UAT Test Plan for [Cotton Leaf]

Team Member	Role
David Penfold	Team Leader
Ben Keeping	Innovator
Edward Gaston	Architect

Morgan Hodge	Architect

1 Scope

1.1 Objectives and business requirements – Edward Gaston

The first goal set by the team was to complete all the user stories we created at the start of the project

When we started the project, we set more goals to break the project down into sections to make the large task more manageable smaller tasks for the team to work through, the first of these was to create a prototype for the application which has a clean appearance which is easy to navigate.

We will measure the success of this goal through our biweekly meetings with the client where we will ask for feedback on the prototype and see what feedback the client provides.

Another goal the team set was to test and train an AI model that was provided by the client, after this we would attempt to implement it so that a leaf could be scanned, and a result given as needed.

The team set another goal which was to create a user friendly app with quick navigation. The navigation bar was used to meet this goal as it provides a simple and easy to understand way for users to navigate around the application.

1.2 Scope – Edward Gaston

Our applications purpose is to calculate if a cotton leaf is healthy or diseased after scanning a photo of the leaf.
For this UAT test we'd like to

- Verify that the camera works correctly and stores the photo so it is ready to be scanned.
- Get feedback from the questionnaire that the application is user friendly and easy to navigate
- Verify that the user is able to edit their user details such as their password

We verified that the camera was working as intended by testing all the features on the camera such as switching the flash on and off, switching between the front and back facing camera and being able to upload a photo from the camera roll instead of taking a photo.

We verified that the users are able to edit their user details such as their password by testing on multiple different accounts, testing if the new password criteria works and only allows users to create a password when they have a number and a capital letter within the password.

2. Testing Team – Morgan Hodge

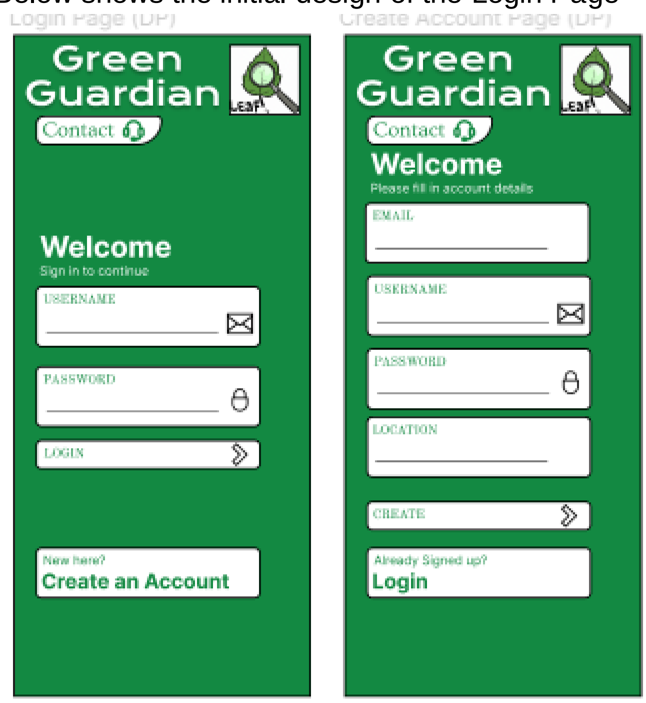
Name	Responsibilities
Morgan Hodge	UAT Coordinator – handled communication between end users and QA team, created Questionnaire Setup feedback system Tester Recruiter – found users that are willing to take the survey for our application
David Penfold	Wrote UAT documentation Created test data for forum Designed test cases
Edward Gaston	Alpha Tester – Tested the application Beta tester - Tested the application
Ben Keeping	Alpha test user – before the application was launched to any test users, David acted as a user of the app and found multiple bugs that needed to be fixed

3 Milestones and deliverables

3.1 Design & testing process – Edward Gaston

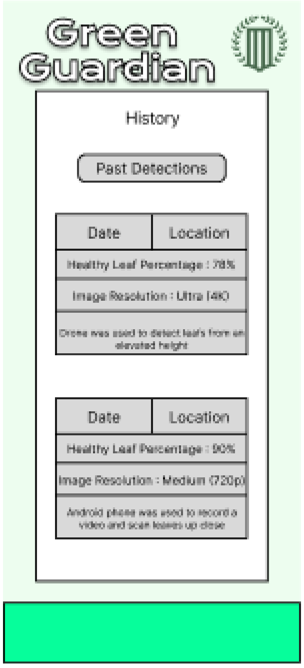
The design stages started with our low fidelity prototype which was created on Figma. We chose Figma because it would allow us to have a more accurate copy of what the application could look like, compared to a wireframe which isn't as accurate. The Figma low fidelity prototype had multiple versions till we settled with the current design.

Below shows the initial design of the Login Page



Below shows the next version of our low prototype design, we made some design changes such as changing the color scheme, below shows the History and Settings pages.

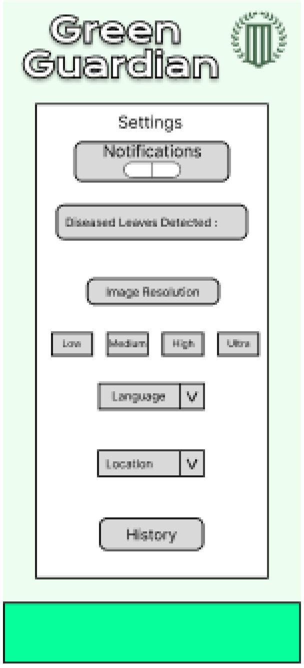
History Page (BK)



Date	Location
Healthy Leaf Percentage : 78%	
Image Resolution : Ultra (4K)	
Drone was used to detect leafs from an elevated height	

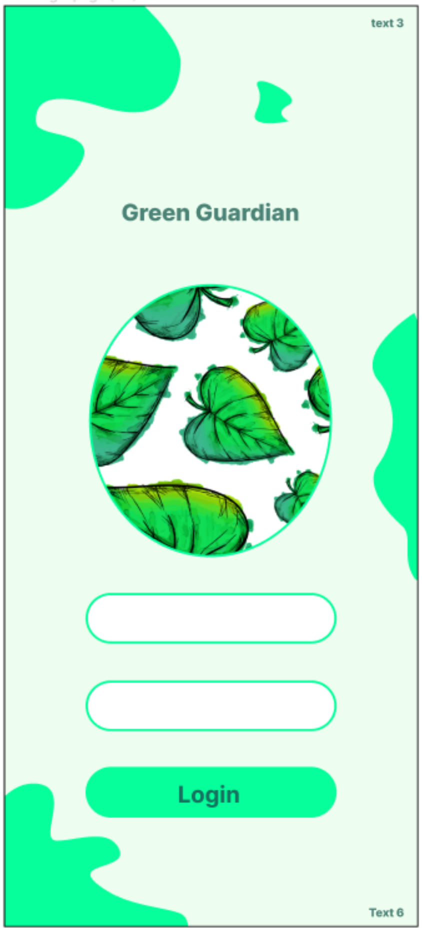
Date	Location
Healthy Leaf Percentage : 90%	
Image Resolution : Medium (720p)	
Android phone was used to record a video and scan leaves up close	

Settings Page (BK)

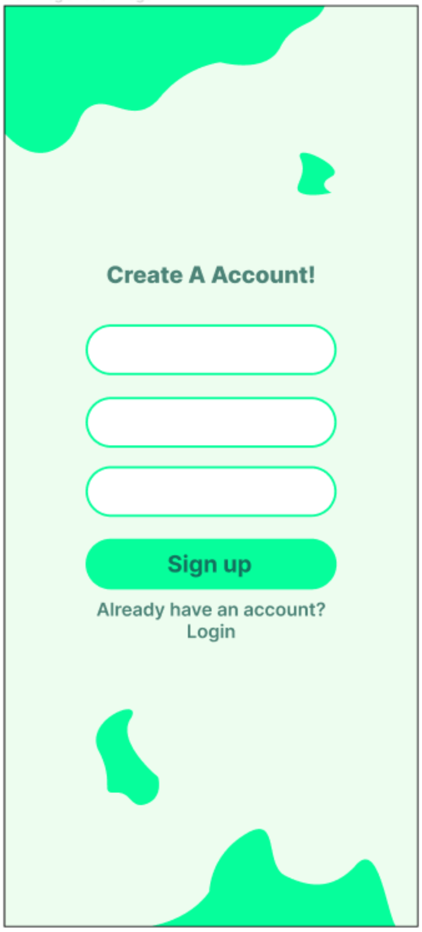


Below shows the next version in the design process, in this version it shows the new login page after a redesign based of client feedback

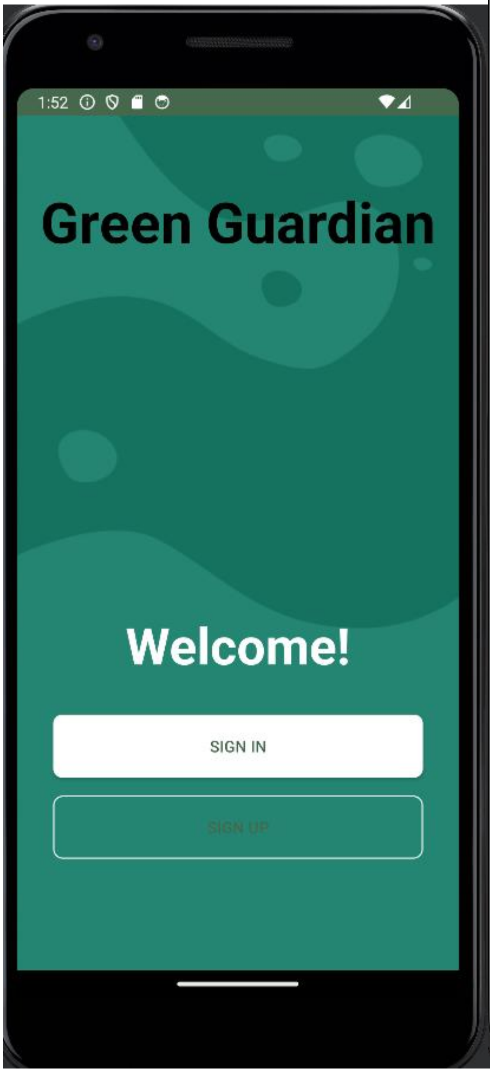
New login page (EG)

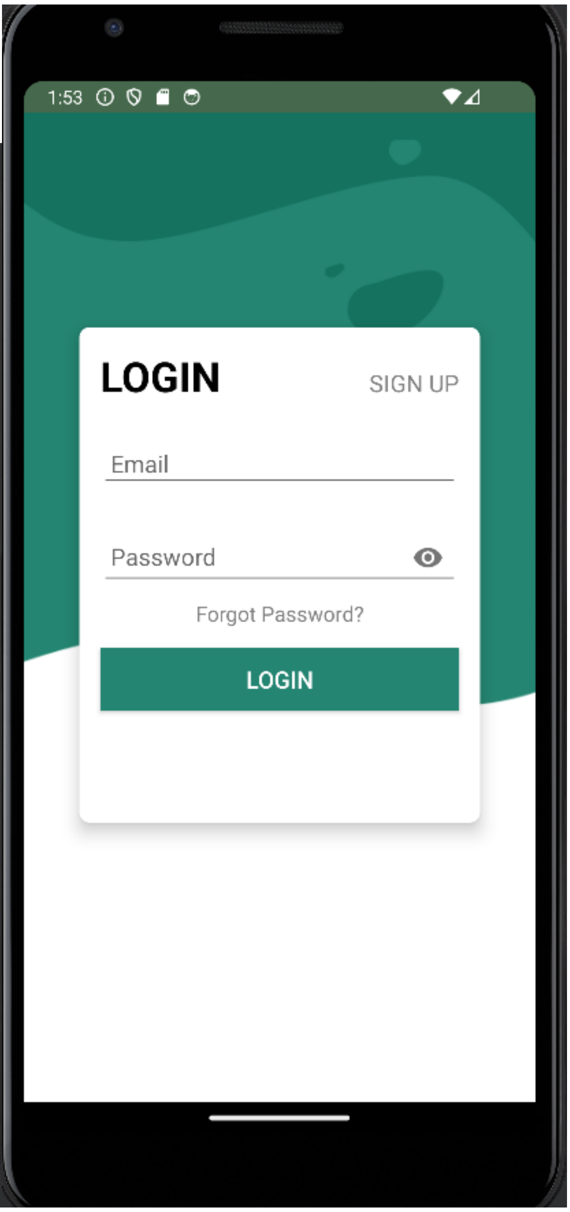


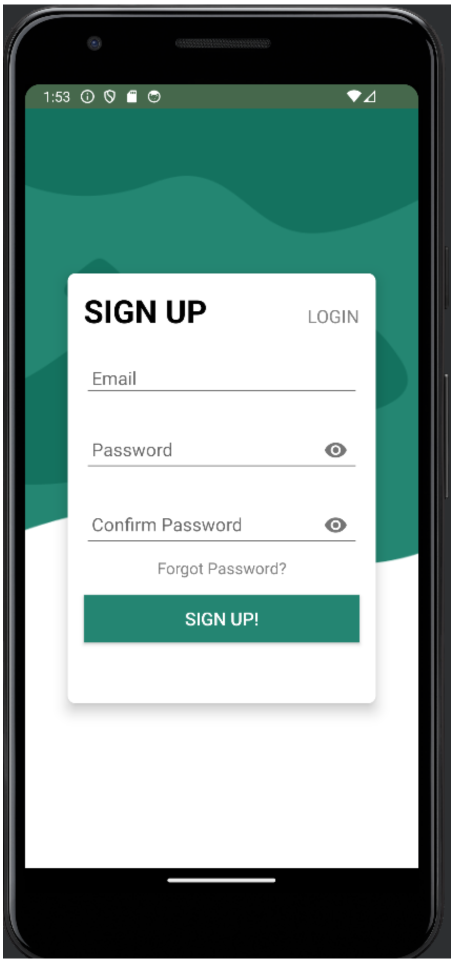
New Registration Page



Below shows the final designs of our login systems application. It highlights the design changes that can be made in the transition from a low fidelity prototype to a high fidelity prototype.







We underwent testing for the application and created test plans to help us test it. The testing environment for the application was done instead android studios emulator. This allowed us to test for bugs on multiple versions of phones hopefully resulting in less bugs and a better user experience on the application

Within the testing plan we tested for bugs in every aspect of the application, every navigational feature and feature built into the app.

3.2 Staging environment – David Penfold

Due to the use of android studio, it proved hard to create a staging environment. To fix this problem we held testing in person in a controlled environment. The requirements for this were as follows:

- No outside interaction between any other people.
- No interactions between test users before using the app.
- Users test the app individually, alone and undisturbed.
- Compatible Android Phone with no hardware or software problems.

3.3 Training – David Penfold

This is the process taken to properly train the users for beta testing. Without this necessary training, the results will be inaccurate and incorrect. This is how we first went about setting up the meetings.

We've had Ed set up the meetings via zoom and walk them through what the new features are and what needs testing and how to make the most out of it. This is the structure of each meeting and dates:

- Meeting 1 - 30 minutes - present the new features to testing
- Meeting 2 - 1 hour – this included how to log to access the staging environment and best ways on how to test the new feature.
- Meeting 3 - 1 hour - how to share results of findings.
- Deadline – April 20th

3.4 UAT Execution – Edward Gaston

Execution will take place over 3 days. During these days we will ensure that all aspects of the applications have been tested multiple times to ensure that they pass.

The stages of testing will be on different parts of the application. On day one we will test all the aspects of the code that relates to the Firebase login system. This would include the login, signup and the profile settings page. We will test that users can create users, sign into users and edit their passwords to their accounts without bugs and without errors. The only errors that the user should receive if that when they are creating or changing their password if they don't match the password creation criteria.

The second stage of testing will be on the camera aspects of the application. We will spend the second day testing that all features on the camera work without bugs and on all types of android phones. After the camera has been used, we will also test for the photos to see if they have been stored correctly.

On the final stage of testing, we will test every other page of the application for bugs and errors. This includes all buttons functioning correctly and all information being displayed correctly.

3.5 Reporting and Data Analysis – Morgan Hodge

Testing was conducted in April this year. This was the same month when the survey was conducted. Individual test case results were stored in a table, data stored included what testers struggled with, areas of improvement and general feedback.

Tester	Date	Issues	Area of improvement	Feedback

Josh Williams	3.4.2024	I could not identify any issues	The font size could be changeable	Can I Login as a Guest to avoid making account?
Mitchell Holmes	3.4.2024	Can't delete account	Make it so the account can be deleted, and all data wiped	Will my data be protected by this app?
Freya Darcy Drake	8.4.2024	No issues identified	N/A	Will the app give me the same result if I scan the same leaf twice?
Luke Willcox	8.4.2024	No colour blind options	Add some more accessibility features	Will there be restrictions on the kind of lighting I have to scan leaves in?

4 Environmental Requirements

4.1 Hardware Requirements – David Penfold

The app has been designed to function with the lowest hardware requirements. This is because it may be harder to gain access to devices with high processing power in countries such as Pakistan, especially rural, where it has been intended to be used.

Hardware Requirements:

- Minimum storage of 50MB – this is due to the AI code being stored locally without their device. The advantage of this is that the user can use the app without an internet connection. In the future the AI code could be stored in the cloud to save storage, however this would require an internet connection.
- Minimum API level of 24. This dates to devices up to 8 years old (2016), allowing a range of users to access the App.
- Minimum RAM of 512MB – this is due to the app passing results through the AI locally.

Alternative usage:

- If a user does not have access to a camera on their device, they can still choose an image from their photo library. The user would need to import the photo from another device such as a camera or another user's device.

4.2 Software Requirements – David Penfold

There is no reason for the App not to run as intended if the hardware requirements are already met. This is because if an android device is 2016 or newer then the necessary software requirements will be met too. There are no additional libraries needed or third-party frameworks. However, a few other requirements need to be met:

- Permissions: The user must have the correct control to accept the permissions. For example, if a parent has parental guidance on their child's account, then they may not be able to access the camera by accepting permissions. Alternatively, the user may be using a work phone where the camera has been disabled- this again will not allow permissions to be accepted. This is not only relating to the camera but in some cases the camera library.
- Offline Support: For the app to function offline as intended, the android device must have offline support. Furthermore, some android devices implement cloud storage for applications. This means the app would not be accessible when there is not a stable internet connection.

5 Features to be tested

5.1. Scanning Page – David Penfold

The scanning page is a crucial part of our app and without it the application will not function as intended. This can be broken down into 6 elements:

- Permissions for Camera Usage – when first opening the scanning page, the app should ask the user if it has permission to use the camera. If it does not then the camera will not be used, alternatively if it does then the camera should work as normal.
- Permission for Photo Library – if the user clicks on choose photo from library, the app should ask the user if it has permission to access the user's camera library. If it does not then the camera library will not open, if it does then the camera library will open as normal.
- Front Camera Functionality – the front camera should show a display of what it is seeing, and an image should be taken when the take photo button is pressed.
- Back Camera Functionality – the back camera should show a display of what it is seeing, and an image should be taken when the take photo button is pressed.
- Flash Functionality – this function, when turned on and using the back camera, should take a picture with the flash on when the photo button is pressed.
- Choose Photo from Camera Library – when choosing the photo from the photo library, the user should be taken to their camera roll. From there the user can select an image of their choice.

5.1.1 Scanning Page Pass/fail criteria – David Penfold

Test Type	Pass/ Fail
Permissions for Camera Usage: Pass – Permissions are asked for usage of the Camera. Fail – No permissions are asked, or permission is still granted after selected no to usage.	
Permission for Photo Library: Pass – Permissions are asked for usage of the Photo Library. Fail – No permissions are asked, or permission is still granted after selected no to usage.	
Front Camera Functionality: Pass - correctly displays what is being shown. Fail – display is incorrect (format, facing the front way, inadequate quality) or there is no display.	
Back Camera Functionality: Pass - correctly displays what is being shown. Fail – display is incorrect (format, facing the front way, inadequate quality) or there is no display.	
Flash Functionality: Pass – when a photo with the back camera the flash is used. Fail – no flash is applied when a photo is taken with the back camera.	
Choose Photo from Camera Library: Pass – the user is taken to the camera library and an image can be selected. Fail – the chosen photo from library does not take the user to the library or a photo cannot be selected.	

5.2. Sign Up Page – Ben Keeping

To be able to access the cotton leaf application the user must first sign up to the application, this requires the user to sign up with a valid email alongside a password that meets specific requirements:

- Sign Up - The user must enter all their information correctly. This includes entering a valid password which meets the requirements (such as containing a number). If the requirements are not met for any part of the account creation, then the account will not be made.

5.2.1 Sign Up Page Pass/fail criteria – David Penfold

Test Type	Pass/Fail
Incorrect Sign-Up credentials test: Pass - Error message is displayed informing the user that the criteria for account creation has not been met. Fail - User account is created, despite incorrect criteria.	
Correct Sign-Up credentials test: Pass - the user inputs information for a valid account, the account is then created. Fail - User enters valid information for account creation, however an account is not made.	

5.3. Navigational Bar – David Penfold

The navigational bar is present in every single page; therefore the functionality is essential:

- If a user is on a page and tries to access that page via the navigational bar, then the app should not change and the app should not move. The purpose of this is because it may cause the app to crash or even become stuck in a loop.
- Layout – the navigational bar should stay consistent across all pages. This means the user can quickly click between pages and the navigational bar should not move at all.
- Camera button – the scanning page is the most important part of the app. Furthermore, it should be accessible from each page. When the user is on the scanning page, the camera button should switch to a slightly different looking button. This indicates that the camera is functional and ready to take a picture.

5.3.1 Navigational Bar Pass/fail criteria – David Penfold

Test Type	Pass/ Fail
Accessing same page: Pass - the application does not change page and the button is unresponsive. Fail – the application crashes, defaults to the home page, causes a black screen or reacts in any unintended way.	
Layout:	

Pass – the navigational bar keeps a consistent shape and layout through each page. Fail – the navigational bar changes shape and layout when navigating between different pages.	
Camera button: Pass – the camera button works on each page and changes shape when reaching the scanning page. Fail – the camera button does not take the user to the scanning page, or the camera button does not change shape on the scanning page.	

5.4. Login Page – David Penfold

Once a user has successfully signed up, they can use their credentials to login to the app:

- Log in – once the account has been created, the user can use their credentials to log in. The user should not have to sign up to the app each time, but instead can log in with their created credentials.

5.4.1 Login Page Pass/fail criteria – David Penfold

Test Type	**Pass/ Fail**
Login test with valid credentials: Pass – the user can login to the App with their correct credentials. Fail – the user cannot login to the App, despite correct credentials.	
Login test with invalid credentials: Pass – the user is unable to login to the App with incorrect credentials. Fail – the user can still login to the App, despite incorrect credentials.	

5.5. Logout Page – David Penfold

The logout page allows the user to complete multiple actions. Of course, the user can use this page to logout but additionally the user can use this page to change their password:

- Logout Button – the user can use this button to logout of the application, the user will be taken to the login page. From here the user can either sign up for a new account or log into an account. This is necessary for multiple reasons such as having multiple accounts for different locations or perhaps letting another user borrow your device.
- Edit Password – this feature requires the user to enter their email address, which will then allow the user to enter a new password which they want to update their account with. The new password must still meet the criteria such as containing a number.

5.5.1 Logout Page Pass/fail criteria – David Penfold

Test Type	Pass/ Fail
Logout test: Pass – the user can logout successfully and is redirected to the login page. Fail – the user cannot logout or is not redirected to the login page.	
Edit Password Correctly: Pass – the user edits the password with the correct email and password credentials, the password is changed. Fail – the user edits the password with the correct email and password criteria, yet the password is not changed.	
Edit Password incorrect email: Pass – the user enters the incorrect email but corrects new password criteria. Therefore, they are unable to edit their password. Fail – the user enters the incorrect email but corrects new password criteria. However, they are still allowed to change their password.	
Edit Password with incorrect criteria: Pass – the user enters the correct email but incorrect new password criteria. Therefore, they are unable to edit their password. Fail – the user enters the correct email but incorrect password criteria. However, they are still allowed to change their password.	

5.2.2. Test cases – Morgan Hodge

Test case Number	Test Case	Test Description
1	App Launch	Test the app launch time Verify that the app launches without crashing
2	Image Capture	Back Facing Camera works Flash Works when taking photo
3	Registry	A new user can create a username and password User Can log into account that was just created

4	User Interface	Test user interface for ease of use Verify the buttons are responsive and lead to correct pages
5	Language Translation	English translation is correct Urdu translation is correct
6	Results	A result is displayed after the user scans a leaf

5.3. Features to avoid testing – Morgan Hodge

Some features do not need to be tested as we know these will work for certain. To avoid testers being sidetracked we will specifically document what features must not be tested. These features shouldn't be tested as we want to center their attention on more specific and complex functions.

5.3.1. Navigation Bar

We know for a fact the navigation bar works as this has been constantly tested throughout the entirety of the project and this was one of the first features that was implemented. This would be a waste of their time if each tester had to test this function.

5.3.2. Buttons

The buttons are a stable feature in this application, each button's logic is correct and is tested every time the app is launched. The team has thoroughly tested each button on each page and can for certainty that they work. Because of this reason we believe this is another feature that should be avoided during testing as it would be a waste of time.

Client Signature here:
To be updated before the 16th.