant them access to the logged in dashboard, users are required to login to utilize the restaurant review section of the application.

The login process much like registration has validation and notifies when the email or password entered is incorrect.

```
private void loginUser(String email, String pass) {
    auth.signInWithEmailAndPassword(email, pass).addOnCompleteListener( activity: this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            if (task.isSuccessful()){
                Toast.makeText( context: LoginActivity.this, text: "Logged in Successfully", Toast.LENGTH_SHORT).show();
                startActivity(new Intent( packageContext: LoginActivity.this, UserDashboardActivity.class));
                finish();
            } else {
                //if login faild
                try
                {
                    throw task.getException();
                }
                // if user enters wrong email.
                catch (FirebaseAuthInvalidUserException invalidEmail)
                {
                    Log.d( tag: "Login Check", msg: "onComplete: invalid_email");
                    Toast.makeText( context: LoginActivity.this, text: "Email is incorrect", Toast.LENGTH_SHORT).show();
                    // TODO: take your actions!
                }
                // if user enters wrong password.
                catch (FirebaseAuthInvalidCredentialsException wrongPassword)
                {
                    Log.d( tag: "Login Check", msg: "onComplete: wrong_password");
                    Toast.makeText( context: LoginActivity.this, text: "Password is incorrect", Toast.LENGTH_SHORT).show();
                    // TODO: Take your action
                }
                catch (Exception e)
                {
                    Log.d( tag: "Login Check", msg: "onComplete: " + e.getMessage());
                }
            }
        }
    });
}
```

The validation for the login section here is a try catch statement that check for exceptions thrown by Firebase. These allow for the creation of informative error messages for users during the login process should errors occur.

1.1.5. Natural Language Processor (Packet Scanner)

For the packet scanner inside GFree a natural language processor (NLP) was created. The NLP effectively converts the raw input data created from Google's Machine Learning text recognition API into a usable ranked list of ingredients. These ranks are as follows:

- Safe
- Potentially Dangerous
- Dangerous

The NLP was developed from scratch following a bag of words approach which effectively means all the text data scanned by Google's text recognition API is compared against a predefined list of ingredients that I have created. If any scanned words match anything in my lists, then add them to the found ingredient list which will be outputted to the user in the end.

The bag of words model is named as such because the structure and order of words is ignored the model is only interested in a word's occurrence and not its position in a sentence hence it is like having words in a bag. (Brownlee, 2017)

The ingredient lists were comprised by myself by doing research on ingredients that are dangerous for celiacs and by talking with my mother who is a diagnosed celiac who has knowledge about what ingredients to avoid.

I initially started by talking with my mother about dangerous ingredients and comprising a list during the conversation. All the ingredients she mentioned and knows of were added to a paper list.

~~Gluten~~ is a protein found in wheat barley & rye
 NOT

(Sulphites) Beer Alcohol
 Brewers yeast ~~X~~ TO TRUCK in BEERS
 Buckwheat - ~~X~~ pure buck
 Dextrin - ~~X~~ made from Corn-ethanol - ~~X~~ Comes from wheat
 Spelt - ~~X~~ distilled -
 hydrolyzed wheat protein ~~X~~
 Sourdough - not Gluten free
 Malt = made Barley not GF ~~X~~
 Oats = Be Careful if processed it eliminates cross-contamination
 Rye ~~X~~ Contains and ~~is~~ Contains Gluten
 (Guar Gum) is Thinking Ingredient
 Large quantities would work like a Laxative
 Seitan Ingredient - ~~X~~
 (Soy Sauce - ~~X~~ (variants)
 Teriyaki Sauce ~~X~~
 Malt Vinegar not distilled ~~X~~
 X Wheat - A grain containing Barley - durum - emmer, farin, Graham, Kamut - Semolina - SPELT -

Figure 1 Initial list of dangerous ingredients.

good

Lecithin - soy is GF. food thickener ✓
 ? Maltodextrin - is GF ✓ if ~~X~~ wheat is used it will taste
 Millet - A grain ✓
 Soy - ✓ GF ✓
 Silicon - dioxide - (found in Spices) ✓
 Starch. = Means Corn GF ✓
 Tapioca. is GF. used to mix through other flours ✓
 Xanthan gum ✓ used for Baking cakes Replaces Gelatin
 Modified ~~Flour~~ Food Starch. is GF. However - if ~~was~~ made with To make it. ~~X~~ No GF

Figure 2 Initial list of safe ingredients.

To supplement my initial basic ingredient list and add more words to my bag of words I began to use external online resources to increase the size and simultaneously the accuracy of GFree's packet scanner as the more ingredients in my ingredient list the more accurate GFree becomes. This was done by using an excellent online resource Gluten-Free living which has an entire index on ingredients with what is safe to consume and what is not. These were directly added to my bag of words. (Gluten-Free Living, 2016)

My initial idea for GFree's packet scanner and its natural language processor was to have a safe and dangerous list of ingredients. However, from speaking with my mother I learned that there is another dynamic to ingredients and that is how they are prepped.

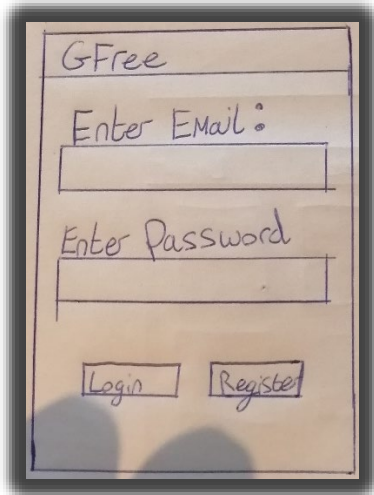
Some ingredients are safe when prepared a certain way and completely unsafe when prepared in another. Learning this information is what gave rise to the potentially dangerous ingredient list. Any ingredients that can have a safe and dangerous version of itself depending on how the ingredient was prepared will be inside the potentially dangerous section which in essence tells the user to use their own discretion.

A good example of this is checking the Food Safety Authority Ireland (FSAI) guidelines around gluten free declarations. This guideline documentation designed for businesses who serve food indicate that oats depending on how they are produced, prepared and or processed can alter if they are safe for celiac consumption. (Food Safety Authority of Ireland, 2017)

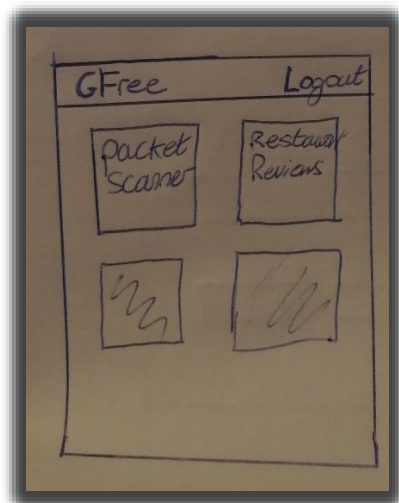
1.2. Graphical User Interface (GUI)

Low Fidelity

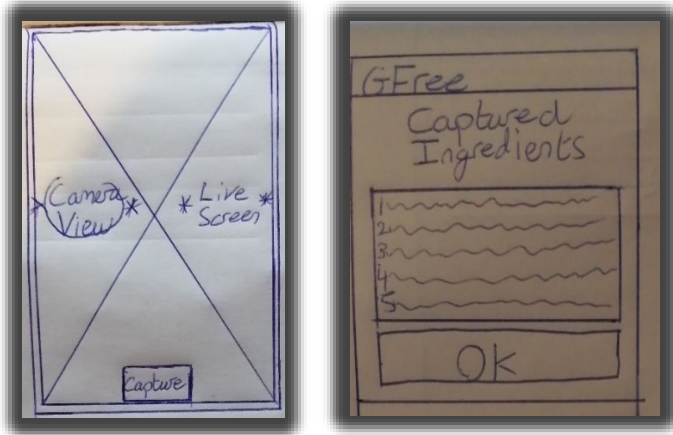
Before any UI was designed, I began by drawing UI drafts with paper these are really early low fidelity prototypes.



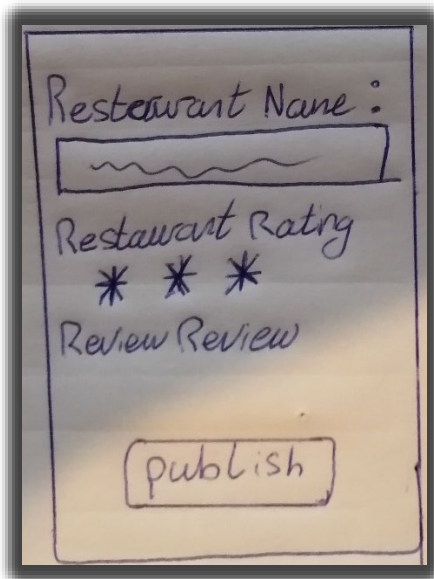
This was my initial login and register page idea with a simple input form and two buttons.



This is my initial dashboard design. I intended to have the logout button up the top right and have my functionality offered in a column and row grid layout.

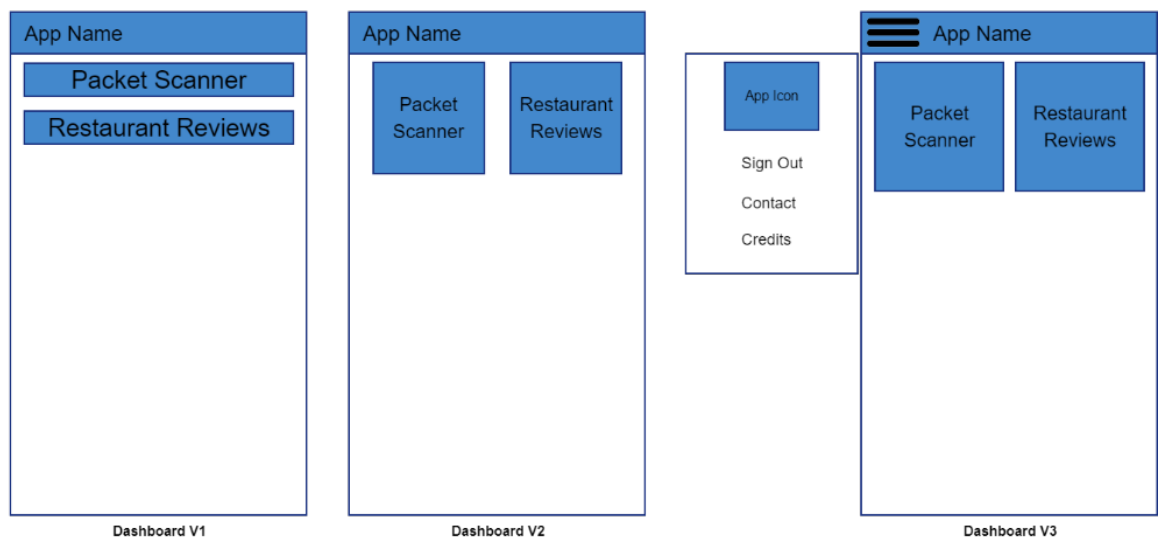


The concept for how my packet scanner ingredient screen will be presented to the user as well as the live camera view used for capturing the image.



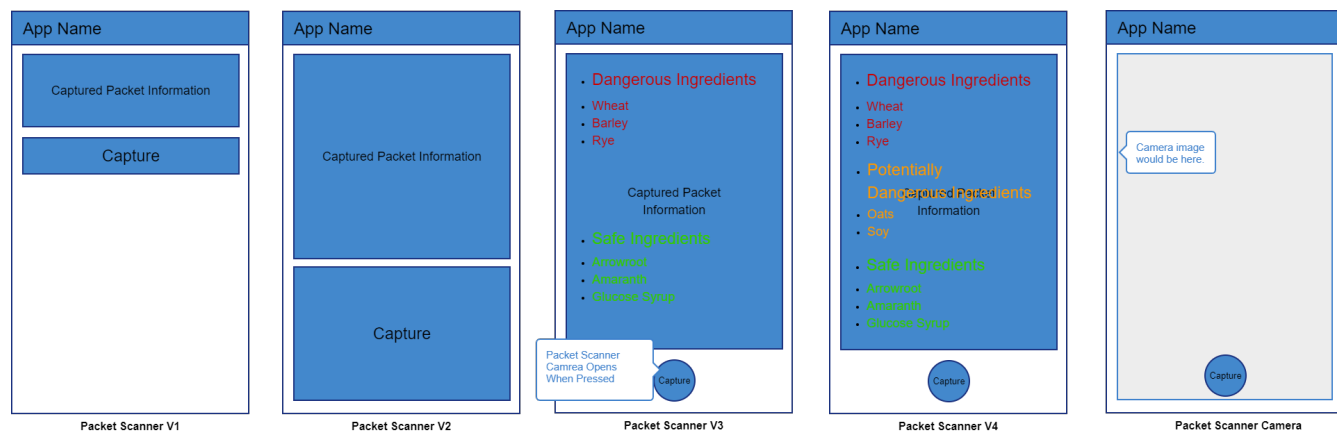
The restaurant review creation screen initially was designed with the restaurant name, description and rating followed by the publish review button.

Wireframes



← Evolution →

The evolution of the dashboard GUI wireframe. Each evolution was constructed with feedback from my focus group. This dashboard will be accessible to end users once logged in.



← Evolution →

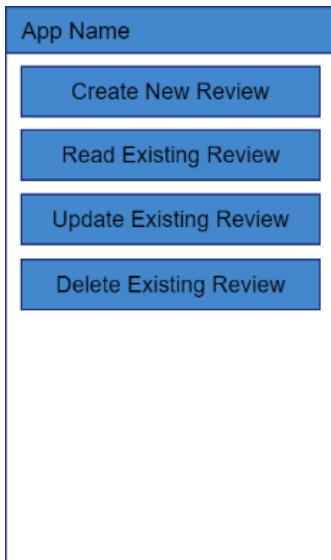
The evolution of the packet scanner GUI wireframe. Excluding the obvious moving of panes, changes to the design such as the round capture button and expanding the captured packet information area was a result of conducting my focus group.

This activity/screen is where the user will see the result of their scanned packet.



Packet Scanner V5 (Final)

The final prototype of the results page of the packet scanner should be displayed to the user in a list format. The dangerous ingredients should be red, potentially dangerous ingredients should be orange, and safe ingredients should be green.



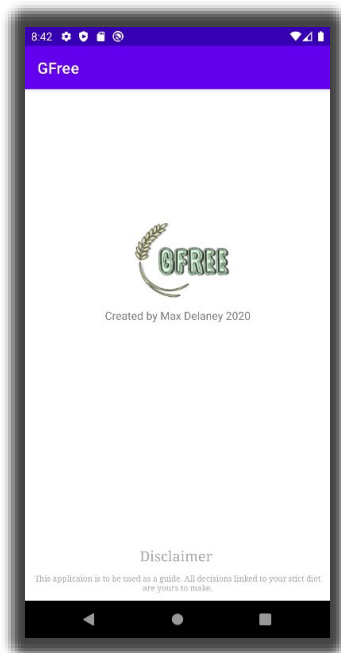
Restaurant Review V1



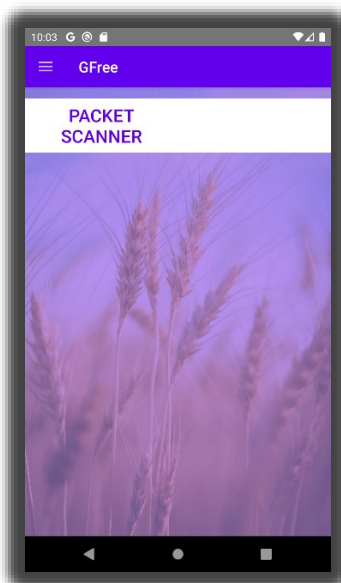
Restaurant Review V2

This is the evolution of the restaurant review section wireframes. My initial design was to have a simple list of buttons with CRUD functionality but after showing it to my focus group it was decided that a grid like tabular layout would be better suited. I could also potentially use icons here instead of sentences.

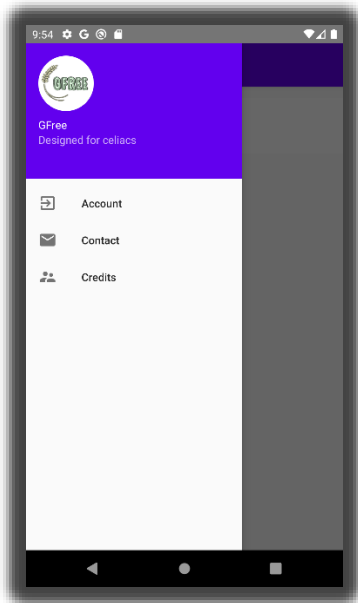
High Fidelity



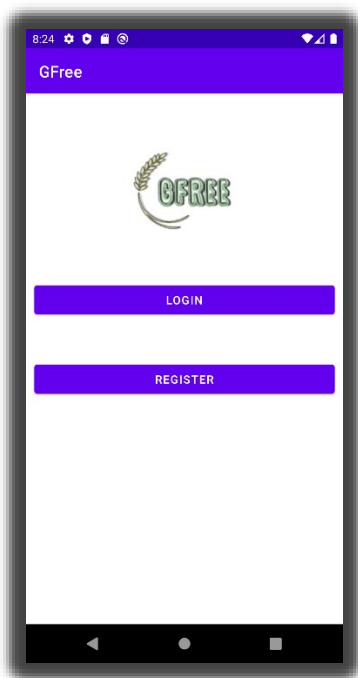
The splash screen activity that is displayed to the user on application start. This is visible for 5 seconds when the application is started. Visible on this screen is the application logo, creator, and a disclaimer.



The non-logged in version of the application dashboard is presented to make registration optional instead of being mandatory. The restaurant review functionality is absent from the menu here as you are required to have an account and be logged in before it becomes visible. This is done because reviews are linked to accounts in the backend.



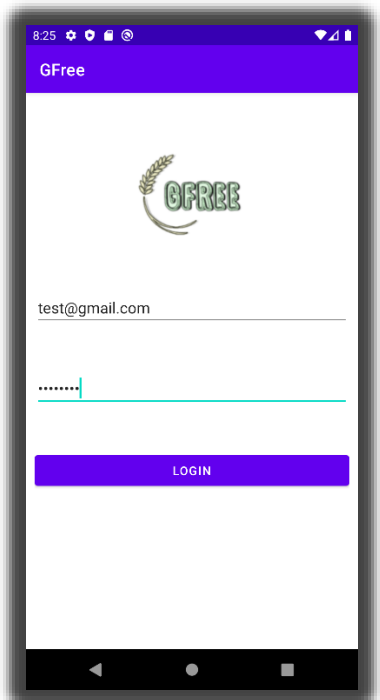
The option to login and register is present in the applications hamburger menu under the heading of Account. Users only need to login to access the restaurant reviews section of the application.



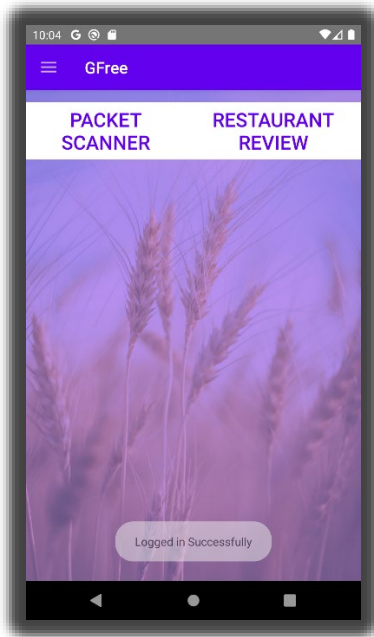
Once account is clicked, the user can decide to login to an existing account or register a new account.



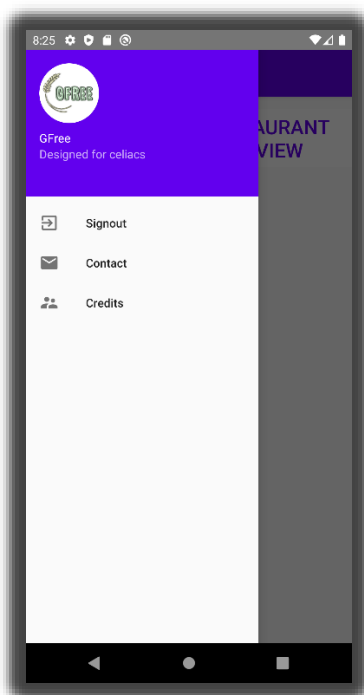
The registration activity is used to register a new user to GFree. Their email as well as their repeated password is required to confirm the password is correct.



The login activity is used to sign in a registered user.



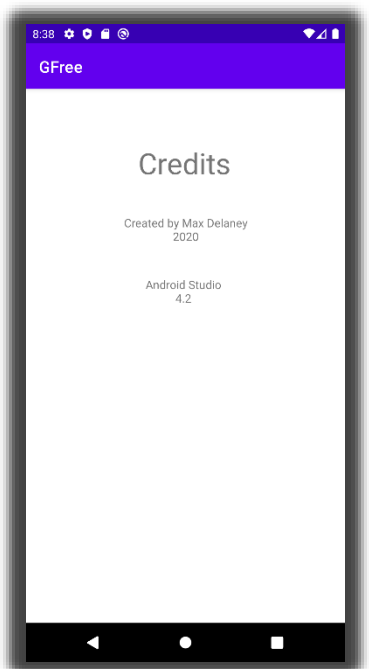
The logged in version of the application dashboard. This is the hub for logged in users where you can navigate to all application features. Visible is the Packet Scanner, Restaurant Review and Hamburger menu in the top left corner.



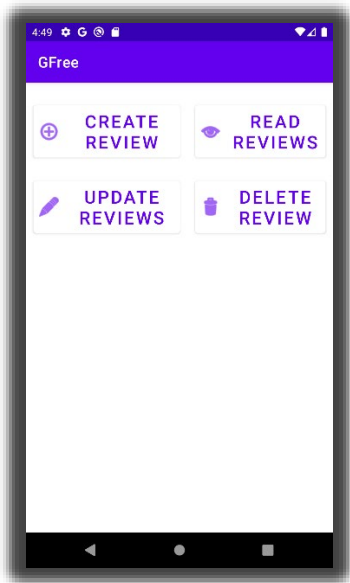
The hamburger menu when logged in shows the functionality to logout of your account.



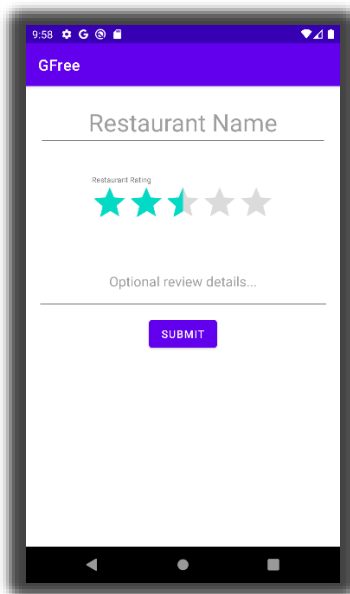
The contact information activity is a simple one designed to give users my email address to contact me on should the need arise. This is accessible through the hamburger menu.



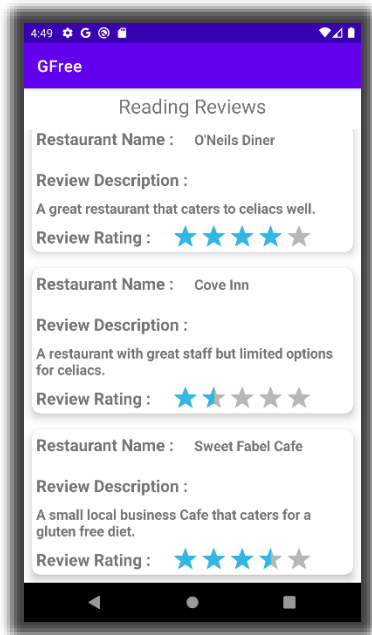
The application credits activity is also a simple one designed to give a look at credits should users want to see who developed and what resources were used in the making of the application.



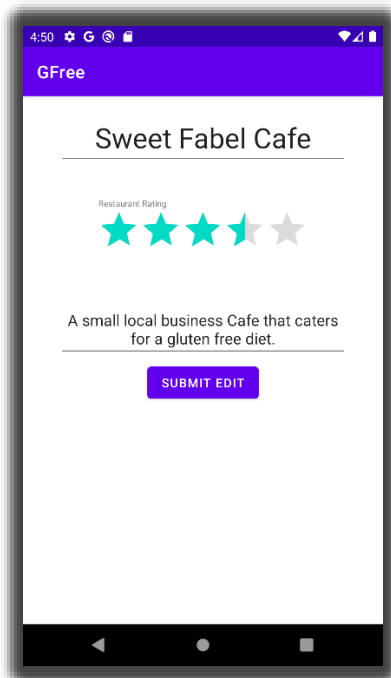
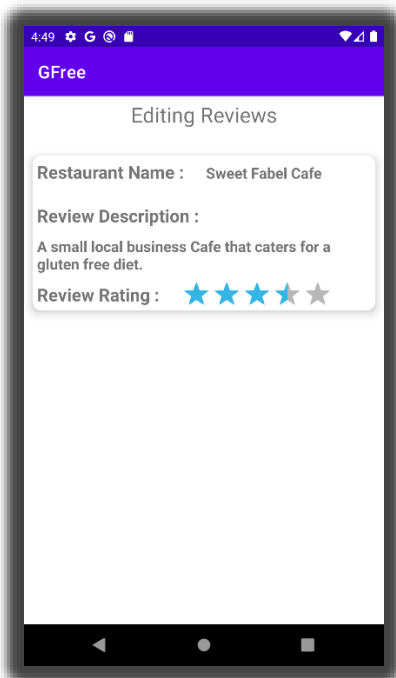
The review dashboard is where users can create a new review, read existing reviews, update an owned existing review, and delete an owned existing review.



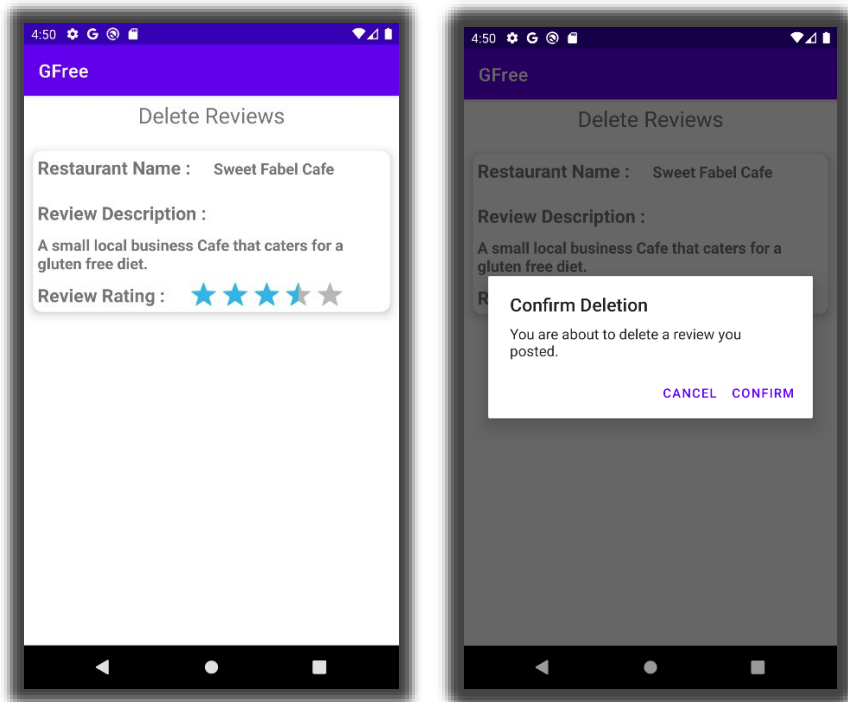
The create review activity screen is displayed to users who choose to create a new review. This activity requires a restaurant name, and star ranking which also contains an optional review body.



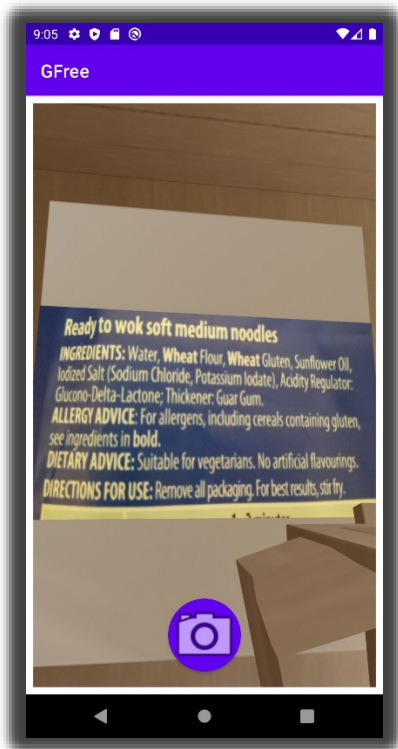
The read reviews activity shows all existing reviews in card views which are contained inside a recycler view. This shows the user all the existing reviews in a tabular column format that is easy to read and navigate.



The update or edit reviews activity shows the existing reviews that are created by the logged in user. Here a logged in user can edit any of their posted reviews and repost the edited version with the altered restaurant name, rating or review description.



The delete reviews activity shows the existing reviews that are created by the logged in user. Here a logged in user can delete any of their posted reviews permanently. A prompt to confirm the deletion is also used to safeguard any miss clicks.



This is the Packet Scanner activity camera viewport. This is the camera of the activity and how it gets the text of a packet. The captured image from here is passed into the Google ML integrated API, from there is then passed into my natural language processor and then returned to the scanned packet information box which is sorted by the severity of the ingredients.



This is the result of the scanned packet from above. My own implemented natural language processor processes the image data from the Google Machine Learning captured image which then compares it against a list of dangerous ingredients.

It then displays it in a formatted list as seen here. I have purposely followed a traffic light system as desired in the final version of my prototype/wireframe.

1.3. Testing

For my class testing I used JUnit to create unit tests. These use assertions or assumptions and allow me to set the expected state of a class instance. I currently have 4 unit tests for the creation, reading, updating, and deleting of reviews.

These unit tests test the class and asserts the expected values. They are designed to mimic what users will be doing when using the application.

This is an image to show the code for the unit test for the updating a user review and essentially is just updating the instance of the restaurant review class.

```
aurantReviewActivityTest.java x UpdateRestaurantReviewActivityTest.java x
@Test
public void onCreate() {

    System.out.println("ABC");

    //Create testing instance
    RestaurantReview review = new RestaurantReview();
    review.reviewId = "ABC";
    review.userEmail = "junittestemail@gmail.com";
    review.rating = 4.5F;
    review.restaurantName = "JUNIT Update Review Restaurant Name";
    review.reviewDescription = "JUNIT Update Review Restaurant Description";

    Log.d( tag: "Update Restaurant Review Initial Id", review.reviewId);
    Log.d( tag: "Update Restaurant Review Initial Email", review.userEmail);
    Log.d( tag: "Update Restaurant Review Initial Rating", String.valueOf(review.rating));
    Log.d( tag: "Update Restaurant Review Initial Restaurant Name", review.restaurantName);
    Log.d( tag: "Update Restaurant Review Initial Restaurant Desc", review.reviewDescription);

    //Assert/Check the created instance.
    assertEquals( expected: "ABC", review.reviewId); //Assert review id
    assertEquals( expected: "junittestemail@gmail.com", review.userEmail); //Assert review email
    assertEquals( expected: 4.5F, review.rating); //Assert review rating
    assertEquals( expected: "JUNIT Update Review Restaurant Name", review.restaurantName); //Assert review name
    assertEquals( expected: "JUNIT Update Review Restaurant Description", review.reviewDescription); //Assert review description

    //Update testing instance - mimicked update
    review.reviewId = "DEF";
    review.userEmail = "junittestemailupdated@gmail.com";
    review.rating = 2.5F;
    review.restaurantName = "JUNIT Update Review Restaurant Name Updated";
    review.reviewDescription = "JUNIT Update Review Restaurant Description Updated";

    Log.d( tag: "Update Restaurant Review Updated Id", review.reviewId);
    Log.d( tag: "Update Restaurant Review Updated Email", review.userEmail);
    Log.d( tag: "Update Restaurant Review Updated Rating", String.valueOf(review.rating));
    Log.d( tag: "Update Restaurant Review Updated Restaurant Name", review.restaurantName);
    Log.d( tag: "Update Restaurant Review Updated Restaurant Desc", review.reviewDescription);

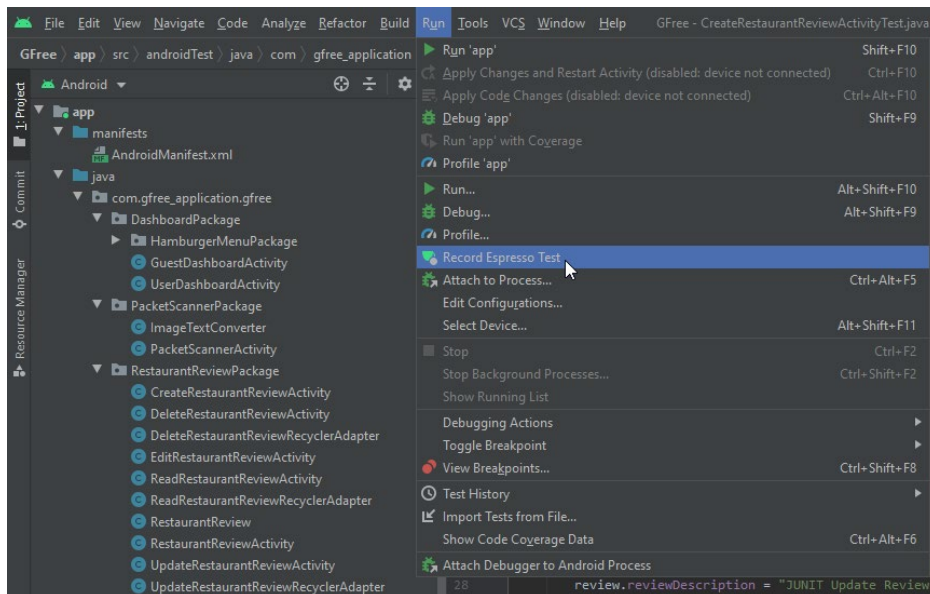
    //Assert/Check the updated created instance.
    assertEquals( expected: "DEF", review.reviewId); //Assert review id
    assertEquals( expected: "junittestemailupdated@gmail.com", review.userEmail); //Assert review email
    assertEquals( expected: 2.5F, review.rating); //Assert review rating
    assertEquals( expected: "JUNIT Update Review Restaurant Name Updated", review.restaurantName); //Assert review name
    assertEquals( expected: "JUNIT Update Review Restaurant Description Updated", review.reviewDescription); //Assert review description

}
}
```

As well as unit tests, I have also included some frontend user interface testing in the form of Espresso which is a testing suite included by default with Android Studio. Espresso allows me to mimic an end user running through my application while also being able to see the test execute real time using my user interface.

Espresso tests are written by using the Record Your Test window this allows me to write Espresso tests super quickly and efficiently by manually running through the application once, then the test can play this back as many times as I need.

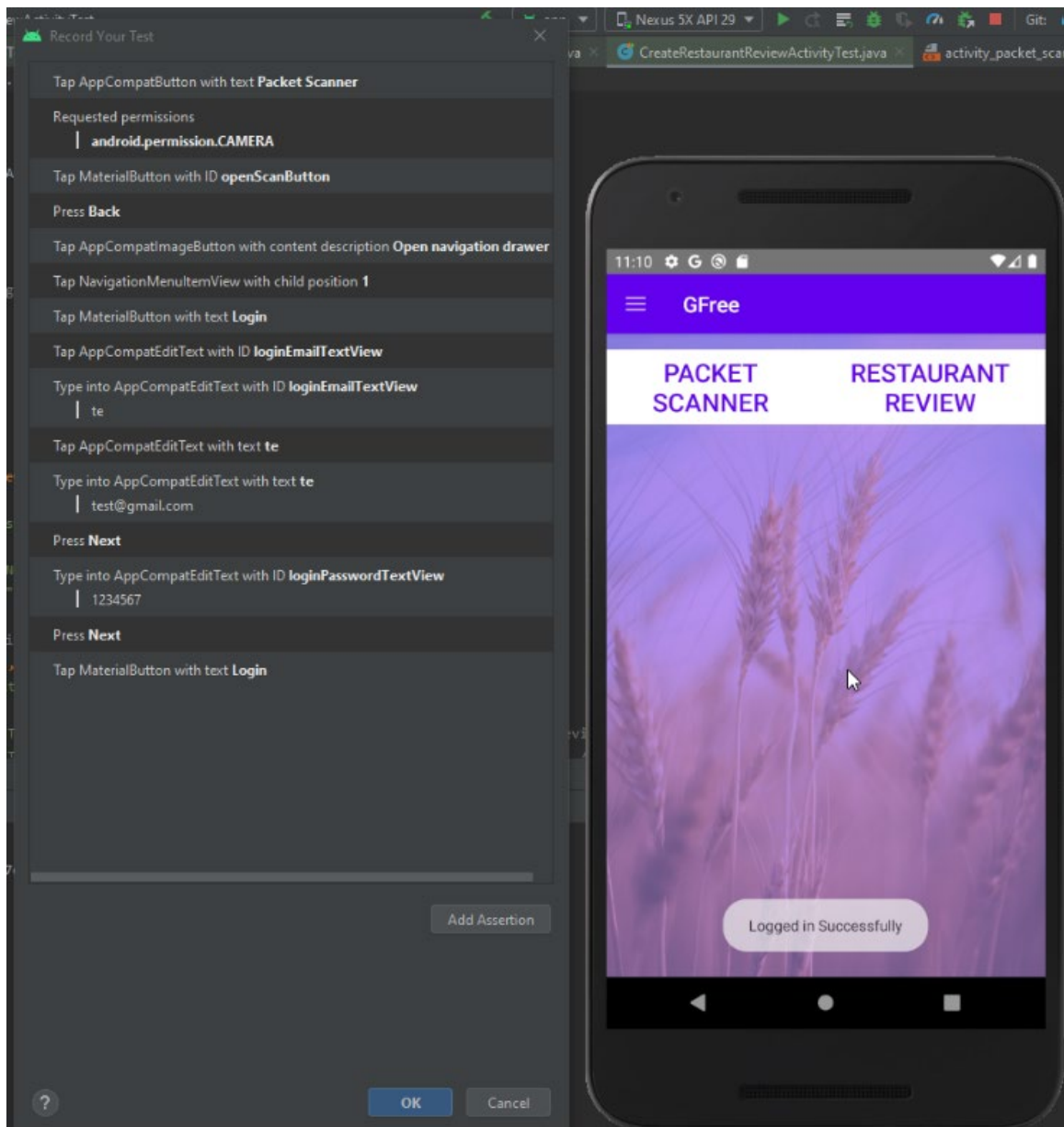
The first step to an Espresso test is to record one.



Once the record Espresso test option is selected, the Record Your Test window will appear along with the application being launched on an emulator or plugged-in debug device.

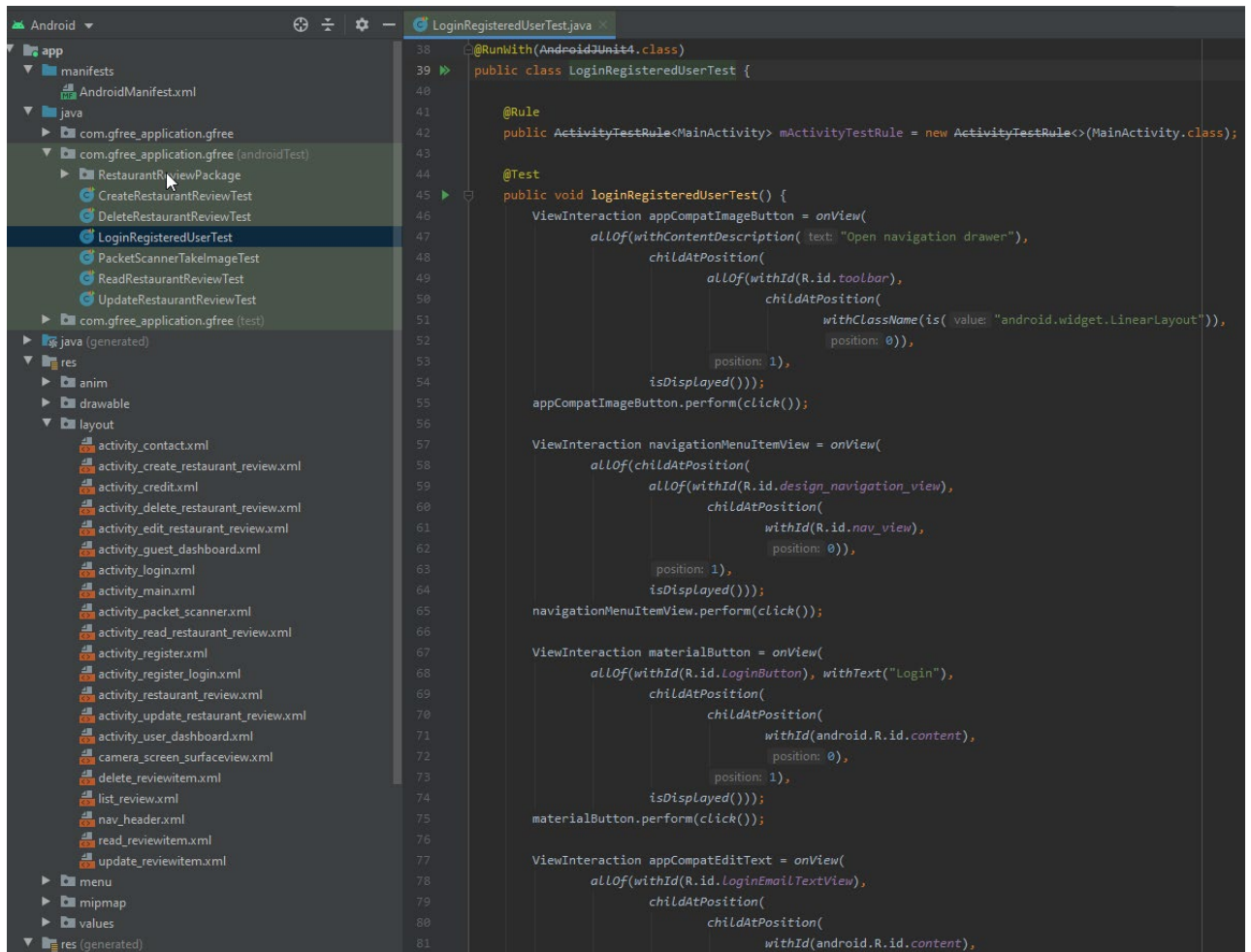
From here I would navigate through the application to activities I want included in my tests. As I navigate the application the Record Your Test window will be updated with every interaction I have had with the application. These interactions or steps will then be saved and will be used in the automated user interface test.

The example image below is me manually navigating through the login process to make an automated Espresso test to test logging in. This process then gives me the Espresso code should I need to make programmatic changes to the tests after they have been recorded.



This is the outputted Espresso code of the LoginRegisteredUserTest.class which was generated from the recorded Espresso login test from above. The Record Your Test wizard removes the need for manual coding for espresso tests and allows for this code to be easily generated in a more user-friendly manner. This is done by simply running through the test once manually while recording.

Each piece of code here is an instruction of what to do next and what should happen. This essentially is the steps to the test but visualised in code.



```
38 @RunWith(AndroidJUnit4.class)
39 public class LoginRegisteredUserTest {
40
41     @Rule
42     public ActivityTestRule<MainActivity> mActivityTestRule = new ActivityTestRule<>(MainActivity.class);
43
44     @Test
45     public void loginRegisteredUserTest() {
46         ViewInteraction appCompatImageButton = onView(
47             allOf(withContentDescription(text("Open navigation drawer")),
48                 childAtPosition(
49                     allOf(withId(R.id.toolbar),
50                         childAtPosition(
51                             withClassName(is(value("android.widget.LinearLayout")),
52                                 position: 0)),
53                             position: 1),
54                 isDisplayed()));
55         appCompatImageButton.perform(click());
56
57         ViewInteraction navigationMenuItemView = onView(
58             allOf(childAtPosition(
59                 allOf(withId(R.id.design_navigation_view),
60                     childAtPosition(
61                         withId(R.id.nav_view),
62                         position: 0)),
63                 position: 1),
64                 isDisplayed()));
65         navigationMenuItemView.perform(click());
66
67         ViewInteraction materialButton = onView(
68             allOf(withId(R.id.LoginButton), withText("Login"),
69                 childAtPosition(
70                     childAtPosition(
71                         withId(android.R.id.content),
72                         position: 0),
73                     position: 1),
74                 isDisplayed()));
75         materialButton.perform(click());
76
77         ViewInteraction appCompatEditText = onView(
78             allOf(withId(R.id.LoginEmailTextView),
79                 childAtPosition(
80                     childAtPosition(
81                         withId(android.R.id.content),
```

For a different academic module, Usability Design I used the GFree project as my topic for the final assignment. During this assignment I did a bunch of End User testing for GFree and so I will include the work created in that module here in my final document as the results are still relevant and relate to GFree.

Testing of GFree was carried out by 5 participants from one of our main user groups, individuals affected by celiacs disease. These are individuals who are affected themselves or know a friend or family member who is diagnosed with the disease.

An informed consent form was signed by each participant following research integrity guidelines and best practices [see appendix 7.4](#).

Every participant was presented the following introduction at the beginning of the testing process.

“Hello there, thank you for taking your time to participate in the testing of our gluten free android application. Our Android application is still currently in development and we would like some feedback from our potential users so we can improve upon our design moving forward.

The purpose of these tests is to improve our application and by no means is it a test on you or your abilities.

The scenario we would like you to imagine yourself in is that *you are a person diagnosed with celiacs disease who is trying to find out what food products are safe to consume in a supermarket/restaurant setting*. We will use a number of testing techniques that will help us understand if our application is in a good user-friendly state.”

Trunk Test

Trunk testing was performed on GFree in which we presented our testers with the dashboard of our application. We asked users:

- What is this application?
- What page do you think you are on, where are you in the application?
- Can you see any major sections or functionality?
- Where can you scan packets of information?
- Where can you make reviews?
- Where do you logout?

Recording of trunk test: <https://youtu.be/hZt-0-6FkNk>

Think Aloud

For think aloud testing we gave our participants tasks and asked them to speak aloud their thinking processes when trying to complete them. The tasks are as follows:

- Register and then login to the application.
- Scan a food product ingredient label using the packet scanner.
- Create a restaurant review.

This process of testing allows us to get an understanding of how a user approaches and comprehends an interface.

Recording of think aloud: https://youtu.be/A9m_tfi9wYw

5 Second Test

The 5 second test was conducted for all 5 participants and from this we evaluated how much of each screenshot shown was remembered by the participant after 5 seconds. Before the screenshot was shown each participant was informed that they will only see the image for 5 seconds and that they would have to remember what was on the screen after the 5 seconds was concluded.

This type of testing allows us to understand if the content is clear and concise or convoluted and complex.

Recording of 5 second test: <https://youtu.be/fpRuGptcCt8>

Click Test

The click test was performed with all 5 participants using images of the application so they can focus on what is presented. I observed the first piece of functionality or button the user interacted with and noted the results.

Heuristic Evaluation

For the heuristic evaluation I evaluated the applications heuristics following the 10 usability heuristics outlined by the Nielsen Norman Group while using participants feedback and the data they allowed us to gather.

A complete version of all test results is available on [Appendix User Testing Results](#).

System Usability Scale Questionnaire

For the debriefing after all testing was concluded we asked all participants to take part in the exit survey which was our SUS questionnaire. The SUS questions were simple but useful in getting critical feedback about different aspects of the application especially since participants could answer anonymously.

The overall score of our SUS questionnaire on our application after calculation is pretty good with our application scoring no lower than 50 and averaging a score of 79 after calculating all participant's feedback. A sample of the exit survey applied can be found in the appendix [Exit Survey](#).

Figure 3 SUS results

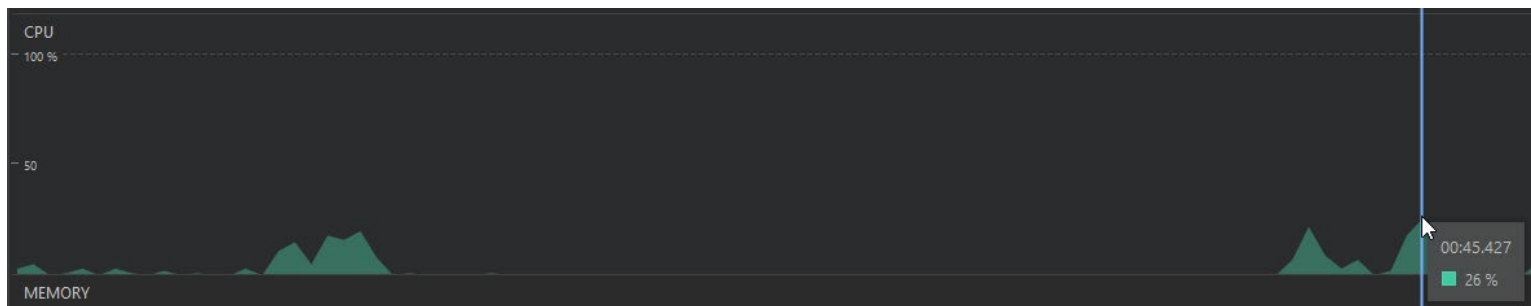
Question	1	2	3	4	5	6	7	8	9	10	SUS Score	Final SUS Score (Out of 100)
Participant 1	4	1	4	2	5	3	4	1	4	1	33	82.5
Participant 2	4	1	3	2	5	1	4	1	4	2	33	82.5
Participant 3	3	3	2	4	4	2	3	4	3	2	20	50
Participant 4	5	1	4	1	5	1	4	2	4	2	35	87.5
Participant 5	5	2	5	2	4	1	5	1	5	1	37	92.5
SUS Score Average:											79	

2.0 Evaluation

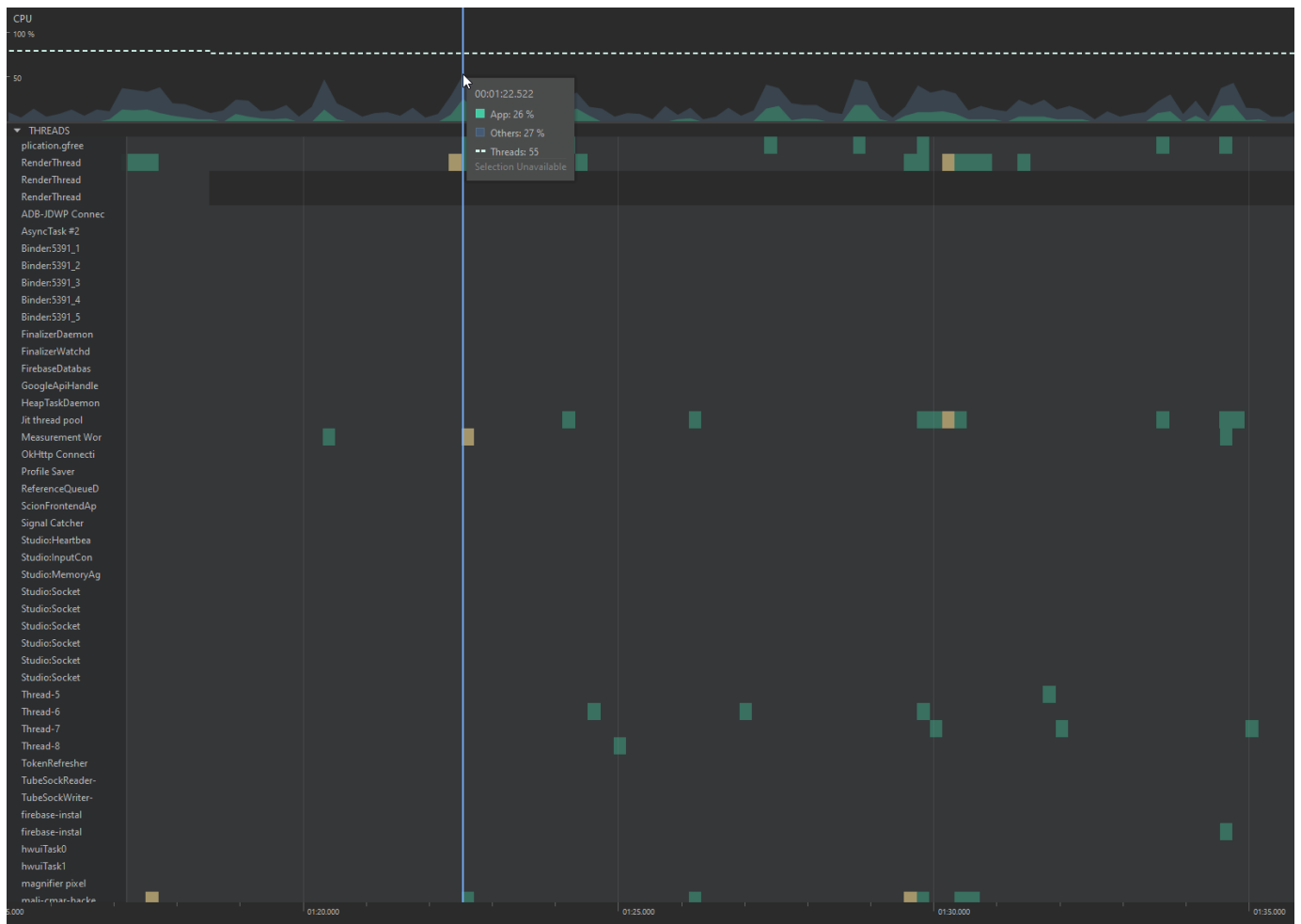
CPU

With ease of navigation as a requirement and a quick responsive application being desired GFree was tested using the Android profiler. The Android profiler is a tool that is included with Android Studio by default and allows me as the developer to monitor resource usage with metrics on the central processing unit (CPU) usage, random access memory (RAM) usage, network or mobile data usage, and the amount of energy used per action showing peaks and troughs for all metrics.

While monitoring GFree the CPU usage never went higher than 26% meaning my application really does not take a lot of resources to use. This is a massive benefit and will allow for quick load times on a wide variety of devices covering all price points, both expensive and cheaper phones.



An image of the CPU usage metric shows how much of the CPU is used during normal usage of the application. During this testing I scanned packets of food and created restaurant reviews which is behaviour expected of end users.



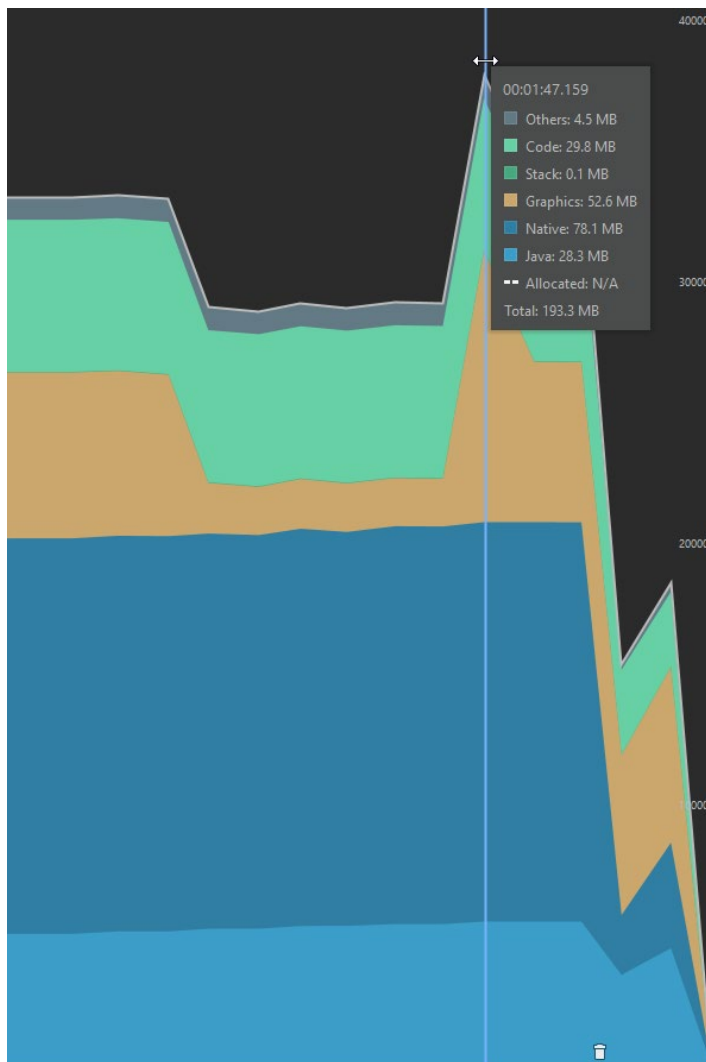
A more detailed look at the CPU metrics again shows the 26% app usage but also viewable now is the individual threads and the processes being served to those threads from left to right.

With the individual processes now being visible represented by green or yellow squares on the graph it is clear that there is no chance of a dead lock scenario occurring. This means the CPU will not be backed up and inundated with processes and is a good sign of an efficient and quick application.

RAM

Now looking at the RAM usage for GFree, RAM usage never peaked over 193MB which again is an excellent amount considering most Android devices come standard with 2GB (2000MB) of RAM available to them.

In Ireland specifically 19.4% of mobile devices have 2GB of RAM this grows to 63.34% when you consider 2GB+ of RAM. Even the individuals with older smartphones who total 14.02% have 1GB of RAM and should be fine. (Device Atlas, 2019)



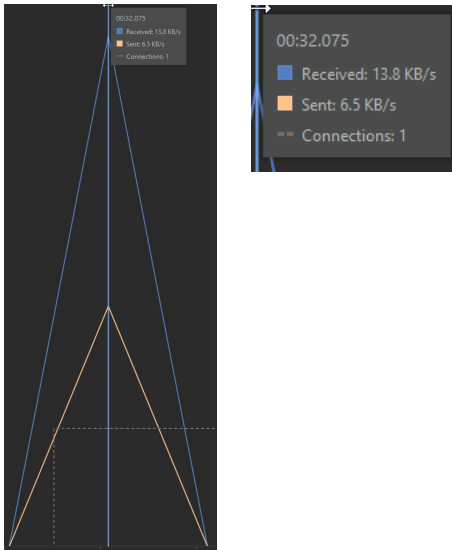
The above graph shows the breakdown of the 193MB total RAM usage and where the resources are being allocated and why. For instance, 28.3MB goes to Java and the Java Run Time Environment (RTE). This is responsible for converting .java files that I have coded into .class files that are readable by the Java Run Time Environment.

78.1 MB goes to the native Android operating system, a further 29.8 MB goes to intermediate code this could be things like Firebase and variables in memory. Followed by the remainder being made up of the visual graphics at 52.6MB and Others at 4.5MB.

This metric shows again that the application does not consume an unusual amount of RAM and it shows me as the developer that all activities are running efficiently.

Network

Network usage is minimal throughout the GFree application. GFree only uses wifi or mobile data when logging in to use the restaurant reviews functionality as the packet scanner does not require an internet connection.



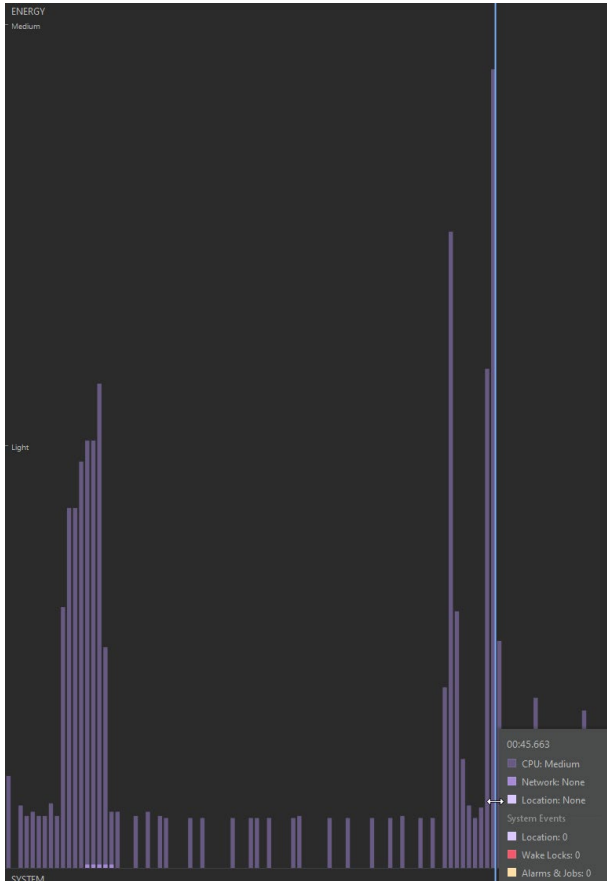
The maximum network data usage is 20 KB/s which is further broken down into 6.5KB/s for the request to the Firebase database, and 13.8 KB/s for the response from the Firebase database. The request visible in this image was for the logging into an existing account.

All network activity in GFree is under 20KB/s total and all communication when viewed appears as a sharp spike like this image meaning no network streams are kept open wasting users' valuable mobile data. This metric is important especially considering the mobile uses of the GFree application in supermarket settings and while out looking for somewhere to eat.

Energy Usage

The final metric given by Android profiler is the energy usage. Energy usage for GFree is important as the application could potentially be used for a prolonged portion of time during shopping.

An energy inefficient application would cause users phone battery to run out quickly during shopping which is not an ideal scenario.



By looking at the graph for power usage GFree does not use a lot of energy. The highest energy usage is in the medium bracket of the graph while the average hovers around light and system. Medium and light power usage are self-explanatory while system is the natural power usage of the mobile device regardless of the GFree application.

GFree only spikes to medium power usage for a second or two when initially loading an activity and it is safe to say that GFree on average stays within the low bracket for power usage.

3.0 Conclusions

GFree as an idea is an excellent one. Not too many applications currently offered on the Google App Store are designed with celiacs in mind. The small number of applications that are available are paid and therefore have a barrier of entry.

As of April 2021 there are 54 applications available that are related to celiacs disease, of these 54 applications only 4 of them are similar in functionality to GFree. (Google, 2021)

All these applications when tested typically offer a barcode scanner which requires a database of food that is compiled by the application owner. One of the disadvantages to this approach is that the database must be updated and most do not have Irish food products registered into the database making the barcode scanner useless.

With GFree the use of smart text reading allows me as the developer to bypass the barcode system and removes the need for a database which would be cumbersome to me during development.

Instead, I can programmatically examine individual ingredients, I can apply localisation so the GFree packet scanner recognises other languages and works abroad. The possibilities for my GFree packet scanner are impressive and this gives me a massive advantage when compared to the barcode approach. From doing my research and using other gluten free applications the packet scanner offered by GFree will be a first making GFree unique in this sense.

When looking at GFree and the restaurant review system that is offered there are features that if implemented would improve the user experience with this section of the application. Currently users can create, read, update, and delete reviews but all reviews do not have a restaurant location for example, so it is hard to know exactly what restaurant is being referenced in a review with only the restaurant name currently being the identifying factor.

A way I could improve upon this disadvantage would be to include an interactive Google map that plots the lat/long coordinates of each review which would allow for easy finding of restaurants when reviews are being read by other users.

Overall, I consider GFree a successful project that has helped me progress my skills in multiple areas like programming, communication, and management. Any stark disadvantages of GFree can be fixed easily with further development so long as time permits. The project and in turn final product are in my opinion very marketable once most of the disadvantages are removed.

4.0 Further Development or Research

With additional resources and generally not having an academic time restraint for my project I think my application has the potential to go to market. With more time I think I could make a more refined product by adding more quality-of-life features like an interactive packet scanner that gives detailed information on every type of ingredient detected from a scanned packet.

There are plenty of applications already available on the Google Play store for celiacs but none have the functionality I offer with GFree and many of them are American based so this will be a project I work on and refine moving forward.

5.0 References

Brownlee, J. (2017) 'A Gentle Introduction to the Bag-of-Words Model', *Machine Learning Mastery*, 8 October. Available at: <https://machinelearningmastery.com/gentle-introduction-bag-words-model/> (Accessed: 5 May 2021).

Coeliac Society Of Ireland (2021) *Coeliac Society Of Ireland – Improving The Quality Of Life For People Living With Coeliac Disease*, *Coeliac Society Of Ireland – Improving The Quality Of Life For People Living With Coeliac Disease*. Available at: <https://coeliac.ie/> (Accessed: 4 May 2021).

Device Atlas (2019) *Most common smartphone RAM by country*, *DeviceAtlas*. Available at: <https://deviceatlas.com/blog/most-common-smartphone-ram-by-country> (Accessed: 15 May 2021).

Food Safety Authority of Ireland (2017) *Gluten-free Declarations | FAQs | The Food Safety Authority of Ireland*, *Gluten-free Declarations*. Available at: https://www.fsai.ie/faq/gluten_free_declarations.html (Accessed: 5 May 2021).

Gluten-Free Living (2016) *Ingredients Index*, *Gluten-Free Living*. Available at: <https://www.glutenfreeliving.com/gluten-free-foods/ingredients/ingredients-index/> (Accessed: 5 May 2021).

Google (2021) *Google Play Store*, *Google Play Store*. Available at: https://play.google.com/store?hl=en_GB&gl=IE (Accessed: 12 May 2021).

Health Service Executive (2011) *HSE Coeliac Disease*, *HSE.ie*. Available at: <https://www.hse.ie/eng/health/az/c/coeliac-disease/> (Accessed: 4 May 2021).

The Institute for Functional Medicine (2021) *Rates of Celiac Disease Continue to Rise Worldwide | IFM*, *The Institute for Functional Medicine*. Available at: <https://www.ifm.org/news-insights/gut-rates-celiac-disease-continue-rise-worldwide/> (Accessed: 4 May 2021).



National College of Ireland

Project Proposal Gluten Free Project 04/11/2020

Computing

Software Development

Academic Year 2020/2021

Max Delaney

18100368

x18100368@student.ncirl.ie

Contents

1.0	Objectives	1
2.0	Background	1
3.0	Technical Approach	1
4.0	Special Resources Required	1
5.0	Project Plan	1
6.0	Technical Details	1
7.0	Evaluation	1
8.0	Invention Disclosure Form (Remove if not filled)	1

Objectives

The objective of my project is to give people with Celiacs disease an application that they can use when wanting to shop, cook, or bake celiac food or when wanting to eat out.

My application will contain a review system for restaurants that will be compiled of reviews left from users who are celiac.

With these reviews my user base will be able to tell immediately if a restaurant caters for celiacs disease.

Another purpose would be to see the extent a given restaurant caters to the disease, many restaurants boast they cater for celiacs disease, but some don't offer the full menu. With my application I want to make my users informed.

A barcode/package scanner will be another feature present in my application that will be used to immediately identify what shopping products are suitable for celiacs and which ones are not.

Background

Celiacs disease is a digestive disorder that prevents individuals from consuming wheat. Any celiacs that consume wheat, once digested will feel immediately sick.

Celiacs disease is known to be life threatening if people diagnosed don't stick to their wheat free diet.

The idea of this project is quite a personal one. My own mother and grandmother have celiacs disease and ever since they were diagnosed with it 10 years ago the shopping list and where they can eat out has been impacted massively.

My mother has visited many restaurants over the years and quite a large margin of them don't cater for celiacs.

Many restaurants that do, offer a tiny menu of "safe foods" which are dishes that are considered to be safe by restaurants but often they will be quite bland dishes like salads for example. I hope that my application could change how celiacs view eating out.

Technical Approach

For the beginning of my research, I downloaded celiac focused apps that are available on the Google play store. I did this to get an idea of the apps that are available and to see what features they offer. There are plenty of apps available that cater to celiacs and some cover similar areas of my idea, but they cost money to access and each is an individual application.

To see if my project was viable and has a potential user base, I had to look no further than my own family as stated before I have two family members with celiacs disease who acted as an early sign that my project has good potential.

On the technological side of research, I must research how Google Firebase works. I will be using their official documentation with regards to how the database is set up. Afterwards I will research how the server is interacted with so I will get an understanding of how data is sent and received.

I already have previous knowledge of my IDE Android Studio and I'm currently studying multimedia and mobile application development as a module this year so there is little to no research required for this aspect of my project.

As for how my natural language processor (NLP) will work, I will need to look further into how it can be implemented into an Android application. I have very basic experience with NLP's from a previous project in the second or third year of my studies but in that context the NLP was a 3rd party API and was for a web-based application so this is something completely new.

Special Resources Required

The requirements of my project will be a database to store recipes and reviews that are left by users.

Another database might be required for the natural language processing aspect of my project depending on how the data is managed, handled and used.

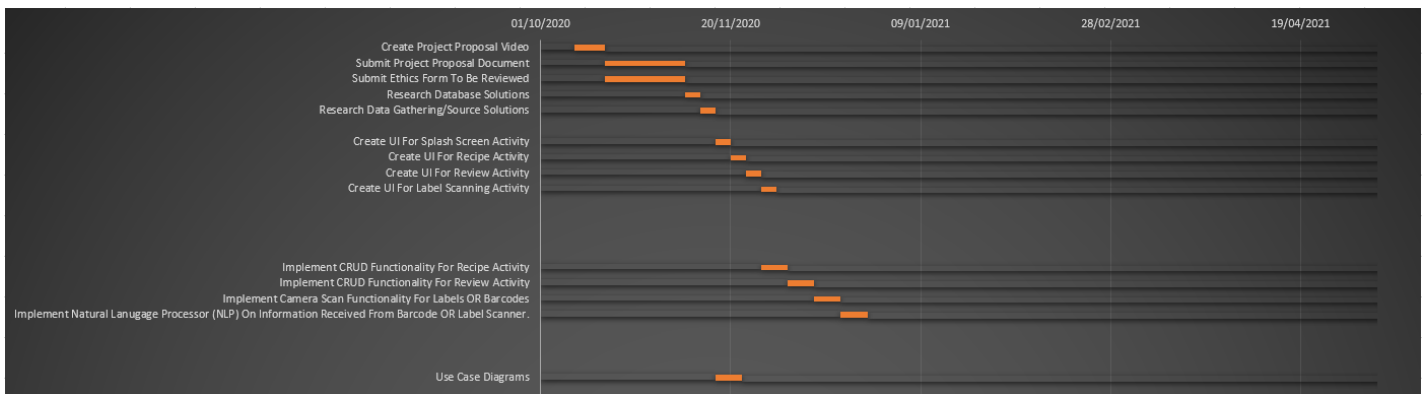
Project Plan

For my project plan I aim to lay out every task that will be undertaken be it research oriented or programming. In my Gantt chart I have set aside some time for research on each part of my project.

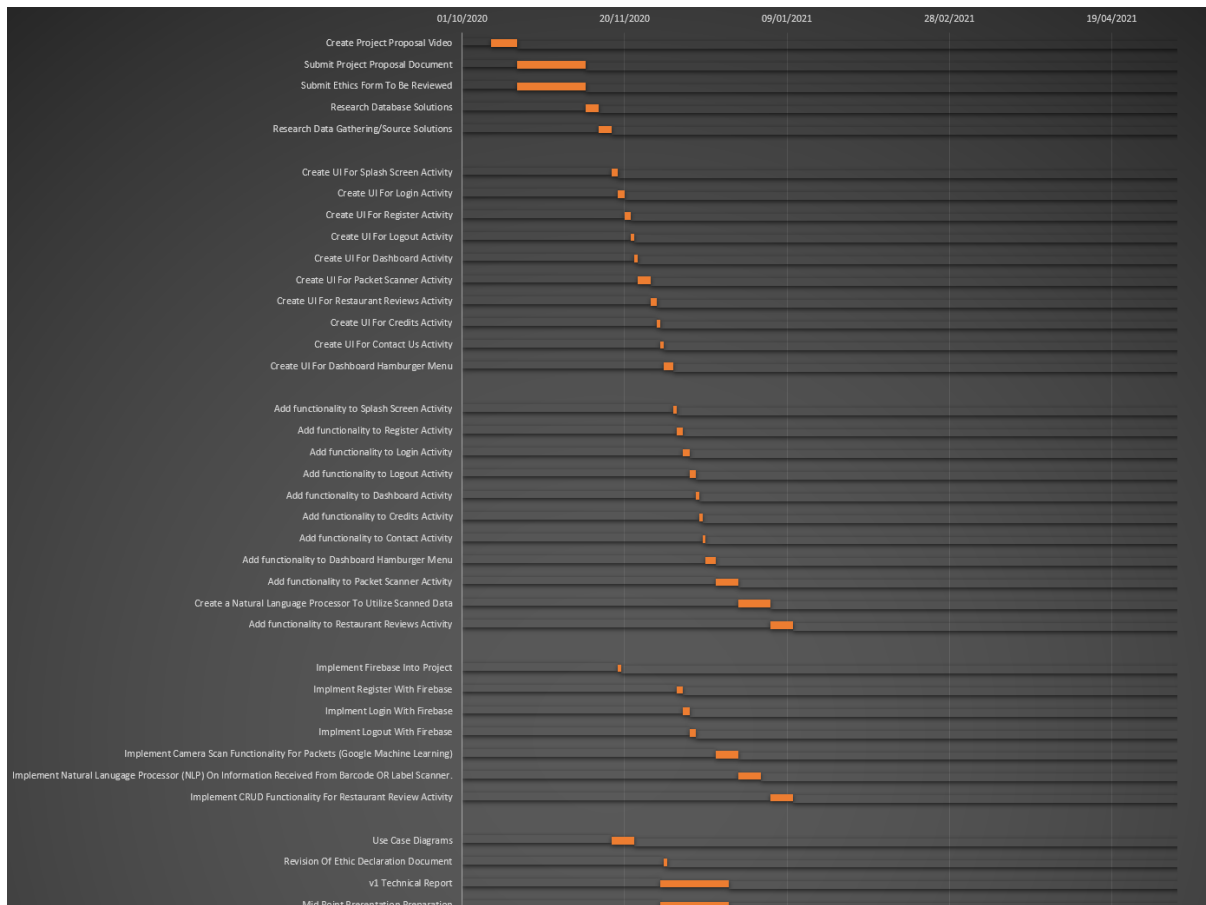
Ignoring research related tasks. My Gantt chart currently has tasks for creating the user interfaces before I begin to fully implement the backend database functionality, which would be implementing my CRUD functionality.

I still have to consider the task and the allotted time needed for implementing the solution to my natural language processor, but I know for sure that will be a later task.

v1



v2



Technical Details

For my applications database solution I will be using a firebase database that will be used to store information from user generated reviews and recipes.

This also could potentially be where my natural language processing will take place, however another service might be required.

With my application being Android based I aim to use Java over Kotlin in Android Studio as I already have some good knowledge of Java and because of this I think this would be the best decision.

I'm also considering making my own API so I can send and receive information using JSON which would allow my application to be more agile and flexible in how the data will be handled. Doing this allows the option of reusing data somewhere else in my application if it was required for any given reason.

Evaluation

For the front-end UI testing of my Android application, I will create some small unit tests. This is something I have experience within web development using Cypress however using JUnit for Android is something that is new to me and will require some reading of the official documentation.

JUnit tests will allow me to rapidly test features like input fields and their validation. JUnit also allows for a modular approach to testing in which I could create individual tests for each activity in my application.

Once my application is in a human testable state, I will ask my family members who have celiacs disease to use the application and evaluate their feedback on the layout, style and features offered.

Once I have received feedback, I can use this to implement improvements or streamline processes.

DECLARATION OF ETHICS CONSIDERATION**School of Computing**

Student Name: Max Delaney

Student ID: 18100368

Programme: BSHC Software Development **Year:** 4

Module: Software Project

Project Title: GFree

Please circle (or highlight) as appropriate

This project involves human participants.	<input checked="" type="radio"/> Yes <input type="radio"/> No
---	---

Introduction

Secondary data refers to data that is collected by someone other than the current researcher. Common sources of secondary data for social science include censuses, information collected by government departments, organizational records and data originally collected for other research purposes. Primary data, by contrast, is collected by the investigator conducting the research.

A project that does not involve human participants requires ONLY completion of Declaration of Ethics Consideration Form and submission of the form on module's Moodle page

A project that involves human participants requires ethical clearance and an Ethics Application Form must be submitted through the module's Moodle page. Please refer to and ensure compliance with the ethical principles stated in NCI Ethics Form available on the Moodle page.

The following decision table will assist you in deciding if you have to complete the Declaration of Ethics Consideration Form or/and the Ethics Application Form.

Public Data	Y	Y	Y	Y	N	N	N	N
Private Data	Y	Y	N	N	Y	Y	N	N
Human Participants	Y	N	Y	N	Y	N	Y	N
Declaration of Ethics Consideration Form	x	X	x	X	X	X	x	
Ethics Application Form	X		X		X		X	

Please circle (or highlight) as appropriate

The project makes use of secondary dataset(s) created by the researcher	Yes	No
The project makes use of public secondary dataset(s)	Yes	No
The project makes use of non-public secondary dataset(s)	Yes /	No
Approval letter from non-public secondary dataset(s) owner received	Yes /	No

Sources of Data:

It is students' responsibility to ensure that they have the correct permissions/authorizations to use any data in a study. Projects that make use of data that does not have authorization to be used, will not be graded for that portion of the study that makes use of such data.

Public Data

A project that makes use of public secondary dataset(s) **does not need ethics permission**, but **needs a letter/email from the copyright holder** regarding potential use.

Some websites and data sources allow their data sets to be used under certain conditions. In these cases, a letter/email from the copyright holder is NOT necessary, but the researcher should cite the source of this permission and indicate under what conditions the data are allowed to be used. See Appendix I for examples of permissions granted by Fingal Open Data, and Eurostat website.

Where websites or data sources indicate that they do not grant permission for data to be used, you will still need a letter/email from the copyright holder. For example, see Appendix II for an example from the Journal of Statistics Education.

Private Data

A project that makes use of non-public (private) secondary dataset(s) must receive data usage permission from School of Computing.

An approval letter/email from the owner (e.g. institution, company, etc.) **of the non-public secondary dataset must be attached to the Declaration of Ethics Consideration**. The letter/email must confirm that the dataset is anonymised and permission for data processing, analysis and public dissemination is granted.

Evidence for use of secondary dataset(s)

Include dataset(s) owner letter/email or cite the source for usage permission.

CHECKLIST

Non-public/private secondary dataset(s) -Owner letter/email is attached to this form	Yes / No
OR Citation and link to the web site where permission is granted – provided in this form	Yes / No

ETHICS CLEARANCE GUIDELINES WHEN HUMAN PARTICIPANTS ARE INVOLVED

The Ethics Application Form must be submitted on Moodle for approval prior to conducting the work.

Considerations in data collection

- Participants will not be identified, directly or through identifiers linked to the subjects in any reports produced by the study
- Responses will not place the participants at risk of professional liability or be damaging to the participants' financial standing, employability or reputation
- No confidential data will be used for personal advantage or that of a third party

Informed consent

- Consent to participate in the study has been given freely by the participants
- participants have the capacity to understand the project goals.
- Participants have been given information sheets that are understandable

- Likely benefits of the project itself have been explained to potential participants
- Risks and benefits of the project have been explained to potential participants
- Participants have been assured they will not suffer physical stress or discomfort or psychological or mental stress
- The participant has been assured s/he may withdraw at any time from the study without loss of benefit or penalty
- Special care has been taken where participants are unable to consent for themselves (e.g children under the age of 18, elders with age 85+, people with intellectual or learning disability, individuals or groups receiving help through the voluntary sector, those in a subordinate position to the researcher, groups who do not understand the consent and research process)
- Participants have been informed of potential conflict of interest issues
- The onus is on the researcher to inform participants if deception methods have to be used in a line of research

I have read, understood, and will adhere to the ethical principles described above in the conduct of the project work.

Signature: Max Delaney

Date: 06/05/2021

Ethics Appendix I

1) Fingal Open Data: <http://data.fingal.ie/About>

Licence

Citizens are free to access and use this data as they wish, free of charge, in accordance with the Creative Commons Attribution 4.0 International License (CC-BY).

Note: From November 2010 to July 2015, data on Fingal Open Data was published in accordance with the PSI general licence.

Use of any published data is subject to Data Protection legislation.

Licence Statement

Under the CC-BY Licence, users must acknowledge the source of the Information in their product or application by including or linking to this attribution statement: "Contains Fingal County Council Data licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) licence".

Multiple Attributions

If using data from several Information Providers and listing multiple attributions is not practical in a product or application, users may include a URI or hyperlink to a resource that contains the required attribution statements.

2) Eurostat: <https://ec.europa.eu/eurostat/about/policies/copyright>

COPYRIGHT NOTICE AND FREE RE-USE OF DATA

Eurostat has a policy of encouraging free re-use of its data, both for non-commercial and commercial purposes. All statistical data, metadata, content of web pages or other dissemination tools, official publications and other documents published on its website, with the exceptions listed below, can be reused without any payment or written licence provided that:

- the source is indicated as Eurostat
- when re-use involves modifications to the data or text, this must be stated clearly to the end user of the information

Ethics Appendix II

Journal of Statistics Education: http://jse.amstat.org/jse_users.htm

JSE Copyright and Usage Policy

Unlike other American Statistical Association journals, the Journal of Statistics Education (JSE) does not require authors to transfer copyright for the published material to JSE. Authors maintain copyright of published material. Because copyright is not transferred from the author, permission to use materials

published by JSE remains with the author. Therefore, to use published material from a JSE article the requesting person must get approval from the author.

User Testing Results

These are the results gathered from the Usability Design assignment that covered GFree as a topic.

Trunk Test

Participant Num	Task Name	Task Goal	Start Time	End Time	Expected/Ideal Behaviour	Actual Behaviour	Notes/ Comments
Participant 1							
Participant 1	What is this application?	The user must be able to identify what the application is and its name.	0:18	0:26	The participant identifies the application name and understands its purpose.	The participant identified the application name.	
Participant 1	What page do you think you are on, where are you in the application?	The participant must be able to clearly identify what page they are on either by name or by function.	0:30	0:36	The participant clearly outlines where they are by referencing the activity name (page name).	The participant correctly outlined they are at the homepage (dashboard) of the application.	
Participant 1	Can you see any major sections or functionality?	The participant must outline the key functionality offered by the application.	0:44	0:50	The participant clearly identifies key functionality offered like the packet scanner and restaurant reviews.	The participant identified the packet scanner and restaurant review functionality.	
Participant 1	Where can you scan packets of food?	The participant must be able to locate the packet scanner.	0:57	1:06	The participant locates the packet scanner on the application dashboard and scans a product.	The participant found the packet scanner.	
Participant 1	Where can you make reviews?	The participant must be able to locate the restaurant reviews section.	1:11	1:20	The participant locates the restaurant review section on the application dashboard and creates a restaurant review.	The participant found the restaurant review section.	
Participant 1	Where do you logout?	The participant must locate the signout functionality	1:26	1:32	The participant locates the signout option located in the hamburger menu on the dashboard.	The participant found the signout option inside the hamburger menu.	

Participant 2

Participant 2	What is this application?	The user must be able to identify what the application is and its name.	0:20	0:39	The participant identifies the application name and understands its purpose.	The participant read aloud the application name from the header/title.	
Participant 2	What page do you think you are on, where are you in the application?	The participant must be able to clearly identify what page they are on either by name or by function.	0:45	0:53	The participant clearly outlines where they are by referencing the activity name (page name).	The participant assumes they are at the homepage (dashboard) of the application.	Judging by the way this question was answered the participant guessed where they were. Clearer information for what activity is present would help.
Participant 2	Can you see any major sections or functionality?	The participant must outline the key functionality offered by the application.	0:56	1:05	The participant clearly identifies key functionality offered like the packet scanner and restaurant reviews.	The participant identified the main functionality to be the packet scanner and restaurant reviews section.	
Participant 2	Where can you scan packets of food?	The participant must be able to locate the packet scanner.	1:10	1:18	The participant locates the packet scanner on the application dashboard and scans a product.	The participant found the packet scanner on the dashboard quickly.	
Participant 2	Where can you make reviews?	The participant must be able to locate the restaurant	1:25	1:32	The participant locates the restaurant review section on the application dashboard and creates a restaurant review.	The participant found the restaurant section and its sub creation activity quickly.	
Participant 2	Where do you logout?	The participant must locate the signout functionality	1:38	1:49	The participant locates the signout option located in the hamburger menu on the dashboard.	The participant had taken a few seconds to find the signout functionality.	

Participant 3

Participant 3	What is this application?	The user must be able to identify what the application is and its name.	0:21	0:34	The participant identifies the application name and understands its purpose.	The participant identified the application as GFree.	
Participant 3	What page do you think you are on, where are you in the application?	The participant must be able to clearly identify what page they are on either by name or by function.	0:39	0:51	The participant clearly outlines where they are by referencing the activity name (page name).	The participant successfully identifies the dashboard as the applications homepage.	
Participant 3	Can you see any major sections or functionality?	The participant must outline the key functionality offered by the application.	1:02	1:11	The participant clearly identifies key functionality offered like the packet scanner and restaurant reviews.	The participant outlines that the packet scanner restaurant review and hamburger menu are menus that contain functionality.	
Participant 3	Where can you scan packets of food?	The participant must be able to locate the packet scanner.	1:18	1:29	The participant locates the packet scanner on the application dashboard and scans a product.	The participant found the packet scanner.	
Participant 3	Where can you make reviews?	The participant must be able to locate the restaurant	1:36	1:47	The participant locates the restaurant review section on the application dashboard and creates a restaurant review.	The participant found the restaurant review section.	
Participant 3	Where do you logout?	The participant must locate the signout functionality	1:53	2:04	The participant locates the signout option located in the hamburger menu on the dashboard.	The participant found the signout functionality after looking in the hamburger menu.	

Participant 4

Participant 4	What is this application?	The user must be able to identify what the application is and its name.	0:26	0:39	The participant identifies the application name and understands its purpose.	The participant had taken their time to answer but correctly remembered GFree.	
Participant 4	What page do you think you are on, where are you in the application?	The participant must be able to clearly identify what page they are on either by name or by function.	0:46	0:55	The participant clearly outlines where they are by referencing the activity name (page name).	The participant stated that they did not know where they were.	Clearer indication of current location is required.
Participant 4	Can you see any major sections or functionality?	The participant must outline the key functionality offered by the application.	1:01	1:16	The participant clearly identifies key functionality offered like the packet scanner and restaurant reviews.	The participant correctly states the key functionality to be the packet scanner and restaurant reviews but was hesitant.	
Participant 4	Where can you scan packets of food?	The participant must be able to locate the packet scanner.	1:32	1:58	The participant locates the packet scanner on the application dashboard and scans a product.	The participant correctly identified the packet scanner but was confused when the results screen showed.	Instead of showing the results screen first. I could change this to go straight to the live camera.
Participant 4	Where can you make reviews?	The participant must be able to locate the restaurant	2:05	2:48	The participant locates the restaurant review section on the application dashboard and creates a restaurant review.	The create restaurant review section was found and was filled in and submitted successfully.	
Participant 4	Where do you logout?	The participant must locate the signout functionality	2:55	3:30	The participant locates the signout option located in the hamburger menu on the dashboard.	The participant could not find the signout section and actively searched the packet scanner before finding it in the hamburger menu	

Participant 5

Participant 5	What is this application?	The user must be able to identify what the application is and its name.	0:20	0:32	The participant identifies the application name and understands its purpose.	The participant correctly named the application and additionally told me its purpose.	
Participant 5	What page do you think you are on, where are you in the application?	The participant must be able to clearly identify what page they are on either by name or by function.	0:38	1:00	The participant clearly outlines where they are by referencing the activity name (page name).	The participant did not clearly outline the activity name and instead referenced the functionality offered.	Perhaps the creation of a header to outline where the user currently is could be a solution to this confusion.
Participant 5	Can you see any major sections or functionality?	The participant must outline the key functionality offered by the application.	1:07	1:24	The participant clearly identifies key functionality offered like the packet scanner and restaurant reviews.	The participant was easily able to identify the key functionality.	
Participant 5	Where can you scan packets of food?	The participant must be able to locate the packet scanner.	1:32	2:45	The participant locates the packet scanner on the application dashboard and scans a product.	The participant easily found the packet scanner and scanned an ingredient label	
Participant 5	Where can you make reviews?	The participant must be able to locate the restaurant	2:50	5:08	The participant locates the restaurant review section on the application dashboard and creates a restaurant review.	The participant from the dashboard easily found the restaurant review section of the application.	
Participant 5	Where do you logout?	The participant must locate the signout functionality	5:12	5:40	The participant locates the signout option located in the hamburger menu on the dashboard.	The participant instinctively clicked the hamburger menu for further functionality and found the sign out functionality.	

Think Aloud

Participant Num	Task Name	Task Goal	Start Time	End Time	Expected/Ideal Behaviour	Actual Behaviour	Notes/ Comments
Participant 1							
Participant 1	Register and then login to the application.	The participant registers a new account and then logs into their new account.	0:21	0:54	The participant successfully registers and logs into their new account.	The participant successfully made an account and logged into the application.	
Participant 1	Scan a food product ingredient label using the packet scanner.	The participant clicks the packet scanner and scans a food label and takes the picture. The ingredients are listed.	1:05	1:46	The participant successfully scans a packet of food and gets the scanned results.	The participant successfully scanned a food product and viewed the scanned results.	
Participant 1	Create a restaurant review.	The participant clicks the restaurant reviews section then clicks create new review and fills in the form and submits it.	1:55	2:20	The participant creates a new review from the restaurant reviews section.	The participant successfully created a restaurant review.	

Participant 2							
Participant 2	Register and then login to the application.	The participant registers a new account and then logs into their new account.	0:26	1:05	The participant successfully registers and logs into their new account.	The participant successfully registered and logged into however they entered their password incorrectly.	Clearer notifications for incorrect password need to be implemented.
Participant 2	Scan a food product ingredient label using the packet scanner.	The participant clicks the packet scanner and scans a food label and takes the picture. The ingredients are listed.	1:15	2:02	The participant successfully scans a packet of food and gets the scanned results.	The participant scanned a food product successfully.	
Participant 2	Create a restaurant review.	The participant clicks the restaurant reviews section then clicks create new	2:15	3:05	The participant creates a new review from the restaurant reviews section.	The participant created a restaurant review successfully.	

		review and fills in the form and submits it.					
--	--	--	--	--	--	--	--

Participant 3							
Participant 3	Register and then login to the application.	The participant registers a new account and then logs into their new account.	0:25	0:50	The participant successfully registers and logs into their new account.	Participant registered and signed into the application.	
Participant 3	Scan a food product ingredient label using the packet scanner.	The participant clicks the packet scanner and scans a food label and takes the picture. The ingredients are listed.	1:03	1:40	The participant successfully scans a packet of food and gets the scanned results.	Participant scanned a food product label and got the returned results.	
Participant 3	Create a restaurant review.	The participant clicks the restaurant reviews section then clicks create new review and fills in the form and submits it.	1:54	2:30	The participant creates a new review from the restaurant reviews section.	The participant created a simple restaurant review successfully.	

Participant 4							
Participant 4	Register and then login to the application.	The participant registers a new account and then logs into their new account.	0:24	0:54	The participant successfully registers and logs into their new account.	The participant registered successfully and correctly logged into the application.	
Participant 4	Scan a food product ingredient label using the packet scanner.	The participant clicks the packet scanner and scans a food label and takes the picture. The ingredients are listed.	1:10	2:05	The participant successfully scans a packet of food and gets the scanned results.	The participant scanned a food ingredient label and read aloud the returned ingredients.	
Participant 4	Create a restaurant review.	The participant clicks the restaurant reviews section then clicks	2:14	2:53	The participant creates a new review from the restaurant	The participant created a restaurant review quickly.	

		create new review and fills in the form and submits it.			reviews section.		
--	--	---	--	--	------------------	--	--

Participant 5							
Participant 5	Register and then login to the application.	The participant registers a new account and then logs into their new account.	0:30	2:37	The participant successfully registers and logs into their new account.	The participant registered and logged in eventually but did have some problems with errors not displaying.	Better notifications for when things have gone wrong like email already used or password incorrect is required.
Participant 5	Scan a food product ingredient label using the packet scanner.	The participant clicks the packet scanner and scans a food label and takes the picture. The ingredients are listed.	2:40	3:40	The participant successfully scans a packet of food and gets the scanned results.	The participant scanned a food product ingredient label using the packet scanner and had taken their time to read some results.	
Participant 5	Create a restaurant review.	The participant clicks the restaurant reviews section then clicks create new review and fills in the form and submits it.	3:50	6:25	The participant creates a new review from the restaurant reviews section.	The participant created a restaurant review successfully but had taken longer than other participants due to one hand typing.	

5 Second Test

Participant Num	Task Name	Task Goal	Actual Behaviour	Notes/ Comments
Participant 1				
Participant 1	What is the applications name?	The participant recognises the application as GFree.	After 5 seconds the participant correctly named the application as GFree.	
Participant 1	What is the application about or who is it aimed at?	The participant recognises the application is about celiacs and gaining information about food products/restaurants.	The participant outlined that the application is for people who cannot have gluten.	This participant when asked forgot the term celiac but clearly knew the limits/boundary of the diet.
Participant 1	What does the application offer in terms of functionality?	The participant recognises a restaurant review section and packet scanner functionality.	The participant correctly stated both the packet scanner and restaurant reviews are functionality.	

Participant 2				
Participant 2	What is the applications name?	The participant recognises the application as GFree.	After 5 seconds the participant correctly named the application as GFree.	
Participant 2	What is the application about or who is it aimed at?	The participant recognises the application is about celiacs and gaining information about food products/restaurants.	The participant correctly stated the application is for celiacs who require a gluten free diet.	
Participant 2	What does the application offer in terms of functionality?	The participant recognises a restaurant review section and packet scanner functionality.	The participant identified and remembered key functionality from the dashboard and the hamburger menu.	

Participant 3

Participant 3	What is the applications name?	The participant recognises the application as GFree.	After 5 seconds the participant correctly named the application as GFree.	
Participant 3	What is the application about or who is it aimed at?	The participant recognises the application is about celiacs and gaining information about food products/restaurants.	The participant outlined the application is for celiacs.	
Participant 3	What does the application offer in terms of functionality?	The participant recognises a restaurant review section and packet scanner functionality.	The participant outlined and remembered the packet scanner but forgot to mention the restaurant reviews.	

Participant 4

Participant 4	What is the applications name?	The participant recognises the application as GFree.	After 5 seconds the participant correctly named the application as GFree.	
Participant 4	What is the application about or who is it aimed at?	The participant recognises the application is about celiacs and gaining information about food products/restaurants.	The participant outlined that the application is for celiacs.	
Participant 4	What does the application offer in terms of functionality?	The participant recognises a restaurant review section and packet scanner functionality.	The participant failed to recall the restaurant reviews as functionality but did remember the packet scanner.	

Participant 5

Participant 5	What is the applications name?	The participant recognises the application as GFree.	The participant correctly answered that the application is called GFree.	
Participant 5	What is the application about or who is it aimed at?	The participant recognises the application is about celiacs and gaining information about food products/restaurants.	The participant pre-emptively answered this question before I asked it. But correctly stated that the application is for celiacs.	
Participant 5	What does the application offer in terms of functionality?	The participant recognises a restaurant review section and packet scanner functionality.	The packet scanner and reviews were correctly outlined as functionality.	

Click Test

Participant Num	First Button Clicked	Comment	Notes
Participant 1	"Hamburger" Submenu	This participant was quite curious and clicked on the hamburger menu before any other button. They actively read aloud the other buttons like packet scanner and restaurant reviews but clicked the submenu first.	It is presumed before the test that the participant was logged into a testing account. This means all functionality is accessible.

Participant Num	First Button Clicked	Comment	Notes
Participant 2	Packet Scanner Button	The participant when presented with the application dashboard clicked the packet scanner functionality first.	It is presumed before the test that the participant was logged into a testing account. This means all functionality is accessible.

Participant Num	First Button Clicked	Comment	Notes
Participant 3	Packet Scanner Button	For this participant's first click in the application they selected the packet scanner from the dashboard.	It is presumed before the test that the participant was logged into a testing account. This means all functionality is accessible.

Participant Num	First Button Clicked	Comment	Notes
Participant 4	Restaurant Reviews	This participant was interested in the restaurant reviews and clicked that first from the dashboard.	It is presumed before the test that the participant was logged into a testing account. This means all functionality is accessible.

Participant Num	First Button Clicked	Comment	Notes
Participant 5	Packet Scanner Button	This user clicked the packet scanner functionality as their first button click.	It is presumed before the test that the participant was logged into a testing account. This means all functionality is accessible.

[Back to User Testing](#)

Heuristic Evaluation

From observing my participants when completing the other types of testing like the trunk test, think aloud and 5 second test I have gotten a clear understanding from watching and listening to how my participants interact with the GFree application.

From my findings while working with my participants I am going to apply Nielsen's heuristics and access GFree through the information gathered from my participants.

Visibility of system status

From my observed testing of the application, it is clear that there is a lack of visibility to the systems status specifically around the register and login system. This was evident when participant 5 entered their password incorrectly and no messages appeared to notify them of the error. This participant then immediately clicked on submit again thinking and assuming nothing had happened.

This is a clear indication that the applications notifications need to be updated to reflect the systems status and what is happening.

Match between system and real world

GFree uses clear concise English and in many areas has accompanying icons to clearly convey functionality and what will happen if a user clicks on some functionality. At no point throughout the application testing were participants confused by any of the wording found in the applications menus.

User control and freedom

GFree when considering navigation and movement through the application does a good job at allowing users to go back pages. GFree uses the Android devices back button instead of an application specific one. This means that when users want to navigate backwards, they use the back button they are used to and that they use every time they use their mobile device. This could also be considered in the Recognition rather than recall heuristic.

Consistency and standards

GFree uses well known and widely adopted functionality in the form of hamburger menus which are like sub menus in Android applications. When all participants were asked to find functionality that was not visibly present currently on the screen all participants instinctively clicked the hamburger menu because they recognise it as location for extra functionality.

The hamburger menu in all Android applications when present is a 'view more functionality' button.

Error prevention

Error prevention is a heuristic that was not rigorously tested. No errors besides the user login/register error with no notifications outlined in the visibility of system status heuristic occurred.

No errors or application breaking bugs occurred during testing.

However, I know as the developer of the application that there are error prevention pieces of code implemented in the application. This is done to keep the user experience consistent. For example, in the packet scanner there is code that prevents an error from occurring when an image is taken with no label in front of the camera.

Recognition rather than recall

GFree uses icons to assist users with identifying key functionality. This was done as a design choice to help users recognise a piece of functionalities purpose if the text alone was not clear enough.

Much like in the consistency and standards heuristic all participants recognised 3 horizontal lines as a hamburger menu, they instinctively went here when the functionality I asked them to find was not visible on their screen. This is because the hamburger menu and its icon are an Android standard.

Flexibility and efficiency of use

GFree for any of the form inputs like register, login and restaurant reviews uses the Android devices standard keyboard this means users will be allowed to use the keyboard they are used to and use for texting etc.

This means my application easily implements with no extra effort on the users behalf auto complete, quick typing and Microsoft Swift keyboard which all accelerates the users typing in the application.

Aesthetic and minimalist design

GFree by design was created with a minimalist approach to the user interface design with only the key pieces of functionality offered in a simple non confusing layout which participants easily navigated through during testing.

The GFree colour scheme was also designed with a dark primary colour alongside a light accent colour to make text and important information stand out on screen. This was actually commented on by participant 5's recording and said how their eye was drawn to the packet scanner and restaurant review sections which was dark text over a bright background.

Help users recognize, diagnose, and recover from errors.

Errors when they occur in GFree does not cause the application to crash or close randomly instead should an error occur it recovers itself. Notifications of the errors and what has happened is clearly notified to the user using the Android standard Toast messages.

[Help and documentation](#)

To help users of GFree I have a contact me button inside the hamburger menu section that allows users to get contact information to reach out to me for help or clarification around the application.

Actual documentation for the application will exist on the Google store page once the application goes live. This means that during the testing with my participants they did not have access to any documentation designed to assist with the application, but it will be created towards the end of development.

[Back to User Testing](#)

6.1. Exit Survey



GFree Exit Survey (SUS)

This is the system usability scale(SUS) questions for when testing of the GFree Android application has concluded.

Questions are measured from:

Measurement Scale!



I think that I would like to use this application frequently.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

I found the application overly complex.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

I found the application easy to use.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

I think the application is not suited for non technical people.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

I found the functionality offered by the application useful.

	1	2	3	4	5	
Strongly Disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Strongly Agree

I thought the system responded was to inconstant.

1 2 3 4 5

Strongly Disagree Strongly Agree

I think anyone who uses the application will learn it easily.

1 2 3 4 5

Strongly Disagree Strongly Agree

I felt the application was hard to use.

1 2 3 4 5

Strongly Disagree Strongly Agree

I felt confident using the application.

1 2 3 4 5

Strongly Disagree Strongly Agree

I need to learn alot to use the application fully.

1 2 3 4 5

Strongly Disagree Strongly Agree

Submit

[Back to SUS Questionnaire](#)

6.2. Other materials used

Any other reference material used in the project for example evaluation surveys etc.

Max Delaney

18100368

October Report

What?

This month I was tasked with researching and fleshing out an idea for my Software Project. I recorded my 5 minute project pitch video that outlined my idea for a restaurant review app alongside a recipe book that's solely catered to people with Celiac's disease.

The process for researching my application idea was easy enough and relatively stress free as it was an idea I had in my for quite awhile.

So What?

The fact I had my idea for quite awhile allowed me to get my project pitch video submitted early and allowed me to do more research around the technologies that could be used for my project.

Some things that currently worry me is the fact that many solutions for my project idea could be implemented using API's, naturally the use of an API doesn't give me marks so it feels like I'm trying to reinvent the wheel and in doing so I'm not sure where I'd even begin. Would I work on getting data on restaurants, provisioning the infrastructure to store said data? The list of doubts is pretty long.

Now What?

I'm currently waiting for my pitch video to be reviewed so I can be told if my idea has enough content to be usable or if I would need to make amendments. I have also just been assigned Emer Thornbury as my supervisor and I have to arrange a date to discuss my project ideas, I would also like to see if there are any other ideas my supervisor could give me or recommend.

Max Delaney

18100368

November Report

What?

This month I did a lot of work regarding other assignments. I didn't get enough time to work on my software project at all this month.

I had five large assignments and they have taken all my time.

So What?

I'm quite stressed out that I didn't make any progress this month. I'm also feeling the stress of the intense workload this year has to bring. Currently my situation is similar and hasn't changed as I still have a large plethora of assignments to do. They're all requiring 5-7 days work. There aren't enough days in the week.

Now What?

I'm currently way behind on quite a few of my modules and therefore my software project. There isn't much more I can do as I'm working on my projects up until 8pm at night and also using my weekends in order to meet tightly packed deadlines.

Max Delaney

18100368

December Report

What?

For December I got a lot of work done in the lead up to Christmas. I wasn't able to catch up with my supervisor for this month due to how busy and hectic my college assignments are and the fact I was working on my project on Christmas eve. I had no time and work to show until the late days of the month which naturally wouldn't be suitable to contact my supervisor on.

So What?

Moving forward into the new year and next semester of work I hope my college schedule is less hectic and more easily manageable as all attempts to get my time balanced between modules hasn't been possible. So many modules have had massive unexpected assignments that take a large amount of time.

Now What?

Once my remaining assignments I got in December that carry over to early January are completed I'll then reassess what needs to be done. I still have much functionality in my assignment to complete which I look forward to doing. For the pacing and what gets done first I will be referencing my Gantt chart I made in September.

Max Delaney

18100368

January Report

What?

For January I restructured my application by moving some functionality from behind the login/register wall. Previously an account was required to use any parts of my application. Now users can use all functionality that does not require a user to be known to the system, a good example of a feature moved from behind the requirement of an account would be the packet scanner.

Functionality like the restaurant review system that is still yet to be implemented is an example of functionality that will remain walled off by a login/register system as the user needs to be known for interactions with reviews.

This month I also had a valuable conversation with my supervisor Emer Thornbury. This allowed me to gain clarity and a vision on what I should focus on in order to implement my minimal viable product (MVP).

During this meeting I also discussed some of the functionality and requirements of the NLP functionality of my application and again my conversation with my supervisor cleared many of the doubts I had about implementing any form of an NLP.

I was able to compose a list of ingredients that will be required for my NLP to function, although simple this is a steppingstone to implementing my complete NLP.

So What?

As a result of the meeting with my supervisor I feel more confident in what is required for me to reach my MVP. No notable troubles or problems have occurred this month.

Now What?

Now I aim to begin working on the first iteration of my NLP which will be the logic that processes the users input from images taken of packets of food and outputs colour coded meaningful ingredients back to the user.

I will also create the documentation/illustration of how this works by utilizing flow charts.

The next coming month will be the month I ideally will get my completed NLP implemented and perhaps some other functionality along with it.

Max Delaney

18100368

February Report

What?

For February I designed and implemented the first iteration of my natural language processor (NLP). I compiled a list of ingredients in January and implemented them programmatically in February to be used by my NLP.

My NLP user interface was redesigned into what I would describe as a traffic light system with dangerous ingredients listed next to a red do not eat symbol. Potentially dangerous/use caution ingredients listed next to an orange warning symbol and a green tick next to a list of all the safe ingredients which can be consumed.

So What?

As a result of this month's work the NLP is now completed, and I can move onto the next step of my project. Completing my NLP gets me a step closer to achieving my minimal viable product (MVP).

Now What?

The next step will be implementing the restaurant review system. I will have to research and decide on an approach to the implementation of this functionality and most likely will use SQL commands to send reviews to and from a database. Currently I am thinking Firebase can be used for this but should any issues or limitations with Firebase popup I'll swiftly move to an alternate database solution.

Max Delaney

18100368

March Report

What?

For March I intended on implementing and completing my restaurant functionality. At the beginning of the month, I had no idea if Firebase supported the ability to read and write custom data or if I would have to find another database solution for storing my user reviews.

So What?

Currently towards the end of March I have not been able to complete my restaurant reviews functionality because of other assignments and their projects eating into my time. I have however researched that Firebase allows for reading and writing to a database using their supported Real Time database solution. This means no other database providers or solutions are required for my project and I can solely stick to Firebase for my backend requirements.

As for how difficult the Real Time database implementation will be, I can safely say it will be simple and easy as I have been able to test writing to the Real Time database but not in any meaningful way towards my minimal viable product (MVP) and overall project.

Now What?

Moving forward into next month, I am aware that I'm coming close to the end of my academic year and the deadline for my project. After the 10th of April I will officially be free of all other project work and from there, I will be able to implement my Firebase Real Time database solution. Once my reviews functionality is implemented my projects MVP will be fulfilled and I can discuss with my supervisor where to go from there in terms of meaningful development time and the implementation of other potential functionalities etc.

Max Delaney

18100368

April Report

What?

For April I finished the implementation of my restaurant review functionality.

At the beginning of the month I had begun by testing read and write functionality from Firebase with simple snippets of test code.

Over the course of April I adapted the simple code into the more complex final code to give users the ability to create, read, update and delete reviews from the database.

Aswell as Java code I had to implement the new views in XML to go alongside the new functionality.

For the first time I used Recycler views with card views which in my implementation could be considered a continuous table used to display reviews dynamically in a list.

So What?

With my minimal viable product (MVP) now completed I can begin refining my application in any areas that need it and can implement the final touches to my code base.

Now What?

I now plan to create unit tests using JUnit for each core piece of Java code so my application is thoroughly tested before submission.

This not only tests functionality and ensures consistency but can also be useful in the future should I revisit my project to test everything works the same as when I left it.

My documentation and final presentation are also important milestones that are constantly on my mind and I intend on working towards writing and refining the documentation now my MVP is completed.

6.4. Testing Informed Consent Form

GFree The Gluten Free Android Application

Consent to take part in testing.

- I agree to voluntarily participate in the testing of GFree.
- I understand that no information that could be used to identify me will be stored, with the exception of my signature.
- I understand that I can withdraw from testing at any time by request.
- I understand that all testing techniques can be recorded both visually and audibly.
- I understand that discussed topics and answers can be anonymously quoted in documentation and reports.
- I understand that participation involves testing an application on an Android device.
- I understand that I am free to contact the testing hosts to seek further information or clarification on anything asked or covered during or after testing has concluded.

Signature of testing participant

.....
Signature

.....
Date

Signature of host/researcher

.....
Signature

.....
Date

6.5. Focus Group Consent Form

GFree The Gluten Free Android Application

Focus Group Consent Form

- I agree to participate in the GFree focus group carried out by Max Delaney to aid with the development and research of the GFree project.
- I am aware of the purpose of GFree and its aims.
- I am aware of the topics that will be discussed during the focus group.
- I understand that no information that could be used to identify me will be collected or stored, with the exception of my signature below.
- I understand that I can withdraw from the focus group at any time by request.
- I understand that I am not obliged to answer any question, I can answer whenever I like using my own free will.
- I understand that discussed topics and answers can be anonymously quoted in documentation and reports.
- I understand that I am free to contact the focus group host to seek further information or clarification on anything asked or covered during or after the focus group has concluded.

Signature of testing participant

.....
Signature

.....
Date

Signature of host/researcher

.....
Signature

.....
Date