

## **Mobile App Design Document (UFCF7H-15-3)**



# **HOOT**

## **Mobile Applications**

Cerys Mooney, Laura Robbie  
and Luke Hammond

# **Hoot**

**Student Name:** Cerys Mooney, Laura Mae Robbie, Luke Hammond

**Student Number:** 21003564, 21010578, 21013675

# Table of Contents

---

<b>Requirements.....</b>	<b>3</b>
Context:.....	3
User Persona/Journey:.....	3
Priority Guide:.....	6
Intent Frame:.....	6
Functional / Non-functional Requirements:.....	7
UI Requirements:.....	8
<b>Wireframes.....</b>	<b>8</b>
Diagrams:.....	8
User Flow:.....	9
Scale / Orientation:.....	10
<b>Composites.....</b>	<b>11</b>
Mock-ups:.....	11
App Icon:.....	12
Colour Schemes:.....	12
UI Asset Scale:.....	12
<b>Bibliography.....</b>	<b>13</b>
Reference list:.....	13

# Requirements

## Context:

Designed by a group of three Digital Media students: Laura Robbie, Luke Hammond, and Cerys Mooney, Hoot is a mental health app where users can write down their feelings for the day and listen to calming or upbeat sounds to improve their mood. It is mainly targeted towards students who may be facing stress or overwhelmed about their studies. Hoot is a place where they can express emotions that may otherwise be difficult to share with their peers.

## User Persona/Journey:

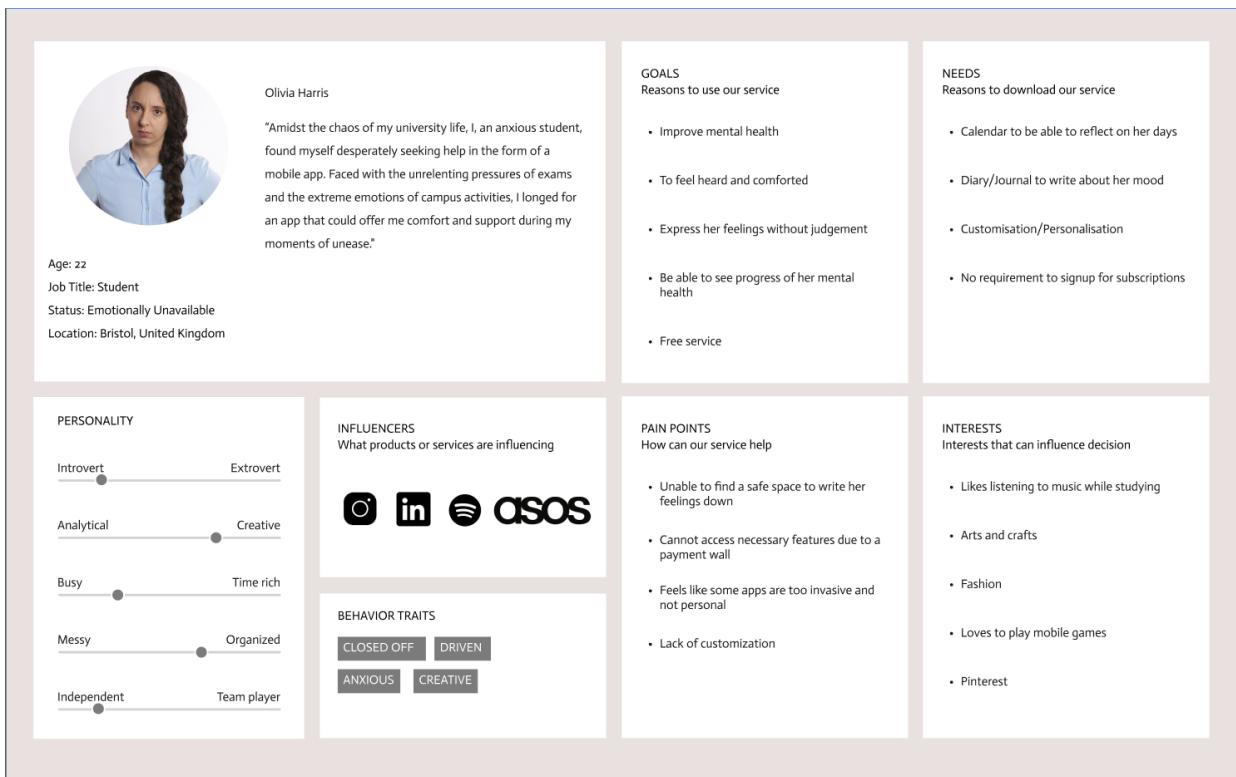


Figure 1 - Olivia Harris User Persona

To begin our work, we created a user persona, “Olivia Harris.” This persona helps us focus on our app and its needs. Our app should reflect on the needs/goals of our persona as well as push beyond that boundary to have an all encompassing application. However, our app needs to meet needs for all types of users and be accessible, therefore, some goals may not be met or perhaps altered in the production stage.

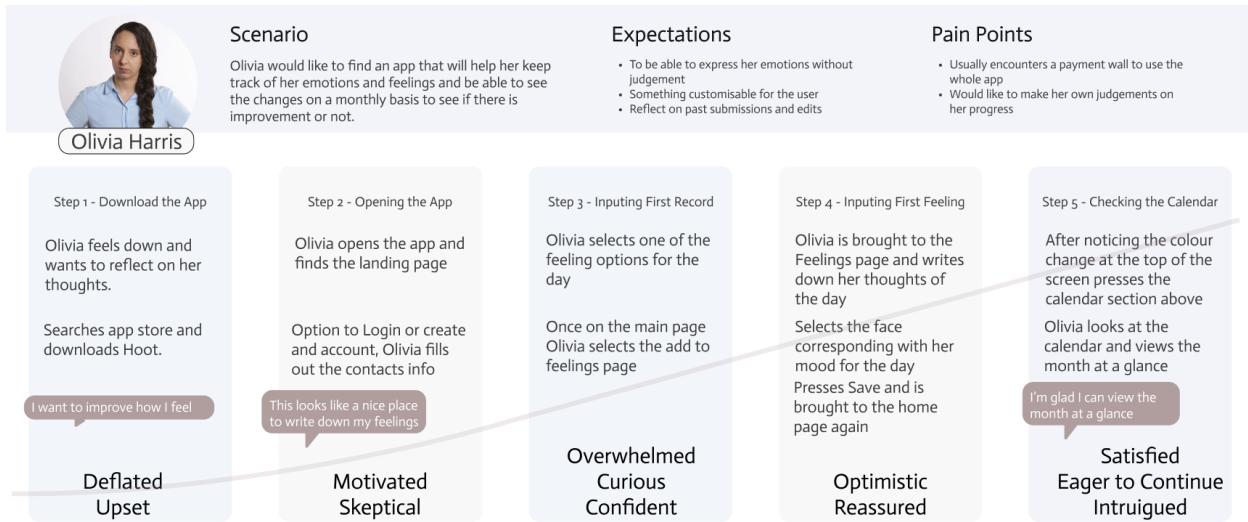
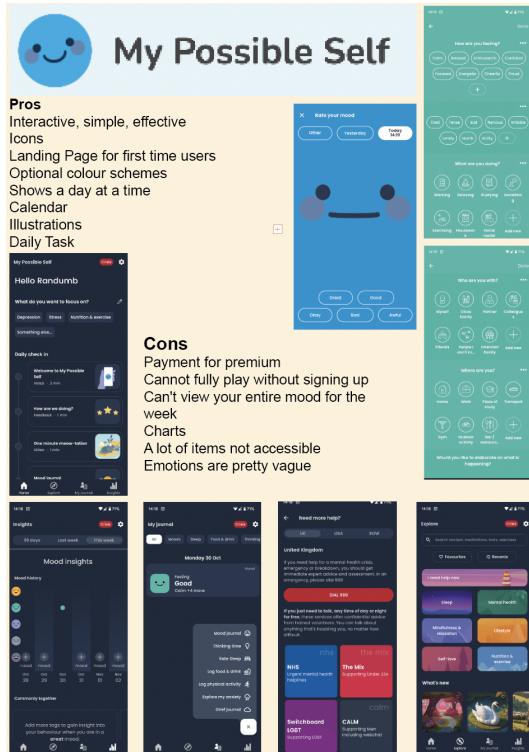


Figure 2 - Olivia Harris - User Journey

The user journey demonstrates what the possible actions and path a user will take upon using our app. In Figure 2, the user is experiencing a range of emotions, which is displayed at the bottom of the text but also through speech bubbles of comments they would make at every stage. These comments help portray how our app should feel. This will be accomplished through feedback on the app. When working in unison with the User Persona, our User Journey, will also help in grounding our application.

## Initial Research/Competitors:



### My Possible Self -

The app was a big influence on our creative inspiration and helped us in designing possible elements for our own app. We liked the animation and interactivity the app provided, which we wanted to emulate, as well as the optional colour schemes, however ours were to be more grounded in the application and its purpose.

We found in our research that many of the applications had elements stuck behind a payment scheme, meaning the app had a lack of accessibility. Additionally, My Possible Self was not accessible without an account.

Figure 3 - My Possible Self Competitor Analysis

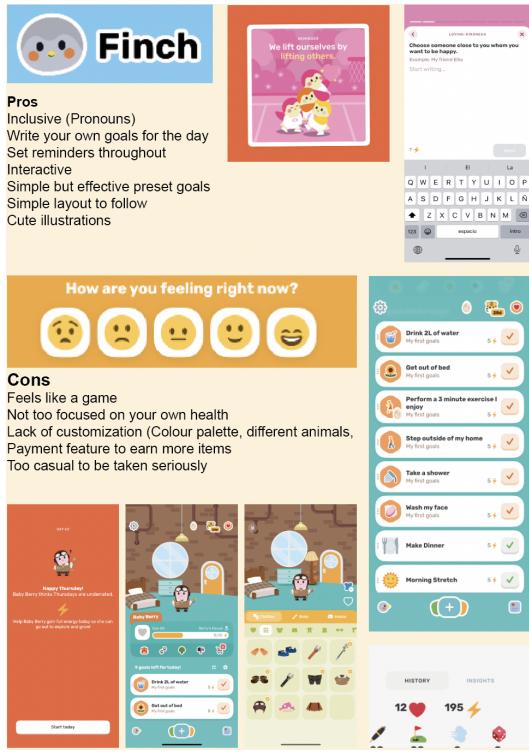


Figure 4 - Finch Competitor Analysis



Figure 5 - Reflectly Competitor Analysis

## Finch -

Finch was a unique experience, similar to a game that users can play and express their emotions. Great takeaways from this app was the accessibility and inclusivity the app provided, as seen in their pronouns. Additionally, the app keeps itself inherently simple, this avoids being intimidating towards users and more welcoming, keeping interest for the user to comeback.

However, due to the app feeling like a game, it may lose out interest for audiences due to it feeling too casual. Furthermore, the lack of customisation and many features being locked behind a pay-wall, users may feel less attached/loyal to the app.

## Reflectly -

Very similar to My Possible Self, due to it containing many interactive elements, this was a big inspiration for our app. The layout for this app was simple and very accessible and when paired with the customisation elements became creative and exceeds its initial simplicity.

However, like all the apps, Reflectly cannot be used without an account and many elements are not accessible without payment. Additionally, the app is rather vague on emotions for users to pick, thus possibly not fulfilling its inherent requirement of a mental health app.

Priority Guide			Priority Guide			Priority Guide			Priority Guide		
Create Account			Landing Page			Welcome Page			Login Page		
Element	Content	Intent	Element	Content	Intent	Element	Content	Intent	Element	Content	Intent
Button	Back Button	Allow the user to travel to login page	Text	Welcoming Text	Call to action for user to answer question	Text	Welcome Text	Introduce and welcome users to the app	Text	Login/Sign-up Title	Inform the user of where they are on
Text	Create Account Title	Inform the user of the page they are on	Video/Image/Animation	Image	Reflect how the user feels based on feedback	Image/Video	Animation	Aware the user of interactivity	Interactive UI	Login Status	Allow users to login
Button	Confirmation Button	Users can confirm their account creation	Interactive UI	Slider	Allows the user input their mood	Button	Button	Move the page onto the next scene (could also be on a timer)	Interactive UI	Create Account	Link the user to create an account
			Text	Emotional Status	Text will change in response to the slider - showing different emotion						
			Button	Confirmation Button	Users can confirm their input and move on to the next page						
Priority Guide			Priority Guide			Priority Guide			Priority Guide		
Goals for Today			Calendar			Home Page			Expand on Feelings		
Element	Content	Intent	Element	Content	Intent	Element	Content	Intent	Element	Content	Intent
Text	Title	Inform user of their current page	Text	Title	Inform users of their current page	Text/Interactive UI	Mood Status	Will inform users on their response as well as allow them to change it	Text	Title	Inform users of current page
Image	Image	Break up text and set mood	Interactive UI	Calendar	Allows users to see past posts and create new posts	Image/Interactive UI	Weekly Bar	Can inform users of weekly mood and the ability to take them to the calendar page	Text	Date	Allow users to type in current date they are filling mood out for
Text	Header	Title of goal	Nav Bar	Nav Bar	Move to different pages	Pop-Up UI	Expand on Feelings	Written diary entry	Interactive UI	Note Section	Users can expand on their feelings in written form
Button	Interactive Button	Users can tick off their goals when achieved				Pop-Up UI	Goal Checklist	Users can add goals or tick off goals completed	Interactive UI	Pick Emotions	Users can select how they are feeling
Button	Add Your Own	Allow users to add their own goals				Nav Bar	Nav Bar	Users can move to different pages	Interactive UI/ Image	Attach Picture	Users can attach a picture they want for their diary post
Nav Bar	Nav Bar	Move to different pages							Interactive UI	Save Button	Users can save their entry to the calendar
									Nav Bar	Nav Bar	Users can move to different pages

Figure 6 - Priority Guide

The priority guide aids in creating the layout for pages of the app by displaying a hierarchical portrayal of elements on the page. By accomplishing this, we have achieved an initial lo-fi prototype for our app.

## Intent Frame:

After creating the priority guide, we then designed an intent frame to visually showcase where our intended content would be based on the phone's screen. This gave the group a detailed insight on how each page was to be laid out and why it was necessary to put our content in each specific spot.

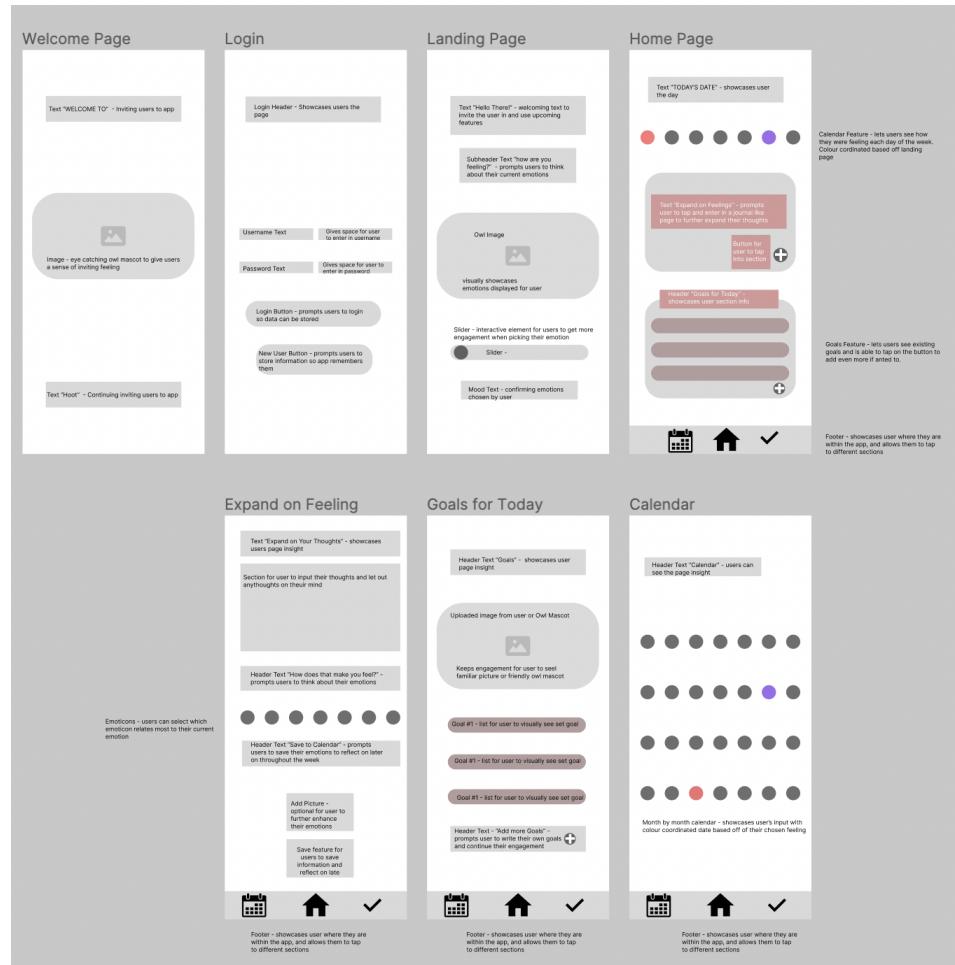
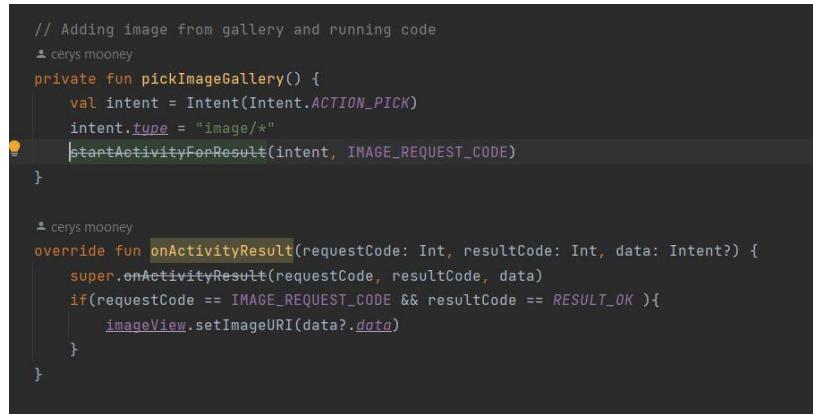


Figure 7 - Intent Frame

## Functional / Non-functional Requirements:

We have added an option to add an image from the user's gallery to the Expand on Feelings page. After searching through many different sources to try and fix this issue, we could only find the version (Figure 8) that worked in our app. Despite searching for something else that works, we unfortunately had to settle with the deprecated version.



```
// Adding image from gallery and running code
▲ cerys mooney
private fun pickImageGallery() {
    val intent = Intent(Intent.ACTION_PICK)
    intent.type = "image/*"
    startActivityForResult(intent, IMAGE_REQUEST_CODE)
}

▲ cerys mooney
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    super.onActivityResult(requestCode, resultCode, data)
    if(requestCode == IMAGE_REQUEST_CODE && resultCode == RESULT_OK ){
        imageView.setImageURI(data?.data)
    }
}
```

Figure 8 - Deprecation in Expand on Feelings Page

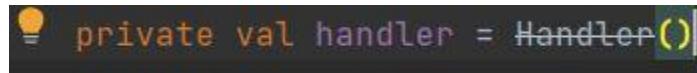
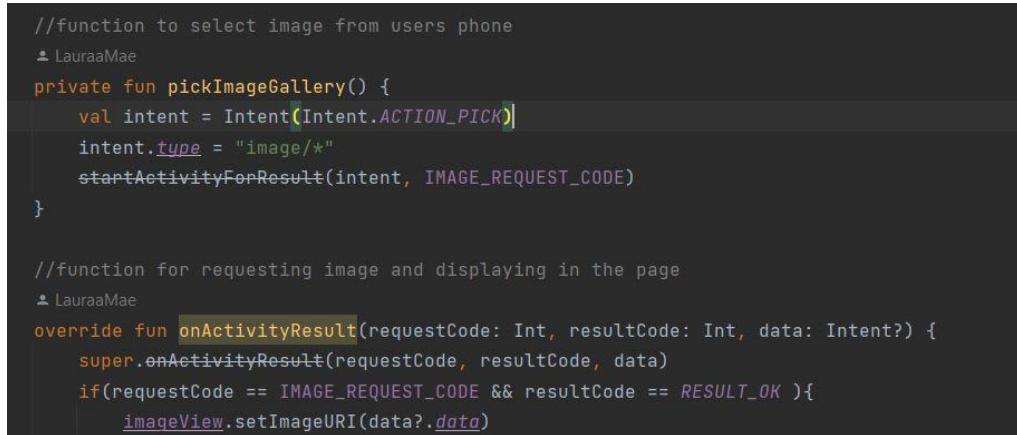


Figure 9 - Deprecated Code



```
//function to select image from users phone
▲ LauraaMae
private fun pickImageGallery() {
    val intent = Intent(Intent.ACTION_PICK)
    intent.type = "image/*"
    startActivityForResult(intent, IMAGE_REQUEST_CODE)
}

//function for requesting image and displaying in the page
▲ LauraaMae
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {
    super.onActivityResult(requestCode, resultCode, data)
    if(requestCode == IMAGE_REQUEST_CODE && resultCode == RESULT_OK ){
        imageView.setImageURI(data?.data)
    }
}
```

Figure 10 - Deprecated Code

We looked at using a built in android calendar for the app, despite having the calendar functioning in our app, we were unable to store the data that the user entered. Due to the nature of our app we felt as if this was an important factor and was best to alter the app by taking away the calendar and adding in another page to help which plays sounds for the users. We feel this page was a much

better addition to the app, as it can help relax the user, and it is also a fun way to help keep the user engaged with the app.

## UI Requirements:

Navigation bar - Built a navigation bar to move between pages, these will be accompanied by buttons which can also navigate pages

Onboarding - An initial introduction to the app as well as allowing the user to customise their colour palette

Buttons - Will enable users to customise the page

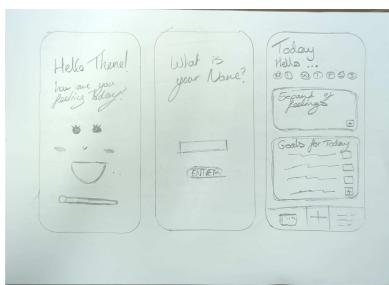
Editable Text - This text will be accessible for the users to edit and will be displayed on the page once saved

Adding Images - On our expand on feelings page, we have enabled users to add images from their gallery

## Wireframes

---

### Diagrams:



As a group, we quickly sketched up a low-fidelity wireframe on a few sheets of paper to confirm the layout that was discussed in both the priority guide and intent frame. Although minimal, the sketched wireframes demonstrated how the app would visually look for a user. It also gave the group even more of an insight of how the content would fit within the given layout.



Figure 11 - Paper Wireframes

## User Flow:

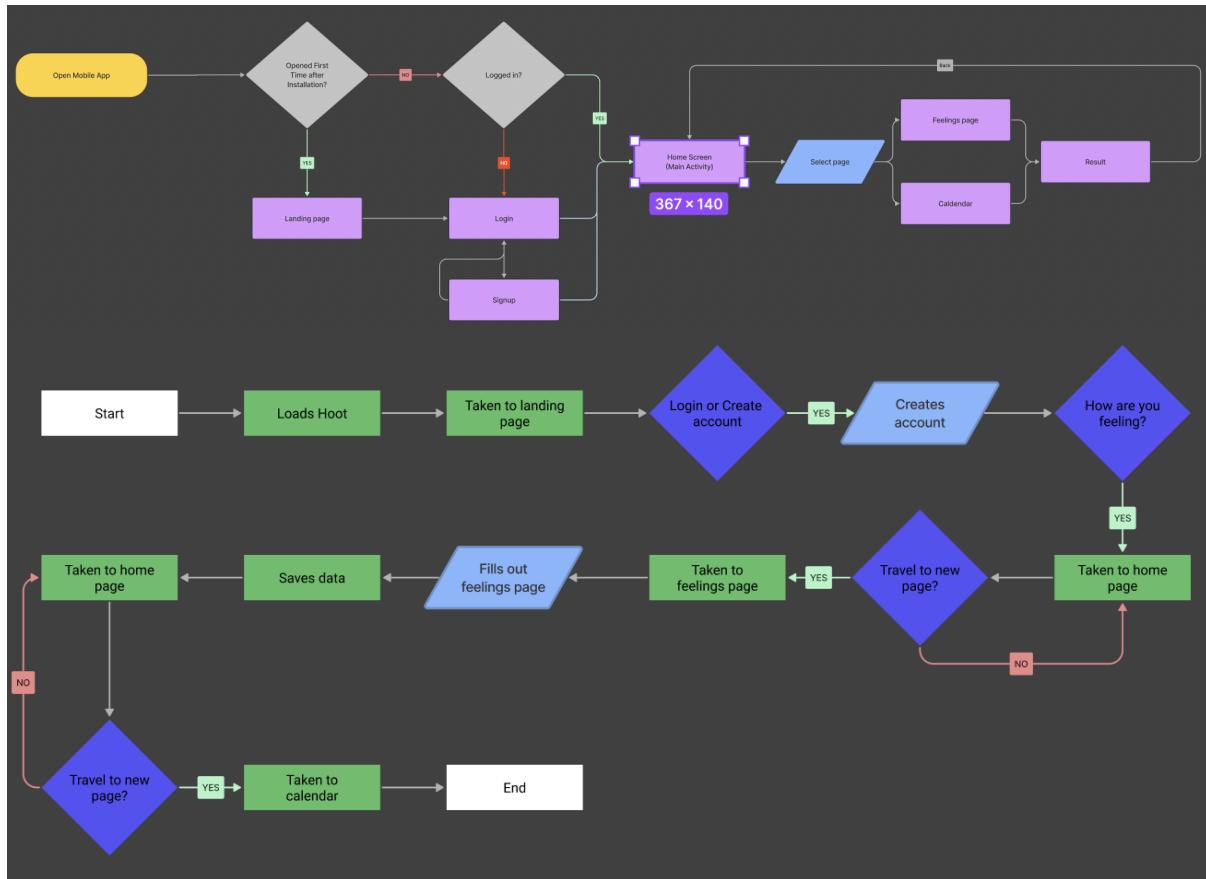


Figure 12 - User Flowchart

The flowchart in Figure 12 demonstrates how the app will run on a more technical level compared to the user journey in Figure 2. The flow chart showcases the users actions and possible decision making. It also demonstrates how our app is accessible to all without creating an account and without having to pay for content. The data and information is to be locally saved to the device, therefore not requiring the need to create an account.

## Scale / Orientation:

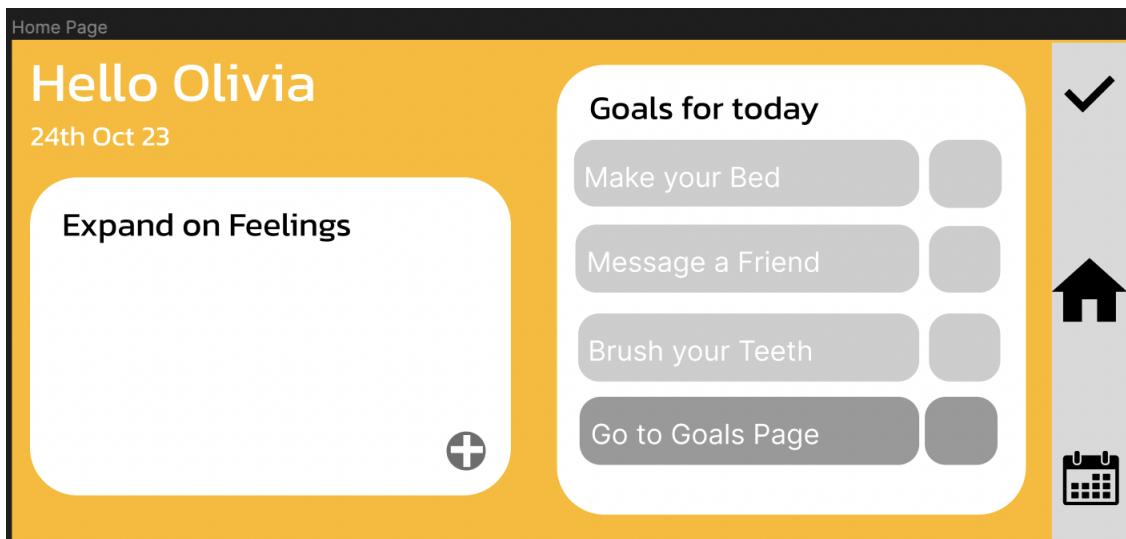


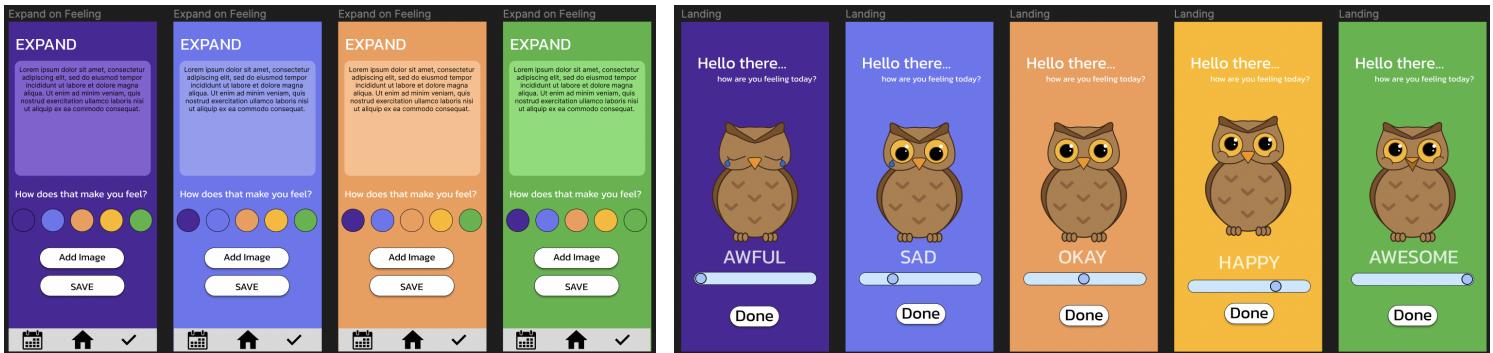
Figure 13 - Landscape Design of Hoot

We considered an app that was versatile with landscape and portrait, and produced a landscape view of the home page, however after looking at other apps with similar functions, and due to the clean look we were going for, we decided to lock the app into place in portrait mode. Furthermore, due to the amount of elements on one page, the landscape design was insufficient for users and hindered the ideal experience we wanted to create.

Additionally, we looked into creating the app on as many devices as possible, this included devices outside of the mobile phone, such as watches, tablets and desktop. However, after designing the elements on each, it was rather complex as well as lacking in accessibility. Therefore, we restricted the app to run only on tablet and mobile.

# Composites

## Mock-ups:



Figures 14 & 15 - Variations of screens



Figure 16 - Variation of home screen

Based on what the user selects on the slider in figure 14, the background changes in correlation to whatever emotion was chosen. This is then saved and is used throughout the app, until the player optionally chooses another emotion shown in figure 15. This will change the value of the global variable for the background colour accordingly.

## App Branding:

As a group we wanted to create a memorable app brand, something to help it feel homely and safe for the users. We looked at using an owl for our app as he represents deep thinking and organisation, this is entirely what our app aims to do. Also by adding on small features like emotions and the twinkle in the eye, makes Hoot more personable.

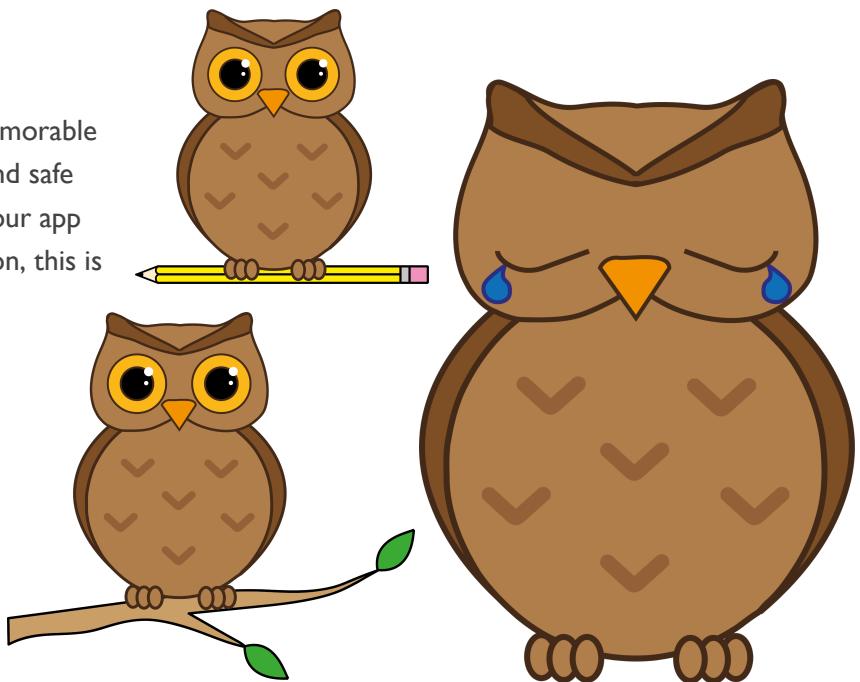


Figure 17 - Owl variations

## **App Icon:**

We decided to use the owl from our branding as our app icon. Doing this reiterates our brand and gives the user a fun and interesting look on their home screen. Additionally, we used the happy version of the owl to draw attention to the app, to help motivate users to use the app.



Figure 18 - App icon

## **Colour Schemes:**

We selected a few colours to work with to represent different emotions, we looked at blues and purples for the sadder emotions and brighter greens and yellows for the happier emotions. We stuck with the first set of colours until we were in our last stages of creating the app, but after putting into a contrast checker we found that the colours are too conflicting so we updated our colour scheme to the second set of colours shown below.

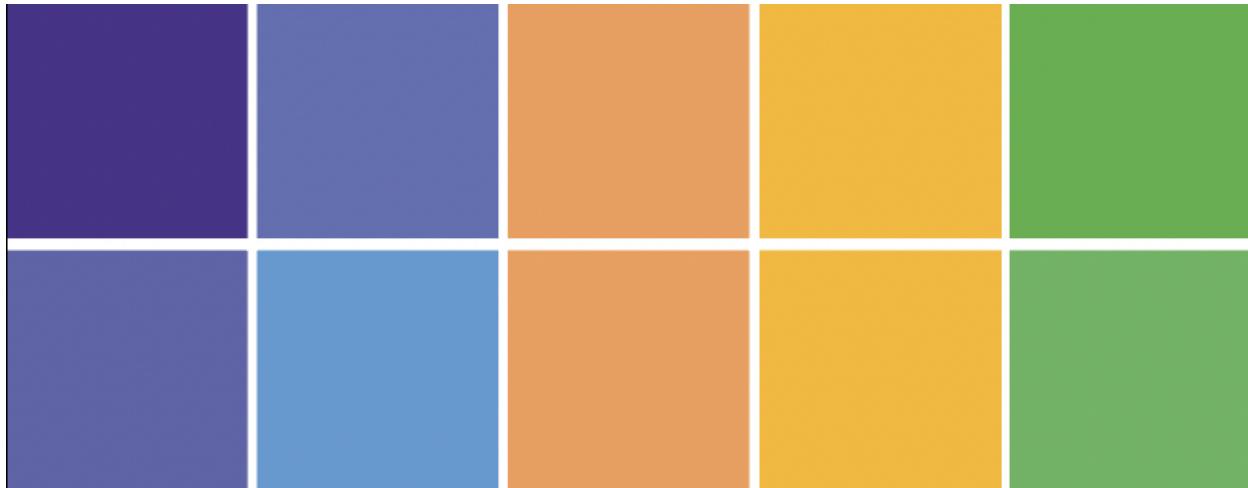


Figure 19 - Colour schemes

## **UI Asset Scale:**

As we have created the app to fit on multiple screens we have constrained the elements on the xml with parents it has scaled automatically for larger devices. With the user interface set-up like this it adapts to many screen sizes without having to create external elements for each screen size.

# Bibliography

---

## Reference list:

'Android Developers' (2023) *Android Developers*. 2023 [online]. Available from:  
<https://developer.android.com/> [Accessed 5 December 2023].

Android Knowledge (2023) Splash Screen in Android Studio using Kotlin | Latest 2023 API Method. *YouTube* [video]. Available from:  
<https://www.youtube.com/watch?v=Eu7lkrDjBq8&t=125s> [Accessed 17 November 2023].

Azhar (2020) *How can I get the current time and date in Kotlin?* [www.tutorialspoint.com](http://www.tutorialspoint.com). 20 April 2020 [online]. Available from:  
<https://www.tutorialspoint.com/how-can-i-get-the-current-time-and-date-in-kotlin> [Accessed 24 November 2023].

Chaitanyamunje (2022) *Calendar View App in Android with Kotlin* *GeeksforGeeks*. 22 May 2022 [online]. Available from:  
<https://www.geeksforgeeks.org/calendar-view-app-in-android-with-kotlin/> [Accessed 10 November 2023].

Code, R. (2022) *Kotlin - Get text input with EditText* *YouTube* [YouTube Video]. Available from:  
<https://www.youtube.com/watch?v=vIOruRkaw8c> [Accessed 10 November 2024].

Developers (2023) *Declare restricted screen support*. Available from:  
<https://developer.android.com/guide/practices/screens-distribution> [Accessed 29 December 2023].

FoxAndroid (2023) *Bottom Navigation Bar - Android Studio | Fragments | Kotlin | 2023*

[www.youtube.com](https://www.youtube.com/watch?app=desktop&v=L_6poZGNXOo). June 2023 [online]. Available from:

[https://www.youtube.com/watch?app=desktop&v=L\\_6poZGNXOo](https://www.youtube.com/watch?app=desktop&v=L_6poZGNXOo) [Accessed 24 November 2023].

FoxAndroid (2020) *Responsive App Design in Android Studio 2021 || Android Studio Tutorial || Foxandroid*. *YouTube* [video]. 29 December. Available from:

<https://www.youtube.com/watch?v=L-TTQsIUGAk> [Accessed 27 December 2023].

Halftone Digital (2020) *JSON Exporting in After Effects*. *YouTube* [video]. 06 August. Available from: <https://www.youtube.com/watch?v=SfaohlYjkqk> [Accessed 28 November 2023]

'How can I prevent my Kotlin app screen from rotating?' (2018) *Stack Overflow*. September 2018 [online]. Available from:

<https://stackoverflow.com/questions/52157672/how-can-i-prevent-my-kotlin-app-screen-from-rotating> [Accessed 30 November 2023].

'Kotlin Android Button - Javatpoint' (2021) [www.javatpoint.com](https://www.javatpoint.com/kotlin-android-button). 2021 [online]. Available from: <https://www.javatpoint.com/kotlin-android-button>.

OpenAI (2023) *ChatGPT* [chat.openai.com](https://chat.openai.com/). 2023 [online]. Available from: <https://chat.openai.com/> [Accessed 13 November 2023].

Ranjoni, A. (2018) *Android Studio Animation: Fade, Slide, Blink, Bounce, and Move YouTube* [YouTube Video]. Available from: <https://www.youtube.com/watch?v=D24UQG364PA&t=4s> [Accessed 14 December 2023].

SoundSnap (2023) *Download Sound Effects / Soundsnap Sound Library* [www.soundsnap.com](https://www.soundsnap.com). 2023 [online]. Available from: <https://www.soundsnap.com/> [Accessed 11 December 2023].

W3 Docs (no date) *How set background drawable programmatically in Android*

[www.w3docs.com](http://www.w3docs.com) [online]. Available from:

<https://www.w3docs.com/snippets/java/how-set-background-drawable-programmatically-in-android.html> [Accessed 21 November 2023].