

# Final Year Project Report

**Full Unit – Final Report**

## A Study In Human Computer Interaction

Manjyot Sandhu

A report submitted in part fulfilment of the degree of

**BSc (Hons) in Computer Science**

**Supervisor:** Gregory Gutin

Department of Computer Science  
Royal Holloway, University of London

September 26, 2013

## Declaration

This report has been prepared on the basis of my own work. Where other published and unpublished source materials have been used, these have been acknowledged.

Word Count: 18802

Student Name: Manjyot Sandhu

Date of Submission: 28/03/2018



Signature:

# Table of Contents

<b>Introduction</b>	<b>5</b>
Project Specification	5
Personal Motivation	6
Goals	6
Following Sections	7
<b>Literature Review and Background Reading</b>	<b>8</b>
Background Reading	8
Art In Interfaces	8
Nielsen's Heuristics	10
Colour Theory	11
Typography	13
Case Studies	14
Mobile Application	14
GUI	17
Google Design Guidelines	18
Literature Review	19
<b>Project Practice and Preparation</b>	<b>21</b>
Proof of concept programs	21
Project Risk Analysis	27
<b>Project Development</b>	<b>28</b>
Mobile Application	28
Textual HTA	28
Interface design	29
Final Product	38
GUI	41
HTA Diagram	41
Interface design	42
Final Product	45
Website	47
HTA Diagram	47
Interface Design	48
Final Product	52
<b>Software Engineering</b>	<b>55</b>
UML	55
Software Engineering Method	58
Developer Environment	58
Professional Issues	59
<b>Self-evaluation</b>	<b>60</b>
Overall Achievements	60

Reflection of Project Progress	62
Results Analysis	63
Future Goals	63
Conclusion	64
<b>Appendices</b>	<b>64</b>
Bibliography	64
Project Diary	65
User Manuals	67
Mobile Application	67
GUI	67
Website	67
Screenshots of working programs	68
Mobile Application	68
GUI	72
Website	74

# Introduction

## Project Specification

As follows, here is the direct aims and final deliverables of my project, as per the Royal Holloway project listing:

### A study in (HCI) human computer interaction

**Aims:** To compare various user interfaces and evaluate their design in terms of human usability

#### Final Deliverables

1. Design and implement a more advanced interface(s)
2. Complete report
3. The programs must have an object-oriented design, using modern software engineering principles.
4. The report will describe the software engineering processes involved in generating your software.
5. The report will include comparisons of interfaces with a discussion of their meanings.
6. The report will include a User Manual.

The notion of ‘design’ has always been an essential one, something that needs to be well thought out, prior to any implementation. Even before the world we live in today, where technology clouds our every move, there has always been a need for careful design to maximise functionality and efficiency. To be more specific to the current project at hand, we have a large variety of human computer interaction that we pass by, every single day. Design can be that critical of a component that it can be the difference between life and death, for example piloting a plane or using dangerous weaponry. What perhaps could be overlooked as a small malfunction in design, could have disastrous effects if misunderstood in certain situations. Furthermore, businesses heavily rely on their design of their websites and applications, in order to increase profit margins. However, with the increased need for good human computer interaction, there also comes many problematic factors with this. Some of these issues are:

- Colour theory
- Human perception
- Gender/age/cultural differences
- Special needs issues
- Speech recognition
- Cognitive issues such as memory
- Design of fonts
- Navigation
- Feedback to user
- Usability

These few issues are just a preview at the many issues that come with designing any interface.

## Personal Motivation

From a young age, having taking an interest in art, the idea of colour theory and any form of design has always been somewhat on my mind. I spent a considerable amount of my childhood drawing and painting, which led me onto a somewhat more creative choice in my GCSE and A Level options. During my art GCSE, I quickly discovered a fierce passion for design through Photoshop, which I tried to incorporate into my art, as often as I could. In addition to this, I took GCSE Graphics, wherein I learn about the depth of colour theory and design, in a more commercial sense. A combination of these GCSE's allowed me to pick up a skill set which came to peak in my art A Level. My coursework was always very photography and photoshop heavy, which led me on to website design. A lot of my free time was spent messing around with HTML and CSS, as I thoroughly enjoyed designing websites.

As mentioned previously, the aspect of 'good design' has always been on my mind. I knew from a young age that I would pursue a career in design, but in what aspect I was not sure. During the course of this degree and taking a human computer interaction module, last year, it only solidified my aspirations to have a career in this field. My love for programming, but also art and design, has always allowed me to have a different approach to how I solve a task. I have always leaned towards more visual tasks, such as the modules of 'Game Design' and 'Visual and Exploratory Analysis'. To put it simply, I have a strong passion for creating something. I enjoy the rigorous process of designing and interface and have it fully implemented, to see the end result. I have spent hours, late at night, exploring photoshop and learning about various CSS tricks. I also wanted to learn something entirely new, which is what I would get from creating a mobile application. My choice of project was one that I carried out without hesitation, as this study was of course an excellent knowledge to further my knowledge on the subject.

## Goals

In the project specification, it is specified that I must implement at least 3 different software interfaces. At the end of this implementation, I must write a report which discusses both the software and hardware interface, whilst drawing any relevant conclusions and comparisons. Using the final deliverables provided and my own goals, I present my goals for this project:

- Implement a mobile application
  - The mobile application will be a task application which will also contain a login and register page. There should be suitable navigation to access both different screens and setting pages. Throughout the application, there should be a consistent design which follows Android conventions.
- Implement a GUI
  - The GUI will be a notes application. This GUI should have a very clean design, similar to those applications used in Windows.
- Implement a website
  - The website will be a personal portfolio website, showcasing some of my skills and work
    - Having provided that I am coming to an end to my degree, I would like this website to be fully operational and suitable to share to future employers
  - The website should be suitable to run on different screen sizes
- Consider HCI issues during these implementations

- During the implementations of the interfaces, I should consider the HCI issues mentioned above and work to solve as many of them as I possibly can
- Evaluate the interfaces, in terms of HCI principles
  - Complete an evaluation or results analysis on the interfaces and discuss the differences
- Delve into a professional issue, to increase my prospects as an IT professional

## Following Sections

Follows, is an explanation of my project report sections that I have decided to use.

### Theory

In this section I delved into all of my research that I carried out, in preparation for the design and implementation of my interfaces. My research sections are:

- Art In Interfaces - A look at how artistic ideas can be applied to my interface designs
- Nielsen's Heuristics - Heuristics which I will need to consider, when designing my interfaces
- Case Studies - A detailed look at other interfaces, of a similar type to that which I will be implementing
- Other minor research sections
  - Colour Theory
  - Typography
  - Google Design Guidelines

### Project Practice and Preparation

In this section, I aimed to focus on any preparation I would need to do, before any actual design or implementation of interfaces. This included a risk analysis, to assess any risks I may come across and how I may work to prevent them. In addition to this, a project plan which would help me to stay on track of my tasks and work against a timeline. On the practice side of things, I delved into my experience with my proof of concept programs and any issues that arose during work on these. There is also a section about how the proof of concept programs helped me to reassess my project and work out what would be feasible for me to complete. Learning how to use the software and any tutorials I completed, prior to development on the code, is also discussed here.

### Project Development

This section concerns the bulk of my development, in terms of my design and implementation. My preliminary designs from my proof of concept programs are assessed, in terms of Nielsen's Heuristics and any other portions of my research. Final designs are then presented and explained, in depth. Discussions about my coding development and my thought process behind some of the implementation is included. Any problems that arose or challenging pieces of code are also presented here.

### Software Engineering

Here I include basic UML diagram of some of my interfaces. As my project is HCI and theory heavy, there are not as many complex classes as more coding projects would have. However, I presented diagrams of many of my classes and my motivations behind my choices. Furthermore, I presented a description of the developer environments that I used and any plugins which provided to be useful to me. As a final section to software engineering, I discussed my professional issues, in depth.

### Self-evaluation

In this section, I complete a reflection of my project progress, in terms of goals that I managed to achieve and those which I perhaps did not. Furthermore, I analysed my results, as this project is focused on comparing the interfaces which I implemented. Future goals are also discussed, if I had more time to work on my interfaces.

### Appendices

Included here is

- Bibliography
- Project Diary
- Project Plan
- User Manuals

## Literature Review and Background Reading

### Background Reading

#### Art In Interfaces

One of the reasons why my interest is so apparent in interface design, is the idea that it is quite similar to art. The different elements and shapes, small details, such as a simple highlight on a painting, will make all the different to the human eye. When I was reading a HCI textbook [1], the perfect quote came up wherein it was said that HCI is a “marriage of art and science” and that the art of interface design is a “scientific method accelerated by artistic insight”. This further prompted me to take notice of all the small details in various interfaces and how even one small detail can have such an enormous impact. Many people have tried to summarise basic rules of interfaces, but Cooper [2] managed to summarise it well. He identified some key features of elements which should always be considered, in any design:

- Shape
- Size
- Hue
- Orientation
- Texture
- Position

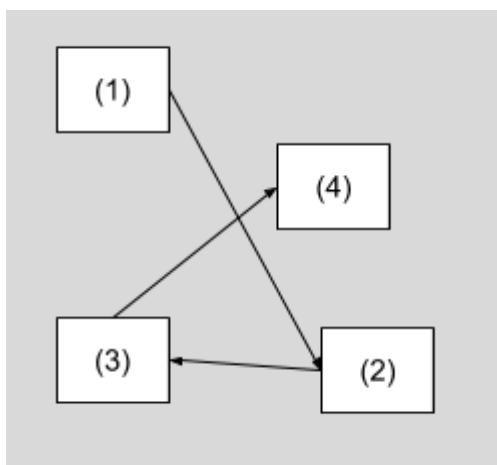
The idea is that these building blocks all work together to create a visual system and meaning, through their various characteristics. From a first glance, the basic silhouette of a shape can help a user to quickly identify what it is. For example, a house symbol will be appropriate for a ‘Homepage’ and will not require much thought from the user to draw this conclusion. The use of a triangle shape for a button may not be entirely suitable, as most real life situation have round or square buttons. This can lead to the user having to observe the interface more carefully and thus decreasing efficiency. However, it requires a considerable amount of attention from the user to distinguish between various shapes, than it does for other characteristics, so it is not best to rely solely on this idea.

As years have passed with using interfaces, certain ‘ideals’ have been developed, such as a line shape can be interpreted as an area for user input. It is also vital in any design to adhere to these basic standards. The size of an element can help to help the user interpret how important an element

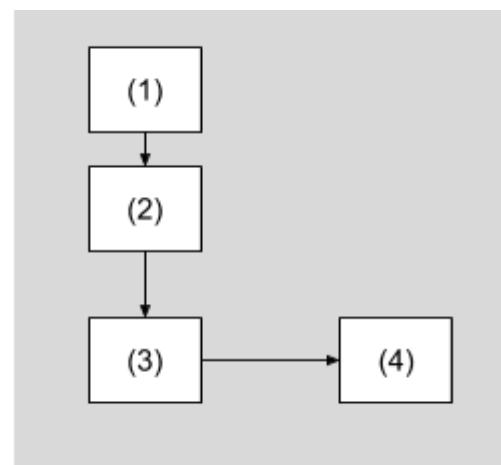
is, just from a quick glance. When a user is observing various elements, they will automatically compare the sizes and likelihood is their eye will be drawn to the largest element first. In a body of text, the notions of headings and subtitles are used to represent the importance and a certain type of hierarchy of information. In many modern interfaces, on a login page, the ‘Register’ button is usually the largest, as a means to encourage more users to click on the button and therefore register with the enterprise.

Colour can have a massive impact on an interface, as it is very easily distinguishable for the human eye, even for children. For this reason, it is so often utilised in learning environments. Various shades of colours and hues can be used to, as Cooper [2] referred to it as, a ‘visual system’, to help the user distinguish between similarities and differences. Certain colours have certain connotations attached to them, such as red has the meaning of negative, which is why it is often used to represent errors or cancel buttons. On the other hand, we can argue that we have this idea in our head because of day to day idea, such as stop signs or traffic lights. In other parts of the world, they may not have these ideals and therefore colour is not a good idea to rely on, to convey a message. In addition to this, colour blindness can mean that certain users are not able to get the full use out of a system. The orientation of an interface can help to provide the idea of a flow or direction for the user. Providing that a user is completing a form in a mobile phone application, in a portrait orientation, the user will naturally complete the form going from the top to bottom. Using icons which are pointing in a left or right direction can also help to direct the user, such as a ‘Next’ button. Even small aspects of a system, such as a scroll bar can help give the user a sense of direction.

Another, slightly neglected characteristic, is texture. The texture of an element, such as a drop shadow or bevel effect can have a subconscious effect on a user. In real life, buttons are usually protruding and therefore in a technological interface, the same ideals will apply subconsciously. If an element appears to have a slight shadow to it, the user may assume it is a button of some kind or that it is draggable. Last but not least, another consideration is the positioning of an element. Similar to the idea of size, wherein the user will make comparisons, similar comparisons will be made between the positions of elements on a page. Grouping together components will create the idea of a relationship between them and separate them from the rest of the interface. This is why grid systems are commonly used in interfaces, as it is an easy way to split up the view. In any system, the flow of the elements is vital and can result in an easy to use and quick interface.



*Example of bad flow*



*Example of good flow*

After analysing a lot of various characteristics of elements and their possible effects on an interface it can be observed that it is imperative that not one aspect is neglected. Too much emphasis on one characteristic can cause a failure for the system, such as colour blindness can render colour useless.

However, each characteristic can play such an important role in an interface and convey so many different messages. We can see that it is not about one artistic feature of an element, it is about finding a way to consider and implement all of them, in a way that they complement each other. Only when all of these artistic notions are considered, can a interface truly shine in a way which is both aesthetically pleasing to the user and functional. In a lot of ways, similar to painting a picture, everything must link and complement each other.

## Nielsen's Heuristics

A heuristic evaluation is a way to assess an interface, in terms of usability. As our project is focused on the development and comparison of three interfaces, researching the different types of heuristics [3] [4] would provide some useful insight into how to complete the design. A number of people have developed their own interpretations of usability heuristics, but perhaps one of the most used standards are Nielsen's.

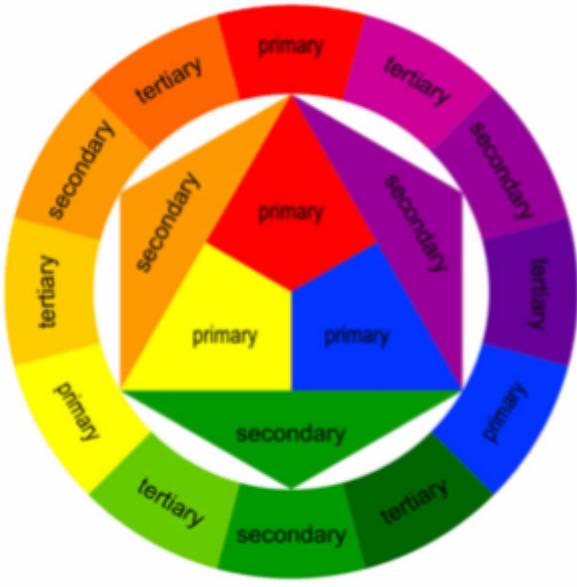
- Visibility of system status
  - The user should be aware of the current state of the system at all times. This will help the user to have an easier time navigating and understanding the system. Furthermore, if for example there was a loading screen and no loading icon, the user may assume the program has crashed. It is vital that the user feels involved in the process of utilising the interface. This feedback done in an appropriate manner, such as graphical or text displays of system status.
  - Examples of use:
    - Progress bar whilst completing a survey
    - Loading animations after user has completed a process
- Match between system and real world
  - The system should follow real-life standards, to make it easier for the user to relate to, and therefore understand the system. Utilising recognisable icons, which also have a place in our day-to-day lives, will mean that the user will already have some kind of idea of how to use the interface. In addition to this, a logical flow of components should occupy the interface, so that using the interface comes somewhat naturally, to the user. When one designs an interface, we must always assume that the user will be someone with no experience of IT systems and therefore will not be familiar with any complex terms or actions. As a result of this, simple understood language should be used, so that any demographic can make use of the interface.
  - Examples of use:
    - A home icon for the home page
    - Proper order of menu items: Home → New → Edit
- User control and freedom
  - When a user is using the system, it is important that they have freedom with their actions and that they are in control of what they are doing. If there are any instances of pop ups or dialogs, the user should be able to perform an 'emergency exit' and leave any unwanted states.
  - Examples
    - Red cross button in the corner of the screen
- Consistency and standards
  - Throughout the interface, the majority of the layout should be kept very similar and follow basic rules, so that it is easy for the user to understand. Furthermore,

phrases and icons should not mean different things, otherwise it will confuse the user.

- Error prevention
  - In a situation where there is room for error, particularly from the user, it is useful to implement some form of error prevention. The coding itself, behind the program, should be done well to help prevent any errors possible
  - Examples of use:
    - In a register form, checking to see if all of the fields contain input
- Recognition rather than recall
  - Whilst using the system, the user may have to go through many options and various screens, which they should not have to recall. Any choices which are made, which will impact a choice later on, should be displayed on the screen for the user to recall from.
- Flexibility and efficiency of use
  - Although many of the heuristics are focused on making the system easy to understand, perhaps for the most basic user, it is also important to cater to an advanced user.
  - Examples of use:
    - Keyboard shortcuts for frequently used actions
- Aesthetic and minimalist design
  - In general, the entire interface should follow a very simple design to ensure there is no overuse of elements, to avoid any confusion. Any unnecessary elements on the interface will take attention away from the important components, which need to be paid attention to.
- Help users recognise, diagnose and recover from errors
  - If an error is to occur, the error should be displayed on the screen, in a manner which is able to be understood by someone who is new to the system. The error messages should not be in any complex form, such as hard code. Once the error has been displayed, there should be solutions on how it can be recovered from, for which the user can carry out
  - Examples of use:
    - When logging in, if the username field is left empty an error saying ‘Please enter username’ can be displayed on the screen
- Help and documentation
  - If a user is new to the system, or simply requires assistance, it is important that there is relevant help and documentation to aid them in any problems.
  - Examples of use:
    - User guide/Help manual
    - Search for help feature

## Colour Theory

Although in this report we are referring to colour within an interface, one could argue that the basic rules of colour theory in everyday life can still apply. As Toledo [5] expresses, certain colours can have a huge impact on the user, as they can convey various messages such as emotions and thoughts. This can be felt even through an electronic device and in this case a website, GUI or mobile application. Below is an image of the colour wheel and corresponding colour schemes:



- Monochromatic → Use same colour, but different hues
- Analogous → Use colours that are next to each other in colour wheel
- Complimentary → Colours that are opposite on the colour wheel
- Triadic → Three colours, evenly spaced on colour wheel

Continuing on, Toledo [5] suggests that specific colours can convey various emotions to the user. For example, using a bold red colour can symbolise power and passion, which is perhaps suitable for a campaign website. Orange is a bright and vibrant colour, which can be used to gain immediate attention. The colour can also be seen as somewhat energetic and much less ‘aggressive’ than red. Yellow has similar characteristics to orange, wherein it can be used to demand attention. However, using yellow in a design should be done sparingly as it can be overwhelming, when used in large quantities, or hard to read text over it. Shades of greens and blues can be seen as quite calming colours, which are suitable for most situations.

On the other hand, Rocheleau [6] argues that emotions through colour is highly subjective and personal, so should not be used as a sole method to convey a specific message to the audience. It can be argued that a more important characteristic of colour is the value, which is the measure of dark to light on the spectrum. This value is what helps to define and help users compare elements to each other, which in turn creates a sense of hierarchy and order on the interface. The saturation of colour is the level of vibrancy of a colour. By a general rule, a colour being used for a large portion of the interface should not too vibrant, as it can be too overwhelming for the user. In Rocheleau’s words, a more “washed” out colour is more effective. When in confusion about a colour system, it may be useful to resort to a black and white scheme, to ensure readability of the theme itself. This can also be a useful way to evaluate if the interface has suitable levels of contrast.

In addition to all of these complex considerations one must take when deciding on colours for an interface, it is also suggested by some designers that pure black should not be used in large quantities. Ian Storm Taylor [7] presented the idea in an article wherein he came to the realisation that in real life, nothing is truly black, it is just a very dark colour. One example of this are shadows, which are not truly black. Seen in the artist, Wayne Thiebaud’s [8] work, who is well renown for his shadow work, we see that in his paintings the shadows are never truly black. Instead, there are tints of various other colours in the shadows, such as blues. One can question if the art matters if it depicts real life, do our interfaces also? In order for interfaces to feel real to a user and relate, we must adopt these day-to-day ideals and attempt to apply them, as best we can. As a result, we should not use pure black in our interfaces, but very dark grey colours, as a replacement.

Furthermore, black is a very overpowering colour and can take away attention from other elements. It is also important to note, that shades of grey can appear dull and make an interface feel lifeless.

To combat this, the saturation can be increased with shades of blue, giving the grey shades a slight lift to them. An example of an interface which does this is Facebook, wherein they add blue tints to many of their grey areas, to add saturation to the overall interface. From this research on colour theory, I gained more of a deeper understanding as to why certain colours are not suitable. My eye has been a lot more open when it comes to noticing designs in real life, not just technological ones, but even signs and architectural visual systems. The most effective colour schemes to me seem to be fairly monochromatic, with a pop of colour. When designing my interfaces, I will most likely make use of greys/whites for the large bodies and small ‘dull’ accent colours.

## Typography

Much like considering various characteristics of components, such as shape, colour and position, there also are many things to be considered in typography.

Tubik Studio [9] wrote an article on the main things to consider when deciding on typography, below are a few of the main points I wanted to expand on.

1. Font size
  - As mentioned previously, in the art of design section, size can represent a hierarchy in the interface. Therefore, it is important to be very careful when selecting specific font sizes.
  - Text should not be too large, otherwise it will overwhelm the user and take attention away from other components.
  - On the other hand, text should not be too small otherwise it will be difficult to read, especially on smaller devices.
2. Line spacing
  - Line spacing can have a huge impact on the user’s reading speed, for large bodies of text. A good balance must be found wherein the lines are spread enough apart for the human eye, but not too unconnected from the main text.
3. Line length
  - We must also consider the fact that a website may not only be viewed on an internet browser, but also a mobile application. In addition to this, someone could be viewing it on a tablet device or even a small laptop. It is important that any long lines of text on the screen do not have too many characters, otherwise it will be too difficult for other users to read.
  - Tubik Studios [9] recommended that 30-40 characters are kept on a line, if it is a mobile application.
4. White space
  - If we want our interface to have a ‘minimalistic’ feel to it, there must be some use of whitespace between characters. However, we must take care to not include too much white space otherwise it can decrease readability.
5. Responsiveness
  - Just as we would design the layout of a website to be as responsive as possible, the same stance should be taken towards any interface. If the interface is able to be resized or have a chance of being on a largely differently sized device, the fonts should scale to the new requirements.

Another point to note is that fonts should be kept very simple, for a computer screen. In real life, books are more often than not, in Serif font. A Serif font is easier to read, for long lines of text, whereas Sans-Serif is more suitable for a computer screen. It is also important to not only note the font overall, but the individual characters. Chris Bowler [10] highlighted how many fonts have characters which are not at all distinguishable from each other. His example he provided was

between the font Lato and Clean Sans. As we can see, the capital I's and lowercase l's look very similar, which can lead to confusion and decrease readability.

### III-gotten type

Lato

### III-gotten type

Clear Sans

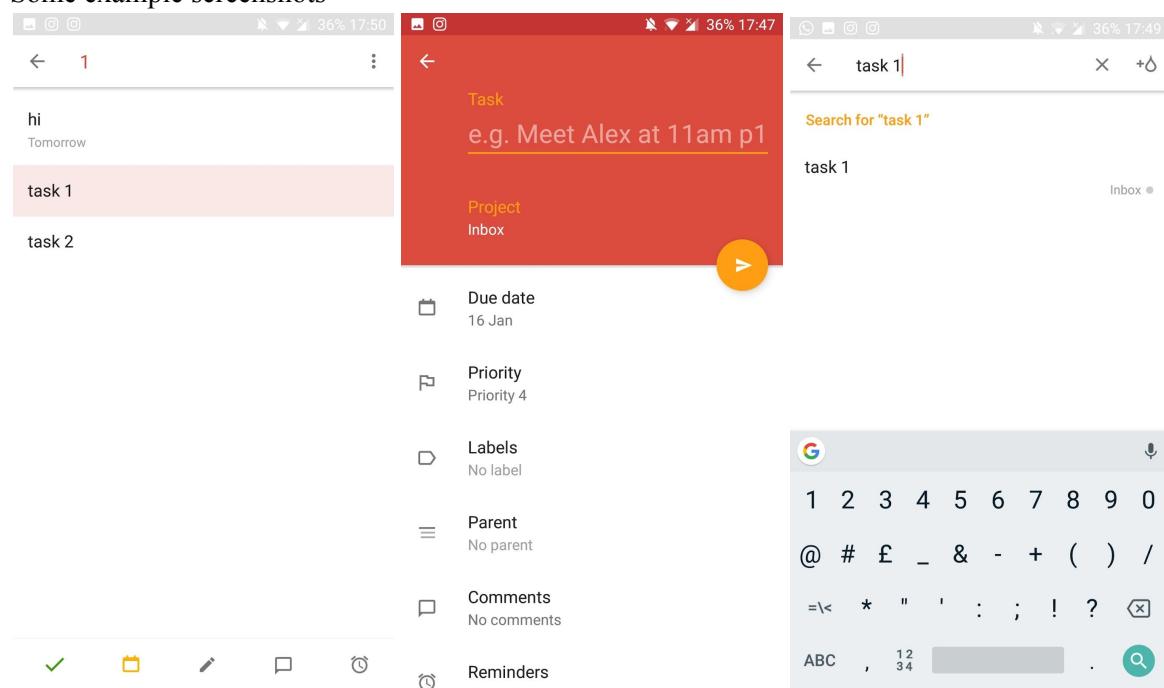
From this research section, I can conclude that I must be very weary when choosing a font. I aim to choose fonts which are very simple, but legible to read. Furthermore, I will ensure to make use of whitespace, but not too much that the word looks disjoint. I will carry out a lot of testing with different devices and window sizes, to test if my fonts are responsive enough.

## Case Studies

### Mobile Application

#### Android Application Case Study 1: Todoist

Some example screenshots



Nielsen's Heuristics applied to this mobile application design

- Visibility of system status

- On the main task page, if there are no tasks created then a symbol will appear, with the words 'No tasks' and 'Add a task'. This will draw the users attention and encourage them to add tasks, while making them aware why the screen is empty. Instead of having a blank, empty space this application makes use of all the space that it can
- For most of the screens, there is a title at the top of the application. This helps the user to identify which screen they are on and therefore have a better idea of the action they are



No tasks.  
Add a task.

carrying out. On some screens, which are more self explanatory, such as creating a new task, there is not a title.

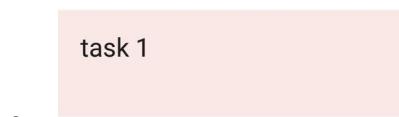


- On the search screen, there is text under the search bar which states the search query that the user has typed in. This notifies the user of exactly what search query is being carried out.



- When a task is selected, it is highlighted in a different colour

**Today** Tue 16 Jan



- Match between system and real world
  - Use of familiar icons



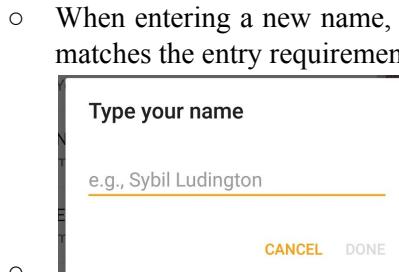
- Logical order of options, such as support and about being at the bottom of the settings page.

- User control and freedom
  - Undo supported, after a task completion

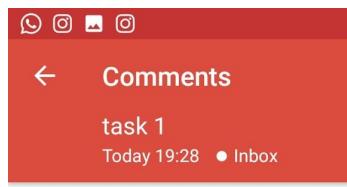


- On the search bar, the user can click the cross to erase the search, instead of having to delete and retype the entire search.
- Back button on most screens, in clear space on screen (top left usually)

- Consistency and standards
  - Icons have the same meaning throughout the app, to avoid confusion
- Error prevention
  - When entering a new name, the 'DONE' button is greyed out, until the user input matches the entry requirements



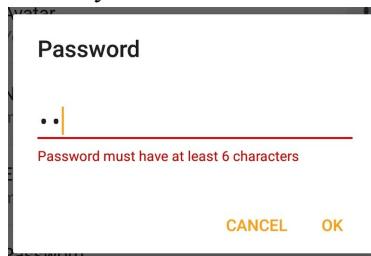
- Recognition rather than recall
  - In some of the sections of creating a new task, where a new screen is navigated to, the task details are displayed at the top, under the title. This means that the user will always be aware of which task they are editing or creating, saving them from having to navigate back a page and losing progress



- Flexibility and efficiency of user
  - When a task is clicked on, these options appear at the bottom. For a more experienced user, who is familiar with the app, they can quickly edit details using these buttons



- Aesthetic and minimalist design
  - Throughout the design, it is very clean
- Help users recognise, diagnose and recover from errors
  - When entering a password which does not satisfy the requirements, the colour of the entry line turns red and an error message appears, in simple English.



- Help and documentation
  - In the main menu, there is a support section which the user can click on to get extensive documentation

### Android Application Case Study 2: Android Studio

In order to fully be able to do an effective design for my mobile application, I needed to have a greater understanding of the various features that were available from Android Studio. To do this I installed Android Studio and had a look for myself for the different types of activities, layouts and components which I thought it would be useful to look into further.

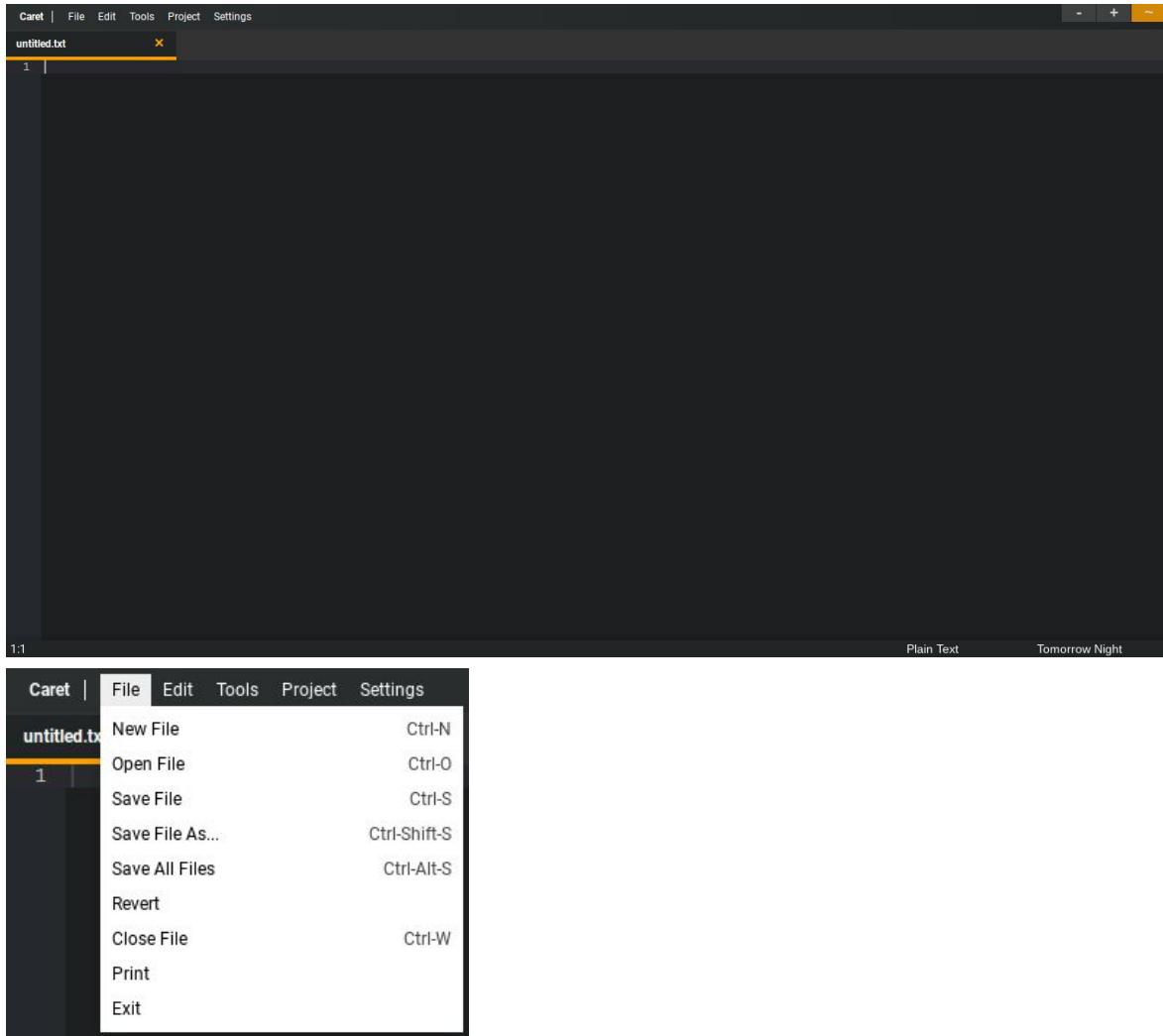
- Layouts
  - Linear Layout
    - Aligns children in single direction (vertically or horizontally)
    - Children of this layout are stacked after one another
  - Relative Layout
    - Displays children in relative positions to each other
    - Position of each child can be positioned relative to sibling elements
    - Will be useful if I want a component to stay in a specific part of the screen
  - Recycler View
  - List View
- Activities
  - Settings
    - Provides settings activity which generates list of preferences
    - Provides components: ringtone preference, checkbox preference, list preference, etc
  - Log in
  - Preferences activity
    - Provides shared preferences that run across the entire app

- Comes with premade multi list, checkbox, dialog preferences
- Navigation drawer

## GUI

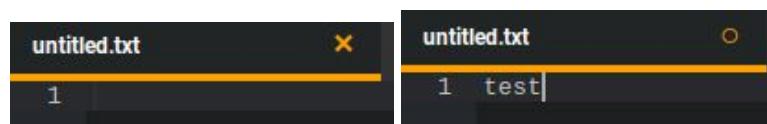
### Case Study 1: Caret

#### Example Screenshots



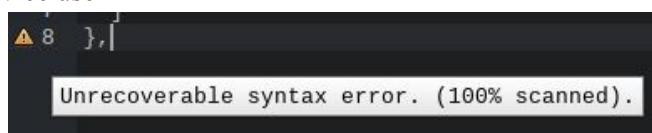
#### Nielsen's Heuristics in this application

- Visibility of system status
  - The title of the document is displayed at the top of the tab
  - When the document is in a state where there are no unsaved changes, the icon in the tab bar will be a cross. When the document has unsaved changes, the icon will change to a circle
- In the bottom left, there are two numbers which display the line and character that the users cursor is currently on
- The type of text and theme is presented in the bottom right of the screen
- Match between system and real world
  - Icons are used which represent real life ideas, such as a cross to close something.



- The only confusion parts of the design is that a ‘plus’ and ‘minus’ symbol is used for minimising and maximising a window, which can be confusing.
- Logical order of menus: File → Edit → Tools → Project → Settings
- User control and freedom
  - Undo and redo are supported, through shortcuts and menu options
  - There is an option called ‘Emergency reset’ under the Settings menu, which can be used to reset all of the settings
- Consistency and standards
  - All of the options are very simple and easy to understand
- Error prevention
  - If a document is closed, without being saved, a confirmation box will appear. The user will then be able to save, discard or cancel the operation.



- Recognition rather than recall
  - When an option is selected, such as search for, a popup dialog box will appear. The name of the option which is selected will appear at the top of the box, so that the user can remember what action they executed
- 
 A screenshot of a dark-themed search dialog box. The title bar says 'Go To:'. Below the title, there is a text input field with a vertical cursor line in it. The entire dialog is set against a dark background.
- Flexibility and efficiency of user
  - This application supports numerous shortcuts, which can also be customised
    - These shortcuts are presented, next to each of the relevant options in the menus
- Aesthetic and minimalist design
  - Overall, the design is very simplistic and has no unnecessary components.
- Help users recognise, diagnose and recover from errors
  - An icon appears next to the line with an error, with a popup ‘error’ message, if hovered over. The error is in plain english, but could be more in ‘simple’ english, for a novice user
- 
 A screenshot of a dark-themed error message box. It shows a small warning icon (triangle with exclamation mark) followed by the number '8'. Below that, there is a line of code with a cursor. A tooltip-like message box is overlaid on the right side, containing the text 'Unrecoverable syntax error. (100% scanned)'.

- Help and documentation
  - There is a ‘Help’ option, under the settings menu

## Google Design Guidelines

When thinking about the design of my Android Studio application, I browsed a few of the ‘default’ applications on my phone, such as image gallery. From looking at these applications I came to the realisation that they all follow certain conventions such as font sizes, button choices and spacing. It was important to me that when designing my application, I also followed these conventions, therefore I researched some key ideas I needed to adopt into my designs.

- Typography
  - Standard typeface is Roboto
  - Font sizes

- App bar = 20sp
- Buttons = 14sp
- Subheadings = 16sp
- Body = 14sp
- Writing
  - Language
    - When addressing users, should be in second person for most situations and first person when emphasizing user's ownership of content or actions
      - AVOID mixing first and second person in same sentence
    - Avoid "we"
    - Begin with the objective in a sentence
      - 'To remove a photo from this album, drag it to the trash' instead of 'Drag a photo to the trash to remove it from this album'
    - Present information positively ('Use 24 characters' instead of 'You must use less than 25 characters')

## Literature Review

*Alan Dix, Janet Finlay, Gregory D. Abowd, and Russell Beale, Human-Computer Interaction (3rd Edition)*

This book was the first source I studied, for this project, and perhaps the one that inspired me the most. I discovered the important quote: "HCI is a marriage of art and science". The book goes into great depth of many aspects of design, which would be too many to list. The book basically covers the entirety of human computer interaction and is a great source point for any one to learn more about design.

*Cooper, About Face 3: The Essentials of Interaction Design, Third Edition*

To further expand on the idea of an artistic overview of design, this book helped me to further understand many other various aspects of design. In particular, it made me focus more on honing on specific elements and the characteristics they possess. When completing my designs, this helped me to have individual elements, which as a whole came together to produce a wholesome design.

*HCI Lecture Slides 7, Royal Holloway University of London*

As a summary to Nielsen's Heuristics, these lecture slides came in useful to provide me with a list of all the heuristics. In particular, the examples also were very helpful and inspired me to think of more ways in which they could apply to my designs. On the other hand, the slides were not entirely detailed and meant that I had to go in search for another source which would expand my knowledge of heuristics in more depth.

[https://en.wikipedia.org/wiki/Heuristic\\_evaluation](https://en.wikipedia.org/wiki/Heuristic_evaluation)

The wikipedia page on Heuristic evaluation helped me to understand the formal definitions and notion of heuristics, whilst not being too specific and interaction, such as the previous source.

*Rob Toledo Article: https://usabilitygeek.com/colour-theory-introduction/*

Through this source, I learn the importance of colour theory and how different colours can represent different emotions to various users. I utilised the idea that bold colours, such as red, showed power and passion. From this, I decided to make my website have a red theme wherein it would capture the attentions of any employers, which was its intended purpose. I was not entirely convinced about this idea that colours could convey these emotions, for a larger interface. For smaller elements, such as a cancel button, I can see the logic. However, I did not incorporate too

many of this ideas into my designs. Apart from this, it did not have that much other useful information about colour theory.

*Jake Rocheleau Article: <http://www.vandelaydesign.com/ultimate-guide-color-theory-designers/>*  
In Jake Rocheleau's article I was able to gain a deeper understanding in how to pick an effective colour scheme for my interfaces. His ideas about using colour very careful is something I was sure to consider when designing my interfaces, such as my mobile application and website.

*Ian Storm Taylor Article, <http://ianstormtaylor.com/design-tip-never-use-black/>*  
Ian Storm presented some very interesting ideas about how black should not be used in design, which is something that had never occurred to me before. I made sure to incorporate this into all of my designs and never use 'pure' black.

*Wayne Thiebaud Paintings*  
This artist's work further reinforced my ideas about how I should not use black in my designs, but did not contribute to any actual design decisions.

*Tubik Article: <https://uxplanet.org/mobile-typography-8-steps-toward-powerful-ui-deaf205274c5>*  
One aspect of this article which helped me in my designs was the ideas about font sizes and how it could represent hierarchies. In my preliminary designs and proof of concept programs, I noticed some pressing problems with my font size choices, particularly in my mobile application. The font sizes for the main body was almost the same size as many of the titles, which didn't support the notion of any form of hierarchy. Furthermore, in my website the body text was often too large and clashed with some of the titles. I made sure to make these changes in my final designs and programs. In addition to this, line spacing is something I had not thought about fully for my website. One of the main goals and focuses for my website was to provide a website that was fully responsive, so line length rules that Tubik suggested of 30-40 characters was implemented into my designs. Overall, this source was able to touch up my knowledge, whilst providing some minor ideas which would improve my designs.

*Chris Bowler Article, <https://www.invisionapp.com/blog/4-tips-on-typography-in-ui-design/>*  
This article, whilst not the most useful, was able to introduce me to one of the fonts that I would go onto use for some of my GUI and website, which was Lato.

*Android Studio Tutorials, <https://developer.android.com/training/index.html>*  
The official Android Studio tutorial website helped me with the setup of much of my mobile application and explanation of many components. Although the information provided was very thorough and extensive, I found that at times the tutorials were a bit difficult to understand and sometimes aimed at a programmer who has a higher level of knowledge than I did.

*Tutorials Point*  
I used Tutorials Point to help me with various syntax, for all of my interfaces. Tutorials point came in useful for any easy to access source, which would simply tell me explanations for various syntaxes. It was not helpful for any in depth code explanations, but was able to give me the foundations of knowledge for which I could build on with other sources.

# Project Practice and Preparation

## Proof of concept programs

The mobile application was coded in Android Studio's, a software and set of syntax and libraries that I was completely unfamiliar with. I ran into a number of problems, at first, because of my unfamiliarity of Android Studio's. However, after conducting more research and gaining a deeper understanding into the use of various layouts, I was able to design a simple screen with ease. For my proof of concept program I created a basic application which loads up a login screen. In this screen, the user can enter their relevant details and it will log you into the application. On the other hand, the user can also register for the application, which can be accessed by clicking on the 'Signup' link. Upon clicking this, a register screen will appear, which has all the necessary input fields.

The idea behind this proof of concept program was to gain an understanding into Android Studio's and how to use the basics. I am very happy with the progress of that, for this term, as I feel like I ran into many problems at the beginning of the development. However, I can now confidently say that I can create an entire basic layout, in a short amount of time. When I move onto my final mobile application, I am sure that I will be much more confident and will be able to work at a faster pace.

### Mobile Application

Below are my initial designs and assessments of them

The image displays two screens of a mobile application interface.

**Top Screen: Login/Register**

- Fields:
  - Username
  - Password
  - Confirm Password
  - E-mail
- Buttons:
  - Log in
  - Sign up
  - Register

**Bottom Screen: Task List**

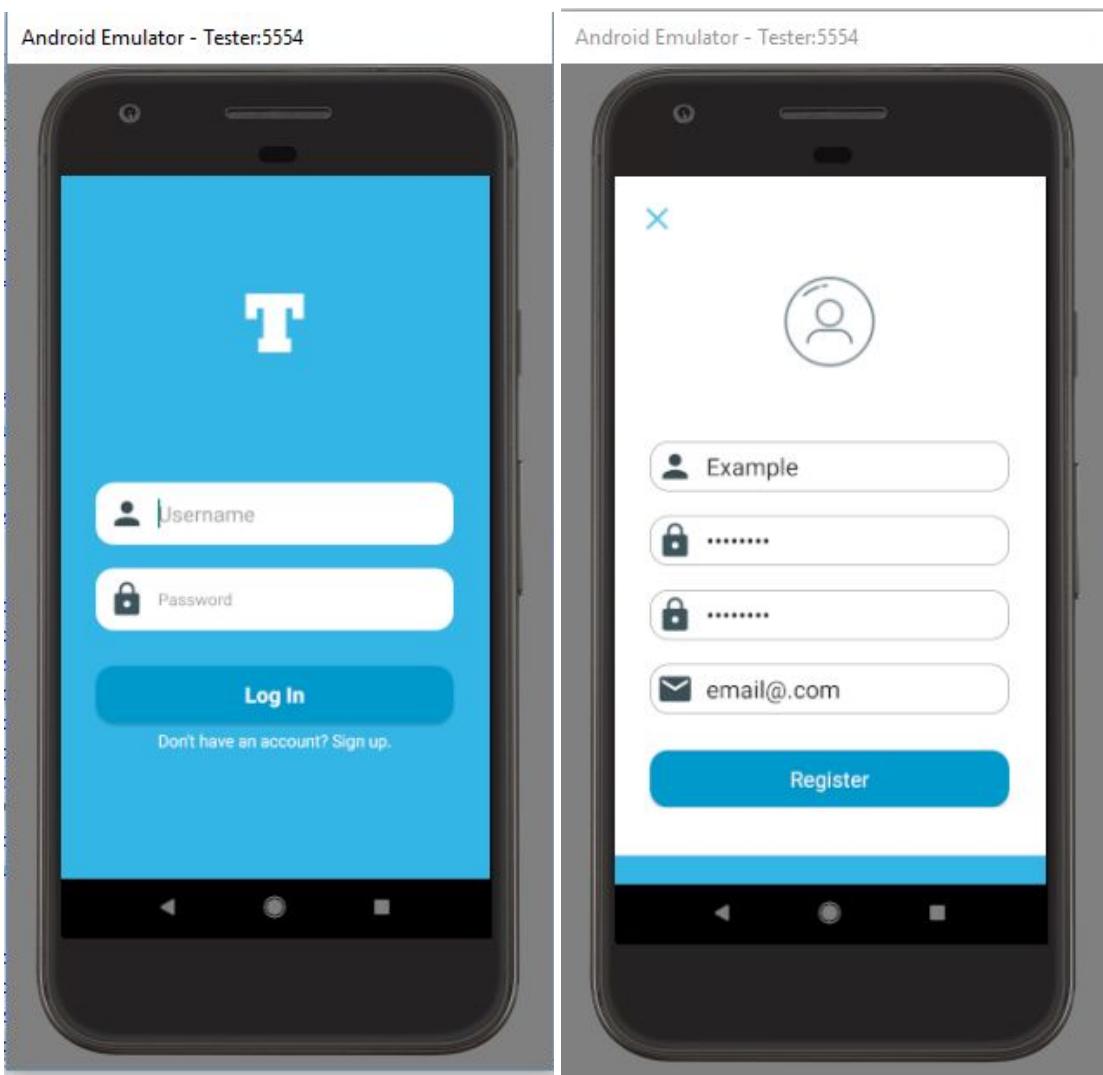
Task List categories:

- Incomplete
  - Task 1 (Due: 4 NOVEMBER)
  - Task 2 (Due: 15 NOVEMBER)
- Complete
  - Task 3 (Due: 5 OCTOBER)
  - Task 4 (Due: 9 OCTOBER)

**New Task Creation Screen**

Fields and controls:

- Task Name: Task 3
- Due date: 5 November
- Priority: (Three colored circles: green, yellow, red with checkmark)



After having reviewed Android Studio examples and a case study, I can now see that there are certain aspects of design missing from my original mobile application designs. I need to implement the following changes:

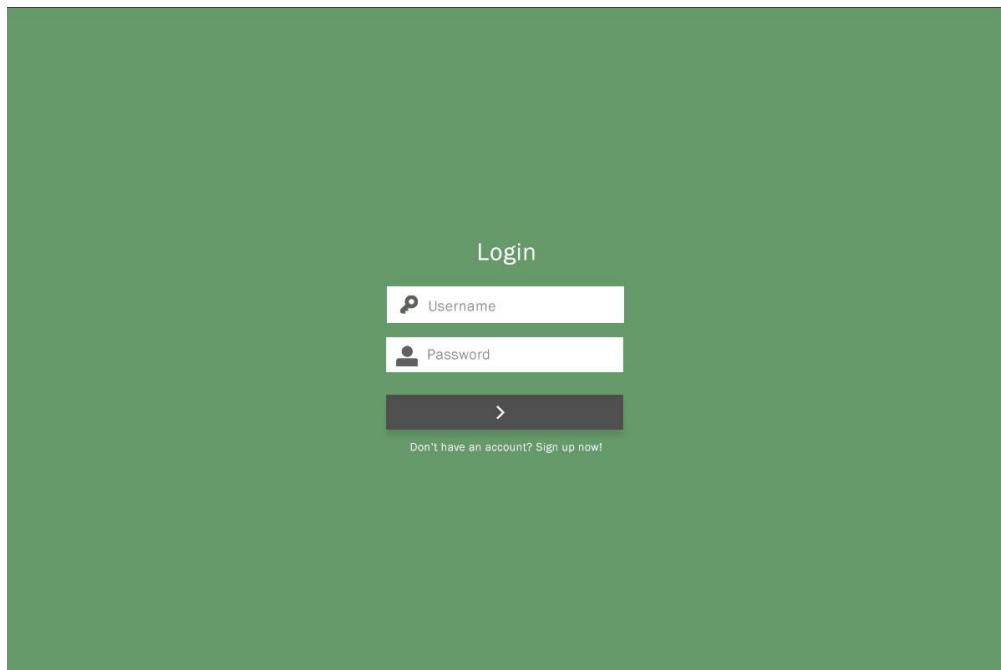
- Move toolbar titles towards left, not middle
  - Elements need to be grouped together more, to present a more aesthetically pleasing design
  - A similar alignment (left alignment), should be adopted throughout the entire application
- Make all designs more ‘standard’
  - In each design, I need to make sure that they all follow the same conventions, such as icon and font sizes. I should also use icons from a similar pack, such as the Android Studio asset library, to ensure that they all match each other in terms of thickness and general appearance.
- Remove the ‘completed tasks’ section from the main task screen
  - This is unnecessary clutter and can instead be put under a menu, under the option ‘Show completed tasks’
- Display of system status, if there are no tasks
  - Have text on screen, if there are no tasks, which will encourage the user to create a new task
- Make changes to the icons

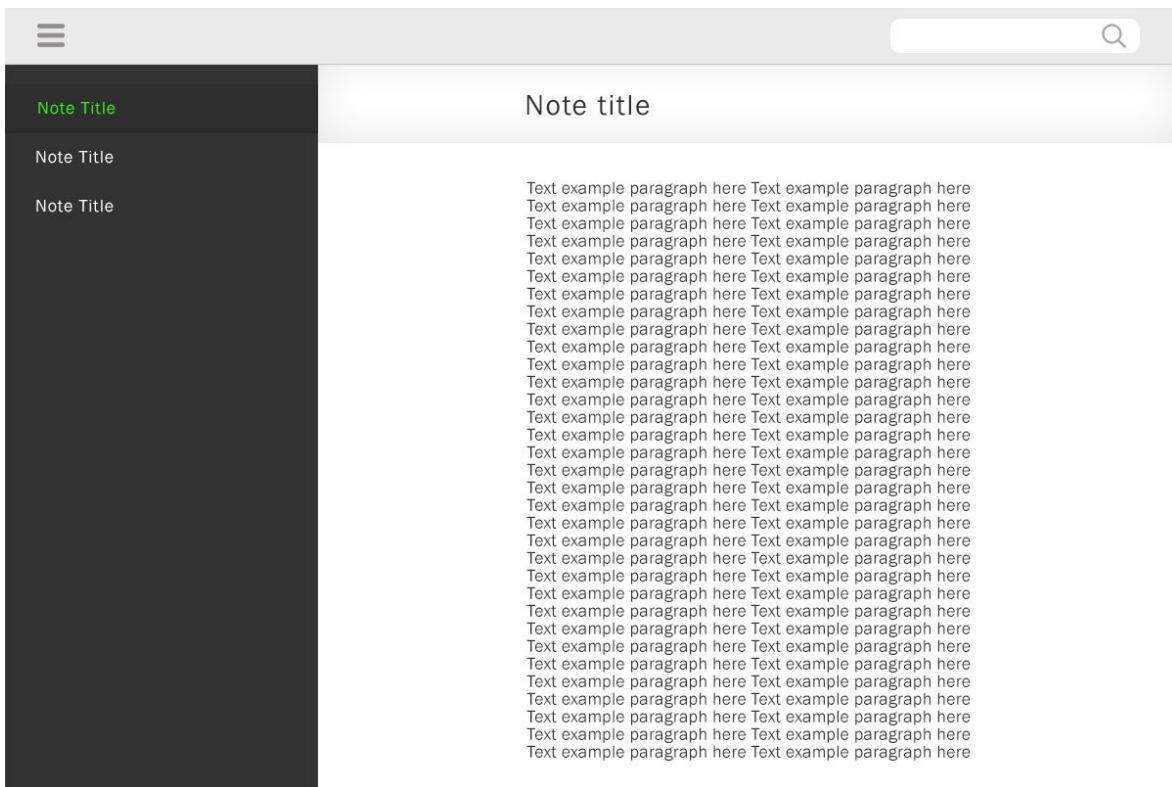
- Change the cog icon for ‘settings’ to three dots, which is more standard for Android applications
- Change icon for ‘task’ done to a plain checkbox
- Change cross icons to back arrow icons
- Ensure that the tick icons are not reused
  - In the current designs, the tick icon is used for priority, checking if a task if done and also confirming a new task. This can be confusing to assign multiple purposes to the same icon.

## GUI

The GUI Program was programmed using Eclipse and JavaFX, using WindowBuilder. From the completion of this program I wanted to mainly experiment using JavaFX, as I was not familiar with it before this project. I wanted to gain a basic understanding of the various elements and how to combine java with it.

For my proof of concept program, I focused on implementing a design with the login and register screen. The main idea was that I wanted to see if I could make the GUI look alike to my photoshop designs and if not, see how I could remedy this. I implemented a certain part of the notes function of the GUI, but I did not focus much on the design. Having being unfamiliar with the syntax and many of the features, I decided that I wanted to get a JList working with the notes and fully understand that. At first, I ran into many problems and struggled with successfully listing the JList, with the corresponding notes. However, after much research into the topic, I was able to create an interface which displays the correct note, when it is selected from the list. However, as stated, the design of the main part of the interface is not complete.



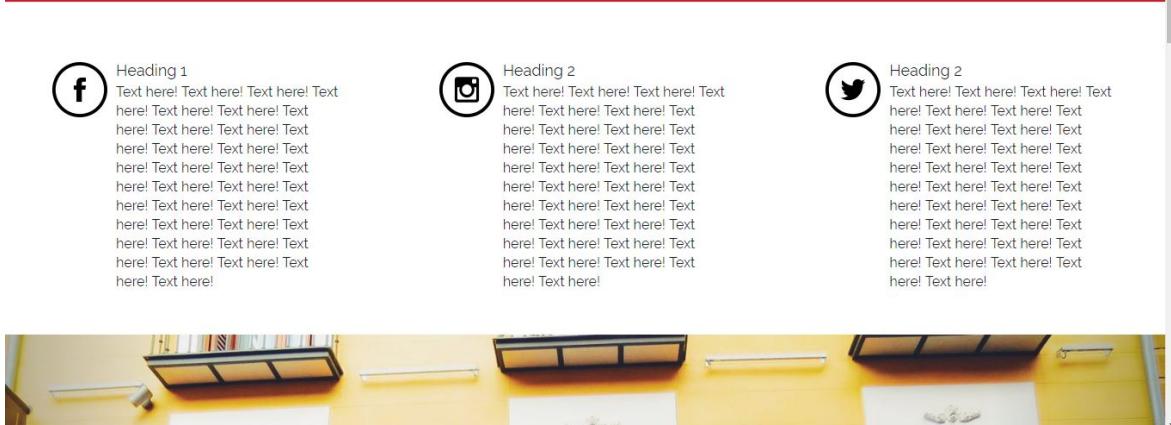
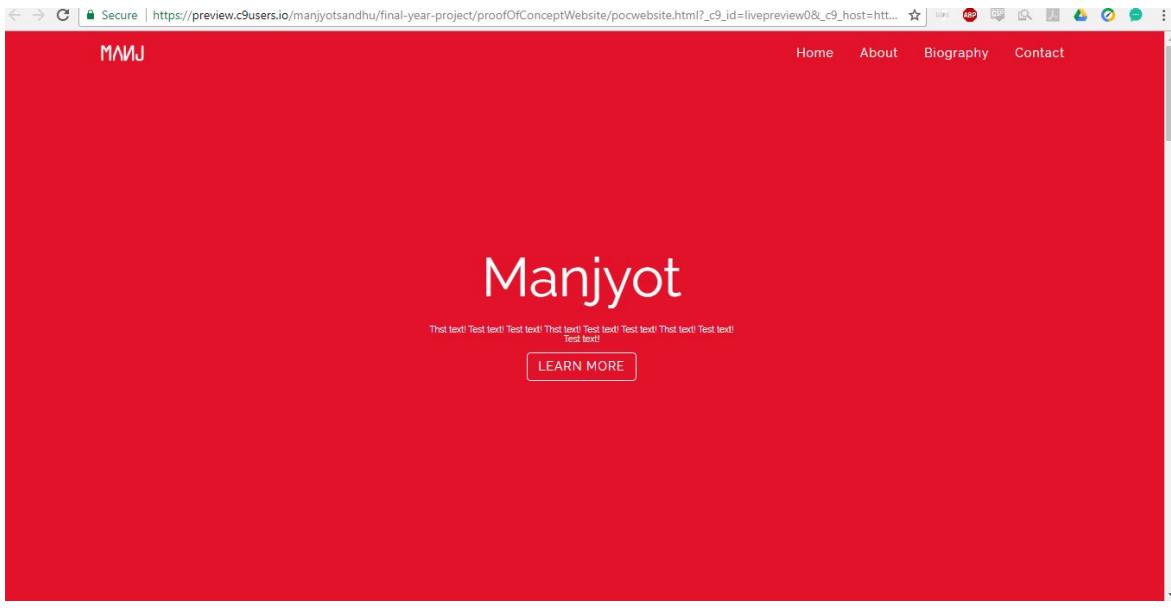


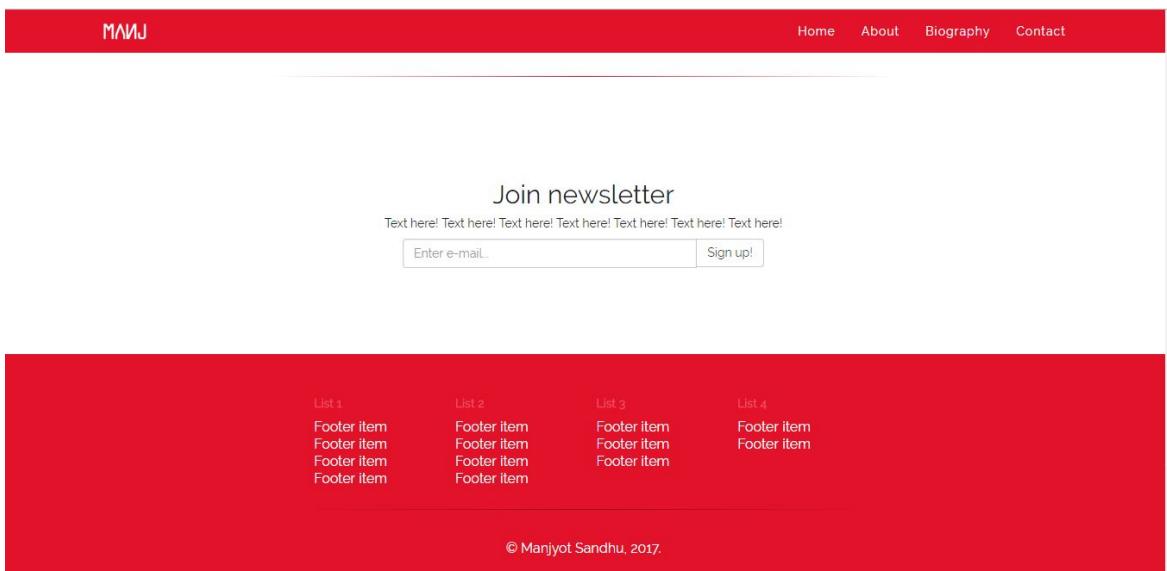
- Revise log in screen error conditions
  - Have an error symbol with hoverable error message
- Delete the search bar, just have the search button
  - The search bar is necessary clutter, as the bar can instead ‘slide out’, when the button is clicked
- Revise the maximise, minimise and close button in the top right of the screen
- Revisit the menu in the design, as it is too similar to the current sidebar
  - Have the menu filling the top of the application and keep the sidebar
- ‘Title’ of the note is not necessary, as it is already highlighted on the left side of the window
- Have the text fill the entirety of the note window, to avoid wasting space
  - Include line numbers, in the note window

## Website

The proof of concept for this website was made using a combination of HTML, CSS, Sass, JQuery and Bootstrap. My decision to use Bootstrap was based on the fact that it would provide me with many basic ‘building blocks’ for my website and other useful features. The building blocks are elements such as basic buttons and navigation bars. By using these features, I was able to construct a website layout with ease and focus on the more important details and design of the actual interface. Furthermore, Bootstrap provides many features to create dynamic layouts, with their column features, further making it easier to arrange my elements on the page. I wanted to become more experienced with Sass, so decided to use it as my CSS preprocessor. It would allow me to style my elements with ease, and implement the use of global variables.

In my proof of concept program, I created one web page and implemented the navigation bar, with various links. In the final program, all of these links will be fully functional to other sections of the website. However, for this section of the project, I just wanted to focus on working with Bootstrap on one web page.





## Project Risk Analysis

Considering the magnitude of the project and the numerous milestones, it is very likely that I could fall behind with my workload. In this project, time keeping and ensuring that targets are met is of extreme importance. To combat this, a project diary will be kept, wherein I will record my progress, so that I can draw comparisons to the intended deadlines. In addition to this, I will dedicate a certain number of hours per week to ensure that I have enough time to complete my milestones. Another risk I could encounter in this project is that I may not have the relevant technical experience with coding mobile applications. The programming for a mobile application is quite unfamiliar to me and therefore could result in problems. The chance of this risk is quite likely, therefore to combat it I will ensure that I allocate extra time for my proof of concept program for the mobile app. This will allow me to use that extra time for more learning and levelling up my skill in the subject area.

Throughout this project, I will be using a Git repository to manage my files. As a result of this, I could encounter problems with uploading and modifying files from the repository. Although the likelihood of this is quite low, I will test using Git with a test repository to revise how to use it. For the creation of the website, I will be using Cloud9 which is an online IDE. One possible risk which may arise from this is that the server could shut down at any given moment and thus I could lose my work. To combat this, I will be using Git for my work, to ensure that it is fully backed up at all times.

# Project Development

## Mobile Application

### Textual HTA

This is a textual HTA for my mobile application, with all of the intended features. For the actual implementation, some of these features may not be included, but the options will be presented, for design purposes. For this HTA it will be assumed that all features will be implemented.

#### 0. Mobile Application

1. Register for application
  - 1.1. Click on ‘Sign up’ button
  - 1.2. Input username into ‘Username’ field
  - 1.3. Input email into ‘Email’ field
  - 1.4. Click on ‘Next’ button
  - 1.5. Input password into ‘Password’ field
  - 1.6. Input password into ‘Confirm Password’ field
  - 1.7. Click ‘Register’ button
  - 1.8. Click ‘Get started’ button
2. Log into application
  - 2.1. Input username into ‘Username’ field
  - 2.2. Input password into ‘Password’ field
  - 2.3. Click ‘Log in’ button
3. Create new task
  - 3.1. Click on the ‘+’ button
  - 3.2. Input the task name into the ‘New Task’ field
  - 3.3. Assign due date
    - 3.3.1. Click ‘Due Date’
    - 3.3.2. Select desirable data from calendar
    - 3.3.3. Click ‘OK’
  - 3.4. Assign priority
    - 3.4.1. Click ‘Priority’
    - 3.4.2. Select desired priority
    - 3.4.3. Click ‘OK’
  - 3.5. Assign tag
    - 3.5.1. Click ‘Tag’
    - 3.5.2. Select desired tag
    - 3.5.3. Click ‘OK’
  - 3.6. Click the tick button
4. View tasks
  - 4.1. Click on the three lines button
  - 4.2. Click on desired type of task
    - 4.2.1. Click on ‘All’
    - 4.2.2. Click on ‘Overdue’
    - 4.2.3. Click on ‘Today’
    - 4.2.4. Click on ‘Tomorrow’
    - 4.2.5. Click on ‘Complete’
5. Sort tasks
  - 5.1. Click on the three dots button
  - 5.2. Click on desired option

- 5.2.1. Click on ‘Search’
- 5.2.2. Click on ‘Sort by’
- 5.2.3. Click on ‘Filter by’
6. Change settings
  - 6.1. Click on the three lines button
  - 6.2. Modify account settings
    - 6.2.1. Click on ‘Account’
    - 6.2.2. Edit display name
      - 6.2.2.1. Click on ‘Display name’
      - 6.2.2.2. Enter new name
      - 6.2.2.3. Click ‘OK’ or ‘Cancel’
    - 6.2.3. Edit email
      - 6.2.3.1. Same as 6.2.2, but with the email field
    - 6.2.4. Edit password
      - 6.2.4.1. Same as 6.2.2, but with the password field
    - 6.2.5. Edit email updates
      - 6.2.5.1. Click on email updates slider
      - 6.2.5.2. Same as 6.2.2 but select relevant email options
  - 6.3. Modify notification settings
    - 6.3.1. Click on ‘Notifications’
      - 6.3.1.1. Edit notification on/off
      - 6.3.1.1.1. Click enable notifications slider
      - 6.3.1.2. Edit ringtone
        - 6.3.1.2.1. Click ringtone
        - 6.3.1.2.2. Select ringtone
        - 6.3.1.2.3. Click ‘OK’ or ‘Cancel’
      - 6.3.1.3. Edit vibrate status
        - 6.3.1.3.1. Click vibrate slider
    - 6.4. View help
      - 6.4.1. Click on ‘help’
    - 6.5. View about
      - 6.5.1. Click on ‘about’
  7. Logout
    - 7.1. Click on three lines button
    - 7.2. Click on ‘Settings’
    - 7.3. Click on ‘Log out’

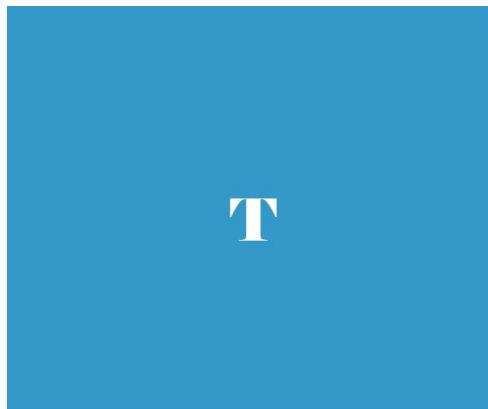
## Interface design

My final designs followed some conventions, throughout the entire application:

- Monochromatic colour scheme
  - After having completed research on colour theory, I discovered that a monochromatic colour scheme would be highly effective and aimed to implement, this throughout my application. I chose to use a faded blue for the main colour, with a darker blue for the accent colour. This accent colour would be used for the underline of text fields and various buttons. Black writing would be used for the majority of the application and white for the main body.
- Icons used
  - All of the icons I used were from the Android Asset Studio, to ensure a sense of similarity.
- Natural flow in the ordering of elements

- In any instance where data input is required from the user, such as registration, log in or a new task, the natural flow of the elements is kept in a logical order. By this, I mean that the input fields will be first, displaying items such as ‘Name’ before ‘Password’ and the confirmation buttons would be at the end of the forms. Furthermore, for the new task page, naming the task is the first field that appears on the screen, as the users cursor is automatically placed here.
- Relevant data validation on login and register
- Back buttons to leave unwanted states
  - Back buttons are on most pages, when away from the main screen, so that the user can easily return to previous pages.
- Use of colour to create visual groups
  - Different large areas of the screen have different background colours, to highlight their differences and group elements together. For example, the tool bar is always blue and the main body white. Furthermore, in the register section, the navigation bar is grey.

Log in screen



 Username

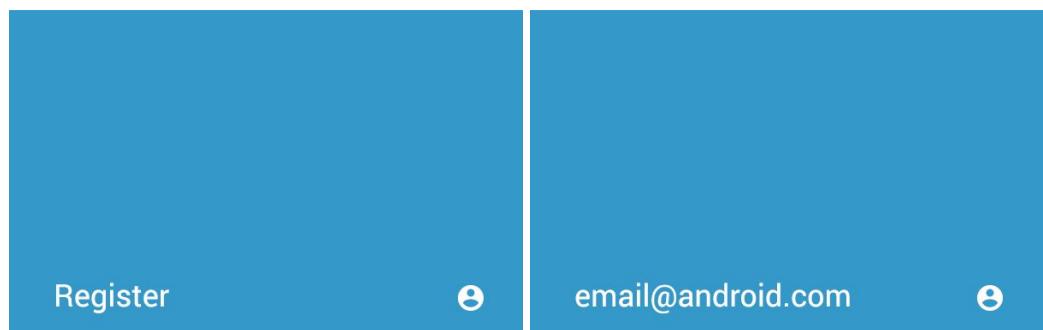
 Password

SIGN UP                    LOG IN →

- Input fields positioned at bottom of the screen
  - When a user is using a mobile device, they will most likely be holding the device from the bottom of the phone. This means that having input fields at the bottom of the screen will increase the efficiency at which they can input data.
- Feedback for incorrect data validation
  - If the data input into the login fields are not correct, an exclamation mark will appear in that chosen field. An exclamation mark usually symbolises an error, so the user will immediately be drawn to it, if they are confused as to why the login has not been successful.
  - Furthermore, a pop out will occur, when a field is focused on, so that the user can get more information about the error has occurred. Note, this pop out will ONLY occur when the field is selected, as to minimise any crowding on the screen.

- Focused input field will change colour
  - The ‘underline’ of an input field, which is currently selected, will change colour to a dark blue. This will help the user to identify which field they are currently editing.

Register screens



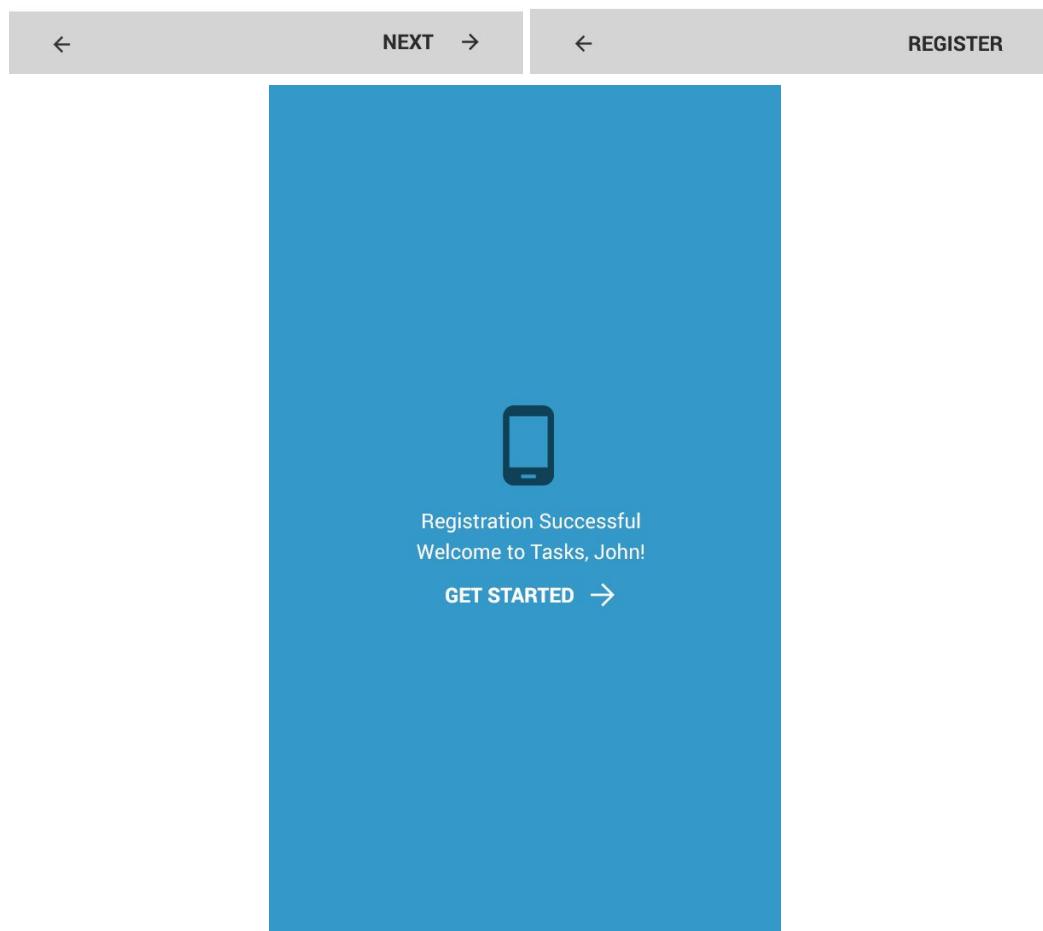
Enter your name

Passwords must be between 6-22 characters and contain one number.

Enter your email

Enter your password

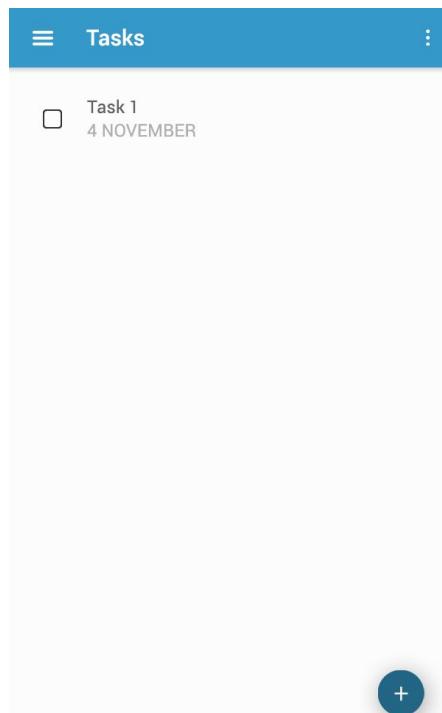
Confirm password



- 'Register' title
  - A title is used, on the first register screen, to make the user aware of the current action they are performing.

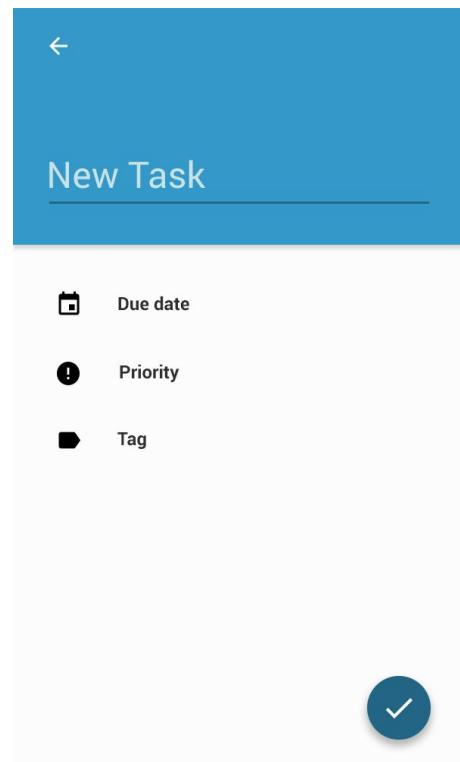
- If the user was to accidentally click on the register button, they would not be confused at what part of the application they have suddenly navigated to.
- Feedback for incorrect data validation
  - Similar concept as login screen
- Focused input field will change colour
  - Similar concept as login screen
- Similar elements are grouped together
  - In order to eliminate as much confusion as possible, similar elements are grouped together on screen, with the use of colour. In the interface so far, blue has already been seen as the ‘main’ colour, so this is used for the top bar, which contains the title. Next, a white background, which is already associated with data input (from the login screen), is used for the input fields. Finally, we have a grey bar along the bottom which is only for navigation.
- Carefully selected english to encourage user input
  - On the previous login screen, there were icons next to the entry fields which would help the user to identify that there are fields which require input from them. However, on this screen that is not the case. In order to ensure that the user is aware that these are in fact fields and not just text on the screen, I chose to use the words ‘Enter name’ and ‘Enter email’.
- Name appears in second register screen, as a title
  - With a similar idea to the title ‘Register’ on the first screen, the enter name is displayed on the second register screen. This is recognition rather than recall and the user is aware of which account details they have entered.
- The final register button is named ‘Register’ instead of ‘Next’
  - To signify completion of registration and not just the movement onto another screen, I renamed the button to ‘Register’, with no icons. This will emphasise to the user the completion of this stage.
  - Furthermore, as the users’ name is used as the title, there is no where on the screen that signifies it is still a registration page. Therefore, the have this button at the bottom of the screen will still give the users an idea of where they are in the system.
- Descriptive text on registration successful screen
  - The entire layout changes on this screen, to signify the end of the registration process.
  - It clearly states, in plain english, that the registration is successful so that the user is aware of the status of the process.
  - The name is included in this block of text, so that the user can be sure of what name they used.
- On registration complete page, the ‘Get Started’ button is in a bigger font
  - To show the difference between text and a button, the ‘Get Started’ is in a bigger font and uses a forward arrow, to emphasise this.

## Main Page



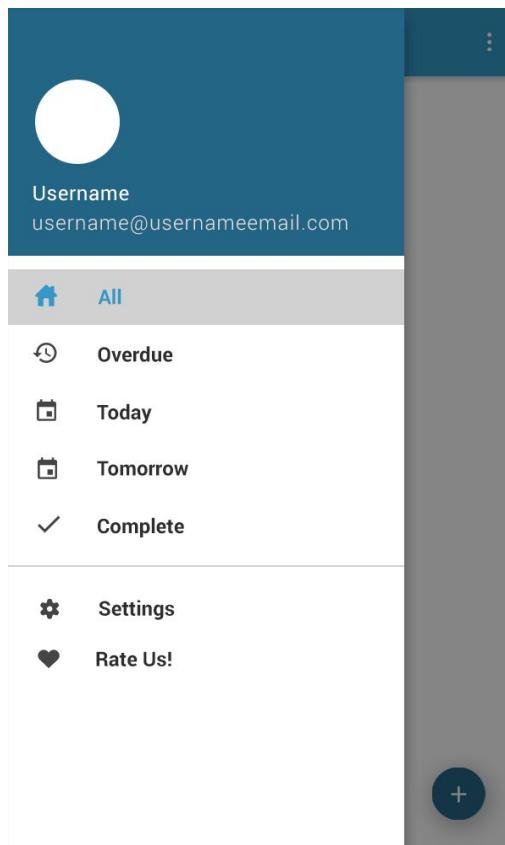
- Navigation drawer button positioned in top left
  - The navigation draw slides out from the left side, which is why this button is positioned on this side of the top bar
- ‘Tasks’ title
  - Title to make it clear to the user what they are viewed
- Display of ‘No Tasks’ status
  - If there are no tasks created, an icon will appear and text to signify that there are no tasks. This will help users to understand why the screen is empty, instead of it just being a blank white screen
- Different font styles for task title and date
  - The task name is a darker colour, to show that this should be read first and further encourages the ‘logical’ order of the elements
- Floating add task button in bottom right of the screen
  - Based on the most typical way that the application will be held, the user can simply use their thumb to quickly create a new task. As a result of this positioning, it increases the efficiency.

New task



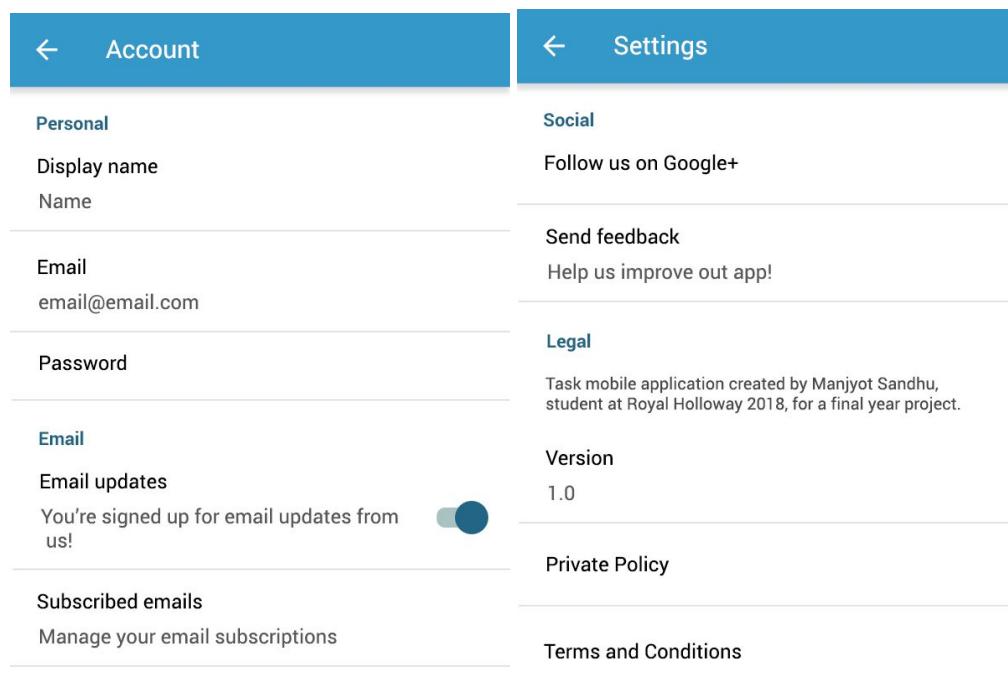
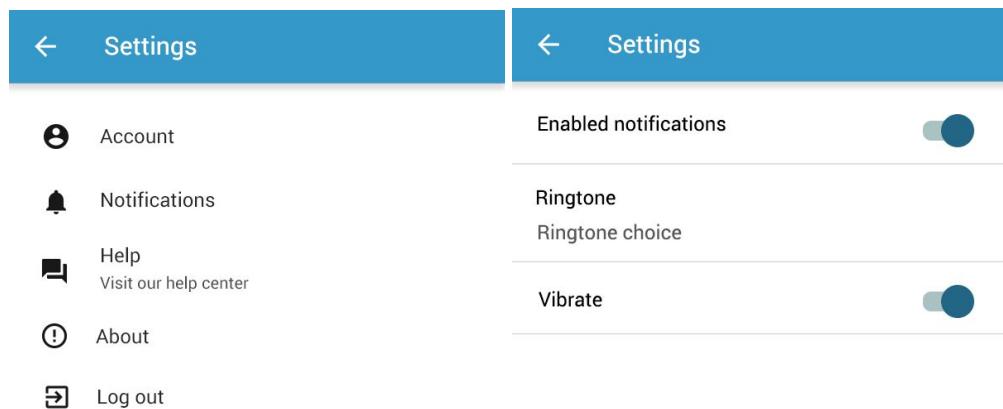
- Cursor automatically starts in 'New Task' field
  - When the screen loads, the cursor is automatically placed inside the 'New Task' field, which prompts the user to start off with this input, further supporting the idea of a logical flow
- 'New Task' in a large font
  - Helps the user to identify what process they are going through
  - Symbolises that the entire page is about the creation of a new task
- Pop ups for each extra option
  - When clicking on 'Due date', 'Priority' or 'Tag', a pop up will appear wherein the user can continue on to complete the necessary input. Pop ups are used to reduce the amount of space taken up and clutter present on the original screen.
- 'Cancel' button to help leave unwanted states, in each of the pop ups
- Floating button for creating the new task
  - Similar ideas as the create a new task button, on the main page

### Navigation Draw



- Current logged in users details at top of navigation draw
  - This adds a sense of familiarity for the user, as they can see their own details displayed here. Furthermore, it also tells the user which account is logged in.
- Display of system status through use of grey background for active item
  - The item which is currently active will have a grey background and the font/icon colour will be changed to a light blue colour. This will help the user to identify which option is selected, allowing recognition rather than recall. The user will not have to close the navigation draw to see which page they are currently on
- Different types of navigation are separated by a line
  - All of the items above the grey separator are those which can have ‘active’ states and selecting them will simply change a portion of the main screen.
  - The items below the separator will direct the user to a different screen entirely.
- Rest of screen is greyed out
  - The rest of the screen, behind the navigation drawer, is greyed out so that the user is aware of which part of the screen is active
- Shortcuts for seasoned users
  - To exit the navigation draw the user can select of the many options, which will direct them to the new page and automatically close the navigation drawer.
  - On the other hand, the user can simply click the ‘greyed’ out background and this will close the navigation drawer. This idea may only be known to an more experience user, who is familiar with android applications. It is always important to include shortcuts for more seasoned users.

## Settings



- Summaries for headings which are not self explanatory
  - Some of the headings are self explanatory, such as ‘Vibrate’, which is clearly an option to turn off and on vibration. On the other hand, some headings such as ‘Help’, on the main settings screen, can lead to some confusion. To combat this, a

short summary is given so that the user has some idea of what clicking on this option will do.

- Use of many setting screens
  - As there are many different settings, it would be too long to contain on one page. To combat this, the settings are implemented across a variety of different pages, which can be accessed from the main settings page
- Display of settings status
  - For some settings, summaries are used to show what the current ‘status’ is. Examples:
    - Email updates summary tells the user if they are subscribed to emails
    - Current name and email are displayed under the relevant headings
  - This again supports the idea of recognition rather than recall
- Timed pop ups when a setting is updated
  - When a setting is updated, a pop up will appear on the bottom of the screen, for a specific amount of time. This pop up tells the user which setting was updated and with what contents. The user can still use the app, even whilst the pop up is on the screen, as it does not interfere with any usability.
  - This helps the user to be aware that the setting did in fact update
- Pop up dialog for updating settings
  - To further reduce clutter on screens, pop up dialogs are used to update any settings. A title of the field which the user is updated is displayed at the top of the dialog, to support recognition rather than recall
  - A cancel button is also supported on these pop ups, so that the user can leave any unwanted states.
- Unavailable options are greyed out
  - If any option is unavailable, such as selecting a ringtone if the enabled notifications button is not ‘ON’, then the field will be greyed out and unclickable. This will make it clear to the user that this field is unavailable
- Section titles to break up content
  - In some screens, where there are lots of options, section titles are used to divide up the screen
- Settings title change, when on different setting pages
  - The title will change accordingly, depending on which setting page the user is on. This will help the user to identify which parts of the settings system they are on
- Pop ups will grey out the screen behind it
  - Similar concept to the navigation drawer
- Shortcuts for seasoned users
  - Similar concept to the navigation drawer, wherein pop ups can be exited through ‘Cancel’ or ‘OK’, but also from clicking the grey background
- Field is highlighted grey when clicked on
  - When a field is clicked on, it will be highlighted grey briefly, in order to show the user that their actions are actually happening

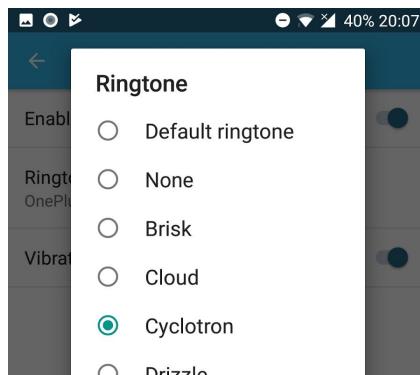
## Final Product

*Full screenshots of the final product can be found in the appendix*

There were a few features I was able to implement in my application, based on what Android Studio had to offer me:

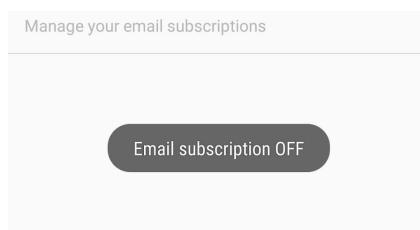
- Popup ringtone preference

When selecting a ringtone, in the settings section, a popup will appear. In this pop up, all of the available ringtones available on the device will appear. If one is to click one, the application will play a musical preview of the ringtone



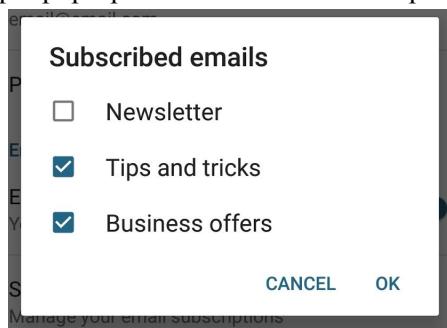
- Toasts

In Android Studio, there is a feature called ‘Toasts’ which allow small pop ups to appear at the bottom of the screen, for a designated amount of time. I programmed this to appear each time a setting was updated, with the relevant setting and new content.



- Multi option pop-up

I was able to implement a simple pop-up box which allows multiple checkboxes to be selected.



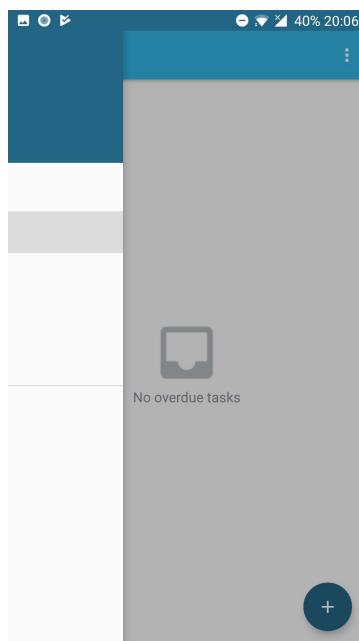
- Pop up dialogs to update text settings

I implemented pop up dialogs which allow text to be edited and saved. In the popup, the title will be the setting that is being updated, which allows recognition rather than recall for the user.



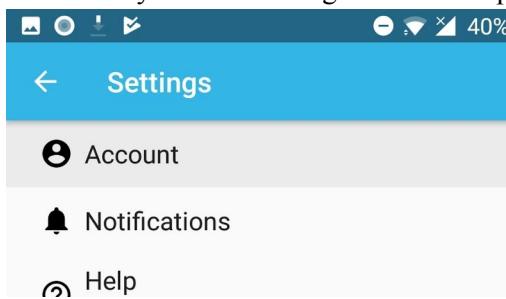
- Sliding navigation draw

The navigation drawer is able to be slid closed or clicked close. Furthermore, the drawer is dynamic and can be resized



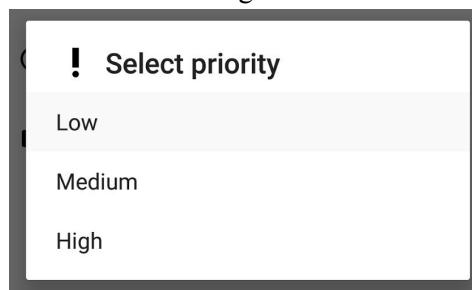
- Feedback when clicking on a field

In the settings screen, when an option or button is clicked on it will be highlighted in grey to give feedback to the user that the actions they are committing are in fact happening



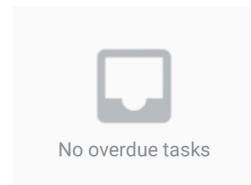
- Recognition rather than recall on the new task screen

If a priority is selected and the user is to reopen the priority dialog, the current choice will be highlighted in grey so that the user is able to recognise what is the selected state.



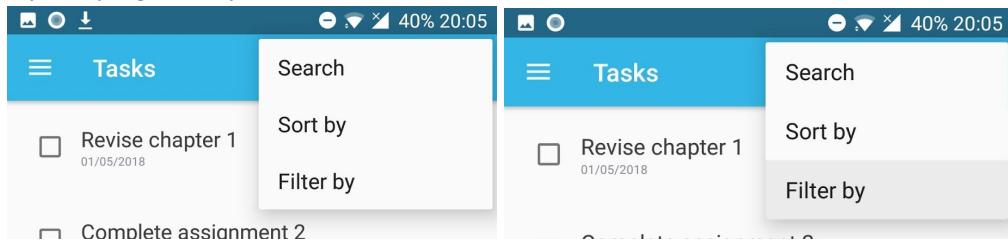
- Representation of an empty task list

If there are no tasks, a grey icon will appear with text that will tell the user there are no tasks. This will lead to no confusion as to why there is a large empty white space area. I believe that there should not be wide empty spaces, with no content, otherwise users will get confused or be led to believe there is some kind of bug



- Popup menu

A popup menu will appear when the three dots icon is clicked. Similar concept with feedback as clicking on fields, wherein selected options will be highlighted grey. This menu can be navigated against by carrying out any other action.



- Data validation for input fields

I was able to implement feedback for all data validation in a very successful way. For each field, when focused on the field underline turns a dark blue, to match the theme. If the input field is invalid, a red error icon will appear at the end of the field. When focusing on the field, a popup will appear which tells the user, in greater depth, of what the error is. The error is in plain english, so even novice users can understand



# GUI

## HTA Diagram

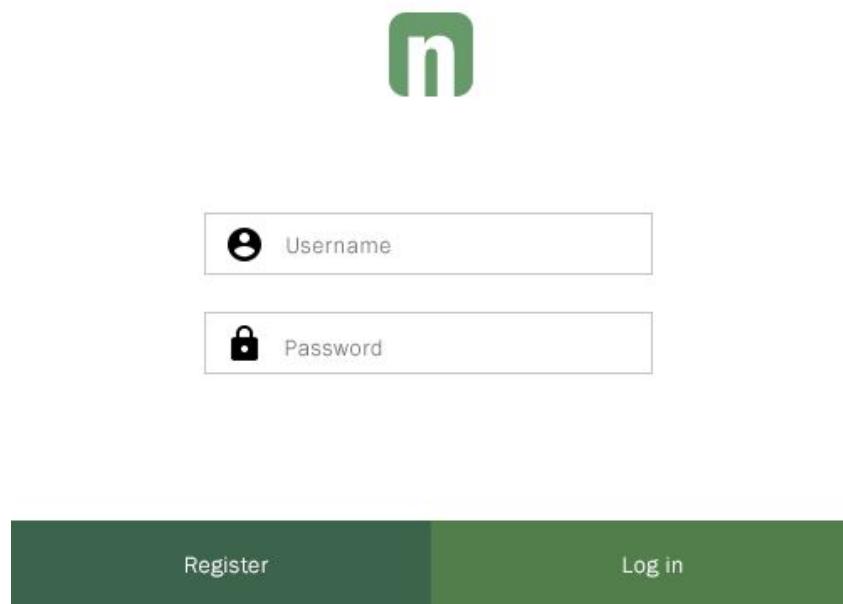
### 0. GUI

1. Register
  - 1.1. Click the 'Register' button
  - 1.2. Enter username in 'Username' field
  - 1.3. Enter password in 'Password' field
  - 1.4. Enter password in 'Confirm Password' field
  - 1.5. Click the 'Register' button
2. Log in
  - 2.1. Enter username in 'Username' field
  - 2.2. Enter password in 'Password' field
  - 2.3. Click the 'Login' button
3. Create new note
  - 3.1. Click the menu icon

- 3.2. Click on ‘New Note’
- 3.3. Enter the title into the ‘Title’ field
- 3.4. Select a privacy type
- 3.5. Select if collaborative
4. View a note
  - 4.1. Locate the note on the side bar
  - 4.2. Click on the note
  - 4.3. View note
5. Adjust view
  - 5.1. Click the arrows on the draggable bar
6. Logout

## Interface design

- Monochromatic colour scheme
  - Throughout my GUI, I kept the colour scheme very simple and used whites, greys, blacks and hues of green. This was to keep the design as minimalistic as possible and not overwhelm the user. Any use of excess colour could also draw attention away from certain elements on the screen
- Logical flow of elements
  - When there was a form to be completed, such as the register screen, the elements were in a natural logical order. By this I mean that the username would come first, followed by the password fields. Any confirmation buttons or cancel buttons were kept at the bottom of the GUI.
- Use of different font sizes and colours to emphasise importance
  - Larger fonts were used to signify a page title, such as on the register screen. The title ‘Register’ is much larger than the subtitle, which describes the contents of the page. As a result of this larger title, the user will more likely be drawn towards this first.
- Well recognised icons to represent fields and actions
- Clean and aesthetic design



- Hints in the input fields to signify that they are input fields
  - If the input fields were simply white, empty boxes, users may get slightly confused and not realise that they are in fact input fields. By supplying small, basic hints, it will encourage the user to enter the correct data.
- Login button positioned in the bottom right
  - The most ‘dominant’ button on an interface is usually contained in the bottom right corner of the interface, which is why ‘OK’ buttons are usually placed there. As a result of these conventions, the user would have a sense of familiarity to this location. I placed the login corner, as to not raise any confusion. For example, the user could spend time filling out the login form only to click ‘Register’ instead of ‘Log in’, if the buttons were reverse

The image shows a mobile application's registration screen. At the top, the word "Register" is displayed. Below it, a subtitle reads "Register an account with us to save notes on the go!". There are three input fields: "Username" (with a person icon), "Password" (with a lock icon), and "Confirm Password" (with a lock icon). Each input field has a red border and a red error icon to its right. Below the "Username" field, a message says "Please enter username". At the bottom, there is a dark grey footer bar with two buttons: "Cancel" on the left and "Register" on the right.

- Form validation
  - If an input field has incorrect validation, three things will happen: a red border will surround that field, a textual error will appear underneath it and a red error icon will appear to the right of it. All of these error methods are very localised, instead of a generic popup dialog with an error message. As a result, it is much more clear where the error persists.
  - The colour red is commonly associated with danger or errors, which is why it is suitable for this purpose
- Grouping of similar elements
  - Similar elements are grouped, such as the password fields. This signifies ‘sections’ in the interface, without having to include boxes or lines. It gives a mental note to the user that they have made progress.

Continuing on, Toledo [5] suggests that specific colours can convey various emotions to the user. For example, using a bold red colour can symbolise power and passion, which is perhaps suitable for a campaign website. Orange is a bright and vibrant colour, which can be used to gain immediate attention. The colour can also be seen as somewhat energetic and much less 'aggressive' than red. Yellow has similar characteristics to orange, wherein it can be used to demand attention. However, using yellow in a design should be done sparingly as it can be overwhelming, when used in large quantities, or hard to read text over it. Shades of greens and blues can be seen as quite calming colours, which are suitable for most situations.

- Relation to the real world
  - I designed the note section to be almost like a book, to relate it to real life. The text space is large and white, while there is a title at the top.

### New note

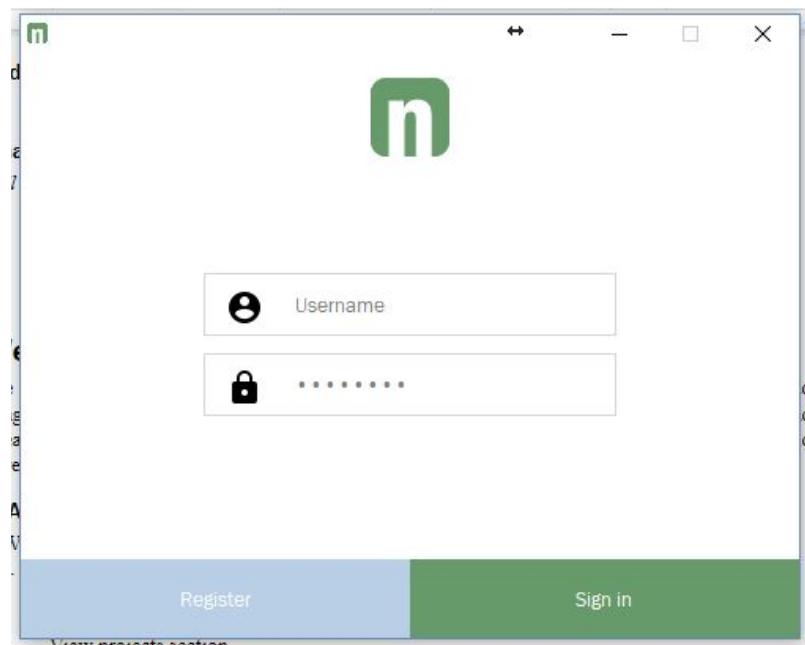
Title

Privacy  Private  Public  Password protected

Collaborative

### Final Product

Full screenshots can be found in the appendix section



- When a button is held down on, it will change colour to blue. This gives the user feedback of what action they are carrying out.
- When the user hovers over the 'X' in the corner, it will turn red, which represents cancellation and will terminate the window
- The minimise button is greyed out, as it is disabled

## Register

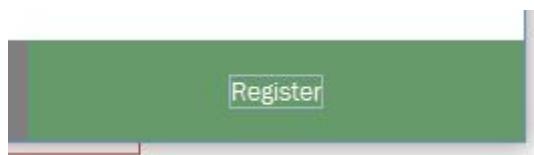
Register an account with us to save notes on the go!

The image shows a registration form with three fields. The first field, labeled 'Username', has a user icon and contains the placeholder 'Username'. A red exclamation mark in a circle is positioned to the right of the input field. Below the field, the error message 'Username not entered' is displayed in red. The second field, labeled 'Password', has a lock icon and contains four placeholder dots ('....'). A red exclamation mark in a circle is positioned to the right of the input field. Below the field, the error message 'Password not entered' is displayed in red. The third field, labeled 'Confirm Password', has a lock icon and is empty. A red exclamation mark in a circle is positioned to the right of the input field.

- A look at how data validation shows on my application, as explained in my designs

The image shows a registration form with two fields. The first field, labeled 'Password', has a lock icon and contains four placeholder dots ('....'). A red exclamation mark in a circle is positioned to the right of the input field. Below the field, the error message 'Please confirm password' is displayed in red. The second field, labeled 'Confirm Password', has a lock icon and is empty.

- There are custom messages, depending on the error



- When a button has been clicked on, it will become ‘focused’ and a box will appear around it

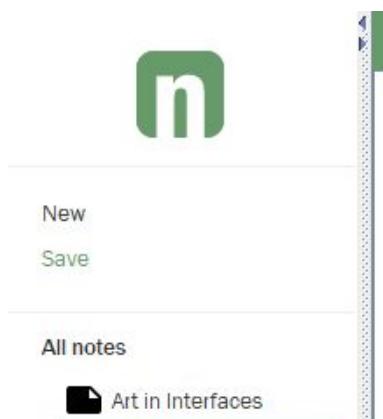


- When a text field is active, a blinking cursor will appear. This is a very common convention in almost all computer based applications. It will be easily recognisable to the user and signify that they can edit the field.

On the other hand, Rocheleau [6] argues that emotions through colour is a method to convey a specific message to the audience. It can be argued that this is the measure of dark to light on the spectrum. This value is what helps which in turn creates a sense of hierarchy and order on the interface. The general rule, a colour being used for a large portion of the interface should reflect Rocheleau's words, a more "washed" out colour is more effective. When integrated into a black and white scheme, to ensure readability of the theme itself. This can be achieved by using different levels of contrast.

In addition to all of these complex considerations one must take when designing a website, it is important to remember that pure black should not be used in large quantities. Ian Storck [7] states that:

- When selecting text, it will change background and text colour, showing the selection



- When hovering over button links they will change colour, which will give feedback to the user

## Website

The website I aimed to create was a simple website which showcases some basic content and images, to act as a kind of portfolio. The idea would be that the website focus on a minimalistic and 'clean' design, whilst being able to successfully convey the desired content. The website would have a large starting page, with a navigation bar at the top of the page.

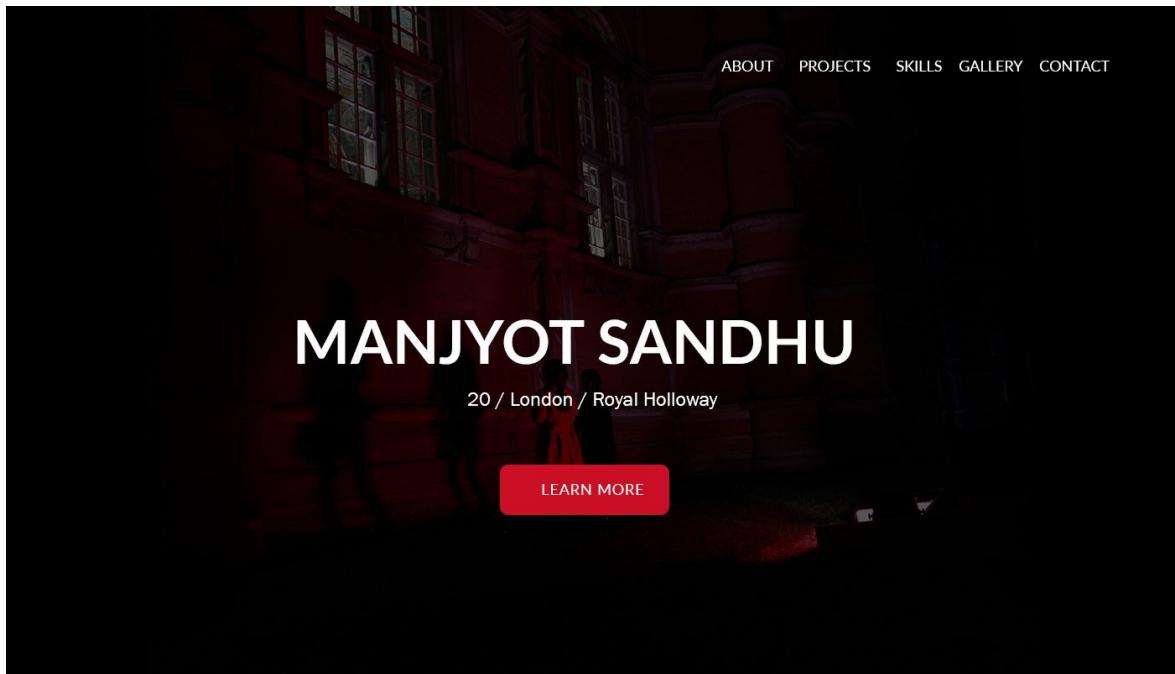
### HTA Diagram

0. Website
  1. View about section
    - 1.1. Click 'Learn More' button in main header OR
    - 1.2. Click 'About' button in navigation bar
  2. View projects section
    - 2.1. Click 'Projects' button in navigation bar
  3. View skills section
    - 3.1. Click 'Skills' button in navigation bar
  4. View gallery section
    - 4.1. Click 'Gallery' button in navigation bar
    - 4.2. View picture location details
      - 4.2.1. Hover over picture for location details
  5. View contact section
    - 5.1. Click on 'Contact' button in navigation bar
    - 5.2. Send a message

- 5.2.1. Enter name in ‘Name’ field
- 5.2.2. Enter email in ‘Email’ field
- 5.2.3. Enter message in ‘Message’ field
- 5.2.4. Click ‘Send Message’ button

## Interface Design

- Monochromatic colour scheme
  - The colour scheme was kept very simple, using red hues throughout. The main body was kept as white and light grey, whilst a dark grey was used for the text. Any accent pieces, such as buttons and active navigation links were coloured red.
- Logical flow of elements
  - The flow of elements was in the correct order, as they were displayed on the navigation bar
  - Furthermore, the contact sections and additional personal links were at the bottom of the page
- Use of different font sizes to display importance
  - Any titles were in uppercase and a considerably larger font, to highlight their importance. This use of larger font would draw the users attention first and therefore enable them to read this content first.
  - Subtitles had a lighter grey text, which was not as noticeable as the titles they followed.



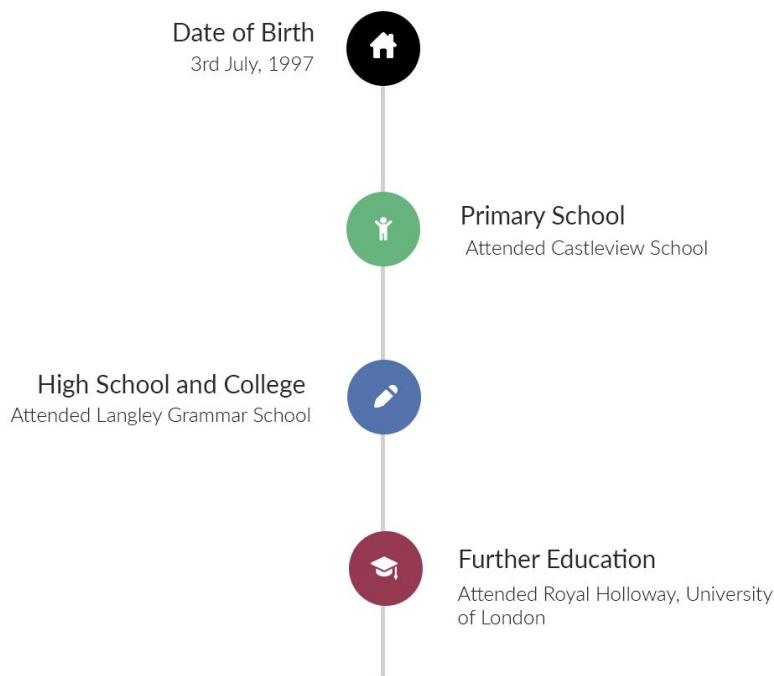
- Long navigation bar, covering the width of the screen
  - To increase efficiency, when using the navigation bar, the width stretches across the entirety of the screen. Whenever the user is on the screen, they can simply move their mouse to the top of the screen and access the navigation bar. Having a greater surface area of the bar will decrease the time it takes for the user to reach the specified element.
- Moving navigation bar
  - To further increase efficiency, the navigation bar is ‘stuck’ to the top of the screen, so that no matter how far down the page the user is, they will be able to access it. If

the navigation bar was only at the top of the page, time would be wasted when scrolling all the way back to the top.

- Navigation bar display of system status
  - The current section which the user is on will be in red, on the navigation bar, which helps the user to identify what position they are on in the webpage.
- Use of grid and center positioning
  - Through the design, center positioning and a ‘grid’ layout was used, to produce a clean interface. The use of a grid system also helps to separate out the elements and create divisions to the web page.

## ABOUT

*Timeline*



### Duke of Edinburgh

In progress of completing the Duke of Edinburgh awards in Silver.

[Find out more](#)



### Maths challenges

Achieved Silver in both Junior and Senior math challenges

[Learn more](#)



### Interests

My other interests include piano, art and photography

[Visit my VSCO](#)

- Use of appropriate icons
  - For the timeline, I used icons which were suitable for the purpose. For example, to represent the ‘Further Education’ symbol, I used a graduation cap icon. This would help the user to gain a sense of real life relation.

## PROJECTS

*Personal work and University projects*

**Photoshop design**  
Self taught photoshop for personal use. Use for designing typography, posters and various photo manipulations

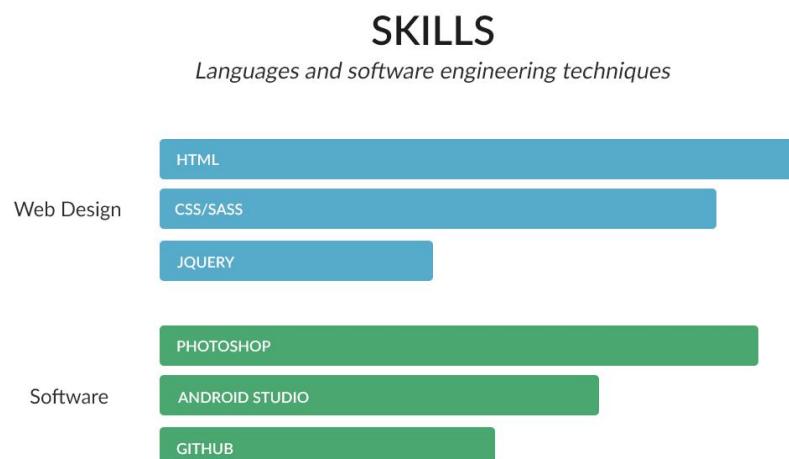
Photoshop

**University Group Project**  
In my second year, I took part in a larger group project where I learnt how to use and implement GitHub with my work.

Teamwork, JavaFX, GitHub

**Android Mobile Application**  
I designed and created a task mobile application, as part of my final year project

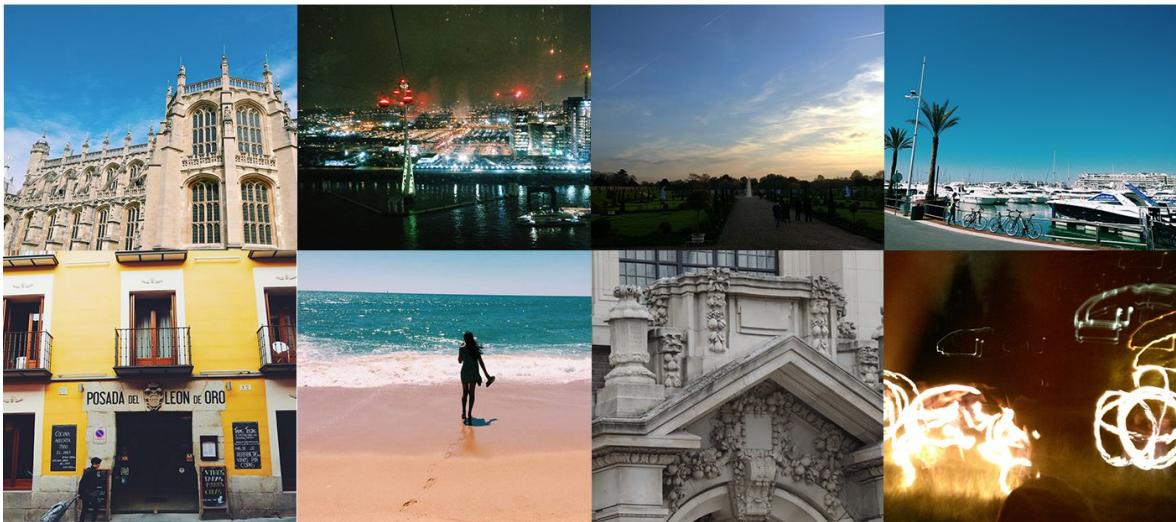
Java, Android Studio



- Use of background colours to divide content
  - Between main sections, the backgrounds would alternate between light grey and white. In addition to the titles and this feature, it would help the user to recognise the transition between different sections.
  - Furthermore, in the project’s section, the skills are kept separate from the main body. This helps to group similar elements and content together.
  - In the skills section, the same colour is used for each different section

## GALLERY

Photographs from around the world



- Extra information available upon hover over gallery pictures
  - In order to reduce the units of information on the page, any not vital information is available from hovering over the gallery images. This reduces clutter and results in a ‘cleaner’ interface

## CONTACT

*Send me a message*

Maximum 250 characters.

SEND MESSAGE

[✉](#) [📷](#) [💬](#)

Manjyot Sandhu, 2018

- Choice of english with the input fields
  - The words for the input fields had ‘Your’ placed in front of each field, so that it was clear the user needed to input their details. In this instance, having ‘Your name’ instead of ‘Name’, makes it much clearer to the user that is infact a field which should have input from them.
- Message character limit placement underneath message field
  - The character limit message expressing that there must be a maximum of 250 characters, is placed under the message box instead of on top. This will avoid confusion as to which box the message applies to.
- Input box is highlighted, when selected
  - Helps the user to recognise which input field they are currently in
  - A useful display of system status
- Data validation feedback
  - Upon the click of the ‘Send Message’ button, there will be valid data validation feedback for the user. Any successful fields will be outlined in green and have the message ‘Looks good!’ underneath it. Any unsuccessful fields will be outlined in red and have an error message underneath it. The use of these colours are well known to users and have specific meanings.
- Error messages in simple, easy to understand english
  - If an email is invalid the error message underneath will simply read ‘Please enter a valid email’, which is easily understood even by the most novice users.

## Final Product

### Code

This website was made using a combination of HTML, CSS, Sass, JQuery and Bootstrap. My decision to use Bootstrap was based on the fact that it would provide me with many basic ‘building blocks’ for my website and other useful features. The building blocks are elements such as basic buttons and navigation bars. By using these features, I was able to construct a website layout with ease and focus on the more important details and design of the actual interface. Furthermore, Bootstrap provides many features to create dynamic layouts, with their column features, further making it easier to arrange my elements on the page. I wanted to become more experienced with Sass, so decided to use it as my CSS preprocessor. It would allow me to style my elements with ease, and implement the use of global variables.

One particular part of the code that was interesting to me, which I had to research and implement in my own way, was how to detect if an element was visible from the viewport. I needed this function so that I could animate certain elements, once they came into view.

```

var top_of_element = $(element).offset().top;
var bottom_of_element = $(element).offset().top + $(element).outerHeight();
var bottom_of_screen = $(window).scrollTop() + window.innerHeight;
var top_of_screen = $(window).scrollTop();

if ((bottom_of_screen > top_of_element) && (top_of_screen <
bottom_of_element)) {
    return true;
}
else {

```

```
    return false;  
}
```

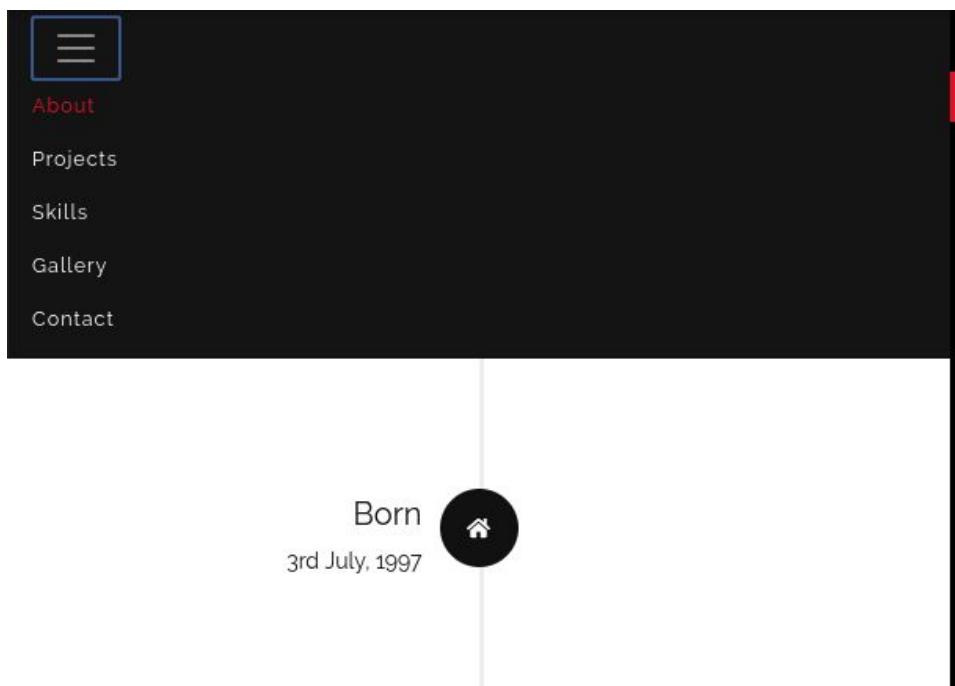
- `top_of_element`
  - Calculated by retrieving the offset of the element from the top, which is the distance from the top
- `bottom_of_element`
  - Calculated by adding the distance from the top of the screen and the outer height of the element, which includes all margin and padding
- `bottom_of_screen`
  - This is the distance scrolled from the top in addition to the height of the window
- `top_of_screen`
  - Distance scrolled from the top of the screen

If the bottom of the screen is larger than the top of the element then this means the element has scrolled into view. Also, if the top of the screen is smaller than the element then this means that the viewport has not completely passed the element. If both of these conditions are fulfilled then the element is in view.

### Final program

*Final product screenshots can be found in the appendix*

When designing my website, I was careful to ensure that it could adapt to many different screen sizes. First and foremost, the navigation bar is collapsible. When the screen is resized so a certain amount, a new button will appear for which all the buttons can be accessed. This is compatible with mobile devices also.



- The projects section goes from 3 columns to 1, in a smaller device

The image shows two side-by-side screenshots of a mobile application interface. Both screens have a red header bar with a white 'PROJECTS' title and a subtitle 'Personal work and University projects'. The left screen is for a larger screen and displays three project cards in a grid: 'Photoshop design' (self-taught Photoshop for personal use), 'University Group Project' (participated in a large group project), and 'Android Mobile Application' (designed and created a task mobile application). The right screen is for a smaller screen and shows a single card for 'Photoshop design' with a sub-section for 'Personal photography'.

**PROJECTS**  
*Personal work and University projects*

**Photoshop design**  
Self taught photoshop for personal use. Use for designing typography, posters and various photo manipulations.

**University Group Project**  
In my second year, I took part in a large group project wherein

**Android Mobile Application**  
I designed and created a task mobile application as part of

**Photoshop**

**Personal photography**  
During the course of my art projects at A Level, I did a

- The gallery images go from 4 in a row to just 1 in a row, in a smaller screen

The image shows two screenshots of a mobile application interface. The top screenshot is for a larger screen and displays a 2x2 grid of four photographs: a yellow building, a person walking on a beach, a marina, and a sunset. The bottom screenshot is for a smaller screen and shows a single large photograph of the yellow building.

**GALLERY**  
*Photographs from around the world*

**GALLERY**  
*Photographs from around the world*

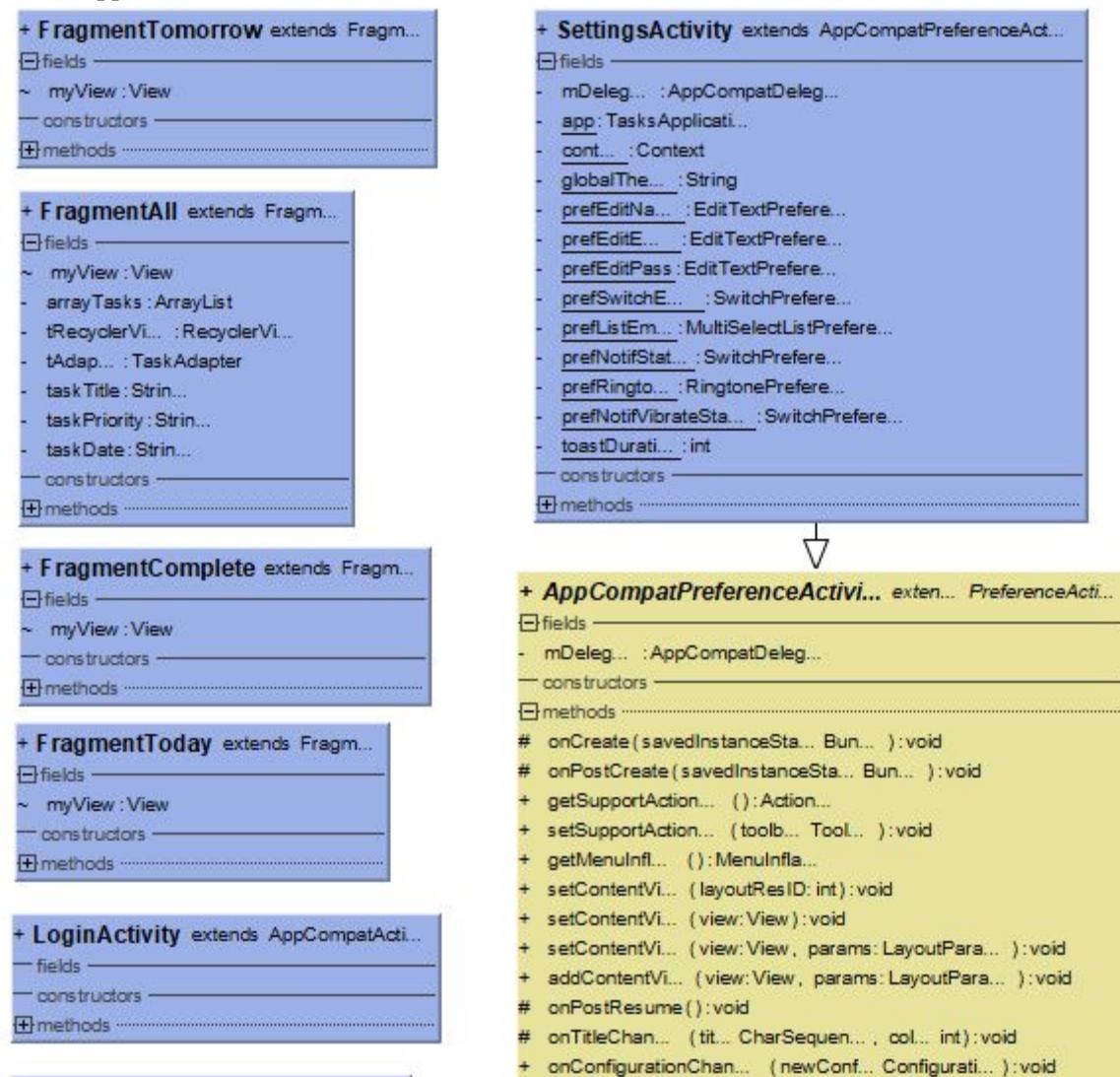
- The contact input fields also resize with the window size

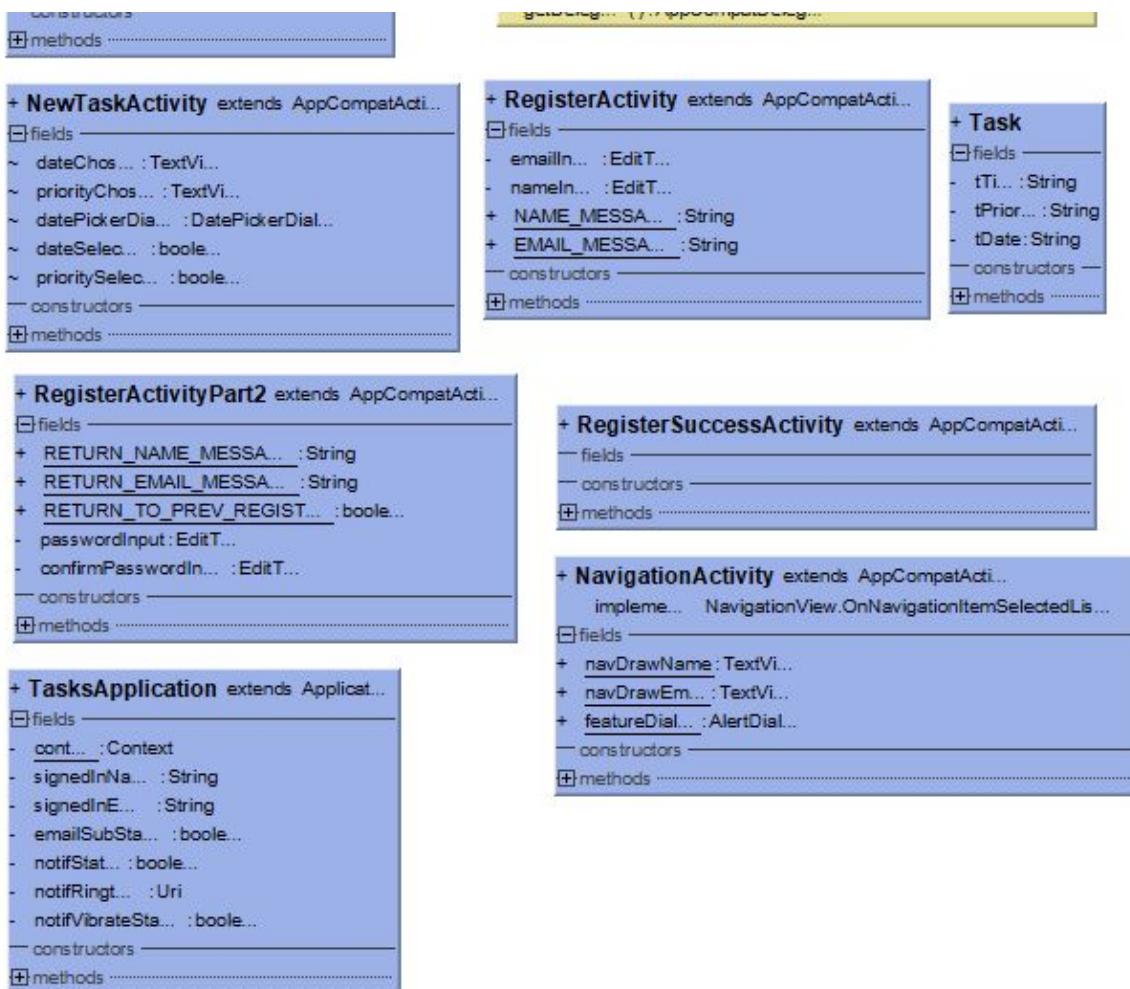
# Software Engineering

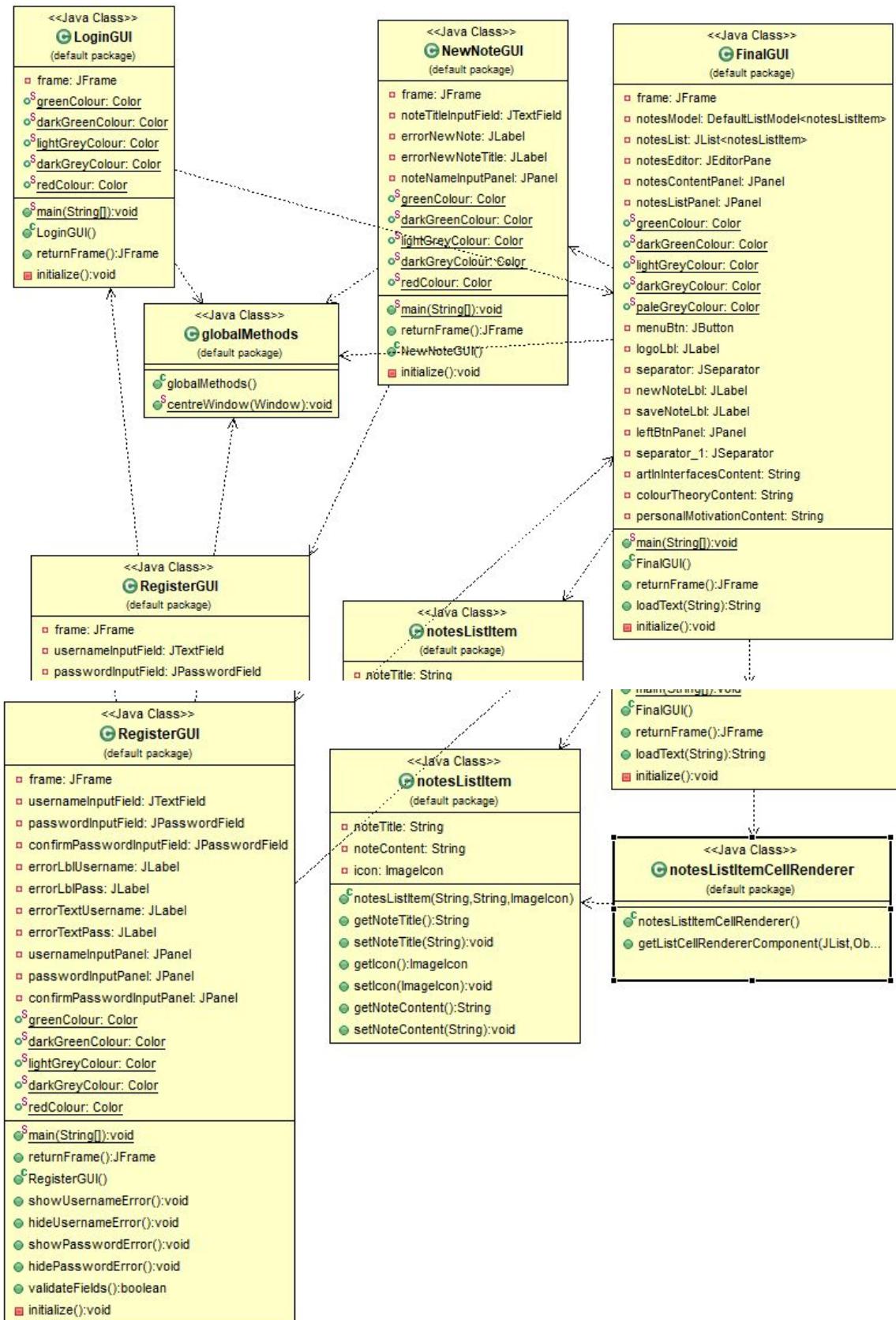
*Sections describing the software engineering method that you used. If your project is based on a software product then this may even be most of your report*

## UML

### Mobile application



GUI



## Software Engineering Method

As I was not working in a team on this project, I could not fully carry out most software engineering methods. However, one method which I took ideas from is extreme programming. Extreme programming is related to putting customers requirements first and constant feedback. The main goal from this type of software engineering is to maximise customer satisfaction, which works in my favour as my project is HCI related and focused on the user experience. Although I did not have any official stakeholders in my project, I consistently communicated with others for their opinions on my applications. I would frequently partake in short demonstrations and testing on real life users, to see how they reacted to my software. I made sure to test on a large age range of people, as I found that novice users had a very different perspective as to those who are more seasoned to computer systems. It was in fact the novice users who would ask more questions and prompt me to consider certain aspects of my designs. This is probably the biggest aspect of extreme programming I took into account.

Another ideal of extreme programming is releasing iterations and small sections of the project. This was certainly achieved in my project, as my project plan clearly outlined that I would work to certain stages in each of my interfaces and then come back to them later. Furthermore, iterations were achieved through my completing my proof of concept programs. I also was very focused on refactoring code constantly. I ensured to always be checking my code, commenting and removing excess code where necessary. If there was a way in which I thought I could refactor code and make it simpler, I would do it on the spot. These are all ideals I took from extreme programming.

## Developer Environment

### Design

- Photoshop

Throughout the course of my project, I utilised a number of different environments to create my interfaces. I had various needs, which could only be met by many different tools. First and foremost, to create my designs I needed to use a photo editor, so that I had complete freedom over my designs. As I had previous experience with Photoshop and had high confidence in myself to create my designs how I wanted, I decided that this would be the best choice for me. Photoshop allowed complete flexibility over my designs and any changes to my design could be carried out easily.

### Website

- Cloud9
- Bootstrap

To develop my website, I wanted an easy way to be able to develop my code and easily preview it. Cloud9 is an online IDE which I used to do this. I was able to preview my website easily, after minor changes, whilst integrating it with GitHub.

### Mobile Application

- Android Studio

Whilst developing my Android Application, it made perfect sense to use Android Studio as it would ease the process of development for a novice, such as myself. The application comes with many useful features and premade classes for various activities. There were also many tutorials online, which proved to be helpful to me.

### GUI

- Eclipse
- WindowBuilder plugin
- Checkstyle plugin

Eclipse, having being a reliable piece of software that I am well acquainted with, seemed like the perfect choice for me. I was familiar with many of the features, which would help me with my development. Eclipse also comes with many available plugins from the marketplace, which proved useful to me. I used the WindowBuilder plugin to help me with the creation of my GUI. This plugin was extremely vital to the process, as it allowed me to edit my design with both the back end code but also through a more visual design page. The visual design screen was more of a ‘drag and drop’ style development tool, meaning it was extremely easy to visualise how my application would look. Furthermore, I used the CheckStyle plugin to ensure that my code followed certain conventions and formatting.

### GitHub

- GitBash
- GitHub

For this project, I used GitHub to act as a backup for my work, along with all of the other features Git will allow me to use. (GitHub link: [https://github.com/RHUL-CS-Projects/FullUnit\\_1718\\_ManjyotSandhu](https://github.com/RHUL-CS-Projects/FullUnit_1718_ManjyotSandhu)). To further use GitHub and integrate it into my applications, I utilised GitBash. GitBash acts as a terminal which simply allows me to easily push and commit to my repository. GitHub came in very useful for my project, as it provided me with a safe way to store my code if anything were to happen to it. Furthermore, I ran into a situation where I had accidentally deleted part of my project and I could simply revert it back using my repository.

## **Professional Issues**

During the course of my project, I had a lot of learning to do and with this came the need for me to expand my knowledge using various tutorials and code sources. When you are learning something entirely new, such as Android Studio, it is difficult to ensure that you properly learn the content instead of copying it. It became apparent to me that I had to increase my awareness of plagiarism. In a more formal sense, plagiarism can be defined as the copying of material from other people or use of material without acknowledgement. This can include removing acknowledgement code from the author, copying another person’s work or not quoting a public source. My desire was to not take ideas from the code I was analysing, but take inspiration instead. As I completed more of my project, I began to understand the importance of plagiarism, from a moral standpoint. The long hours and grueling tasks one must go through, in order to complete a task, should be accredited to whomever has carried out this coding. I find that it can be properly understood once you overcome a problem you were working on for hours and the feeling if someone were to steal and claim credit for it.

From a practical standpoint, if one is to make their code open source, it is vital that the correct methods of acknowledgement is used. If someone is to come across my code and wish to further explore an idea, the correct acknowledgements should be implemented so that they can then go onto learn more about the code. This also applied to any research that I have used in my work, wherein I must ensure to provide correct citations. In the instance of teamwork or group work, it is also vital to include the relevant acknowledgements so that colleagues are aware of where the ideas have come from.

In terms of a more academic standpoint, any plagiarism found in my project can result in an automatic failure, which future reinforces the idea that plagiarism is a very serious issue. Depending on the severity of the plagiarism, some institutions such as Royal Holloway, may find it appropriate to remove myself from the course. This would have massive effects, not only in my personal life, but also my professional one. The fact that I was expelled from my degree would forever be a stain on my career, as all future employers would be made aware of this fact. No one could blame an employer who would be resistant to hire someone with such a negative reputation, so early on in their career. A respectable job would definitely be hard to come across, making it near to possible to progress in my career path. To delve into the more legal issues, there are laws in place which prevent people from copying other people's work, whether it be code or writing.

Whilst developing my website, I stumbled upon a JQuery plugin which automatically detected if a specific element was in the current window viewport. I used the source code in my project, whilst providing the correct acknowledgements, to detect this and trigger my animations. In my website, I also used someone's code for 'smooth scrolling', throughout my website, wherein I also included the acknowledgements. For my mobile application I used Android Studio, which comes with a number of 'helper' classes that one can generate, such as log in or setting activities. I used the navigation draw class to generate the basis of my navigation code. Although a lot of the code had to be modified and changed, to suit my purposes and the code almost looked entirely different, I still provided acknowledgements to Android Studio for the class and the basic structure of the navigation. For my GUI, I did not need acknowledgements toward anyone, as I mostly used Java which I already had prior knowledge of.

As mentioned before, throughout my project I was constantly cautious of any instances of plagiarism. I had a strong desire to ensure that I could proudly stand and say that my project was mostly of my own work, excluding any code that I had used and acknowledged. I believe that in the end I was successful with this and made sure to take any measures to avoid any issues. For example, when writing code I would not copy and paste anything, no matter how small the line of code was, I would hard code everything. This further helped to ensure that all ideas were of my own making and that I was capable of creating my interfaces. My drive to ensure no plagiarism meant that I carried out a lot more research on different coding implementations, instead of just looking at one source. I would study the syntax in more depth, from the relevant websites, and make more of an effort to push myself to understand it, so that I can implement my ideas myself. I have definitely gained a sense of what it is to be a IT professional.

## Self-evaluation

## Overall Achievements

Perhaps the most obvious and important achievement in this project is the actual end result, of the interfaces which I created. The grueling process of going back and forth between various designs, to the struggle of implementation, is a rewarding one when I have three interfaces that I can see in front of me. I was successfully able to implement my portfolio website, including all of the required content. Furthermore, I was even able to add some additional animations, which I had not planned for, to add interactivity for my website. My mobile application was also implemented to suit the Android design guidelines, which was my intended purpose. Last but not least, my GUI was implemented and fully working. All three interface achievements are further discussed in the ‘Reflection of Project Progress’ section. Another achievement which I gained from designing my interfaces, was the improvement of my Photoshop skills. As a result of the heavy use of photoshop to plan out my designs, I was able to brush up on many old skills and learn some new ones.

In terms of development environments, Android Studio was an entirely new concept to me, as was much of the syntax. I was completely unfamiliar with the complex methods and classes which I would have to face, in order to implement my application. I would say, in terms of a learning perspective, this would definitely be my biggest achievement. At the end of this project, I can now proudly say that I can develop a mobile application, with effective validation and navigation. To go from knowing absolutely nothing about a topic, to having a decent understanding, is a massive achievement to me.

My human-computer interaction perspective, particularly on principles and conventions, has definitely had a massive change. From the transition of my initial designs and proof of concept programs to final designs and background reading, I have gained a substantial understanding of what is required from interfaces. In my second year at Royal Holloway, I took a human-computer interaction module which allowed me a brief introduction to HCI, mostly completing assignments in groups. However, this project was entirely individual, forcing me to think deeper about my designs, without relying on other people’s opinions or skills.

My background reading allowed me to learn so many things about HCI and the sheer depth and care that must go into even the smallest aspects of a design. For example, I learnt about how the colour black should rarely ever be used in a design. Prior to this research, I had never noticed how most interfaces do not in fact use black for their main body of font but in fact they use a dark grey. Another instance of what I have learnt is about colour theory, which is again something I did not take note of before. Colour is possibly one of the most important aspects of an interfaces and I was able to learn and implement about the use of monochromatic interfaces and accent colours. When I look at my initial designs and final designs, I can see a clear difference and can notice how I have applied HCI principles, such as Nielsen’s Heuristics to my new designs. In general, I gained a massive appreciation for anyone who works in the HCI industry, as I can understand of how a complex but underestimated role it can be. After completing my research and moving onto my final designs on Photoshop, it was almost as if I had a massive checklist of HCI principles which I needed to me. Beforehand, I would mostly just try to create an aesthetic design, without considering the usability as much.

From a more personal standpoint, which is where I am perhaps the most happiest, is the joy and experience I have gained from this project. I would say the achievement here is that I am now able to have a clearer idea of what career I would like to pursue, be it in website design or Android mobile development. Before this project, I was not entirely sure of which career path I wanted to take in design, but having completed the work now I can see that this is what my passion is. My personal motivation for this project was that I would have some clarity on the issue, which I have well and truly gained.

## Reflection of Project Progress

Having discussed my achievements previously and sharing my joy about the project, I can appreciate the aspects of the project of which I am most pleased with. On the other hand, I am not oblivious to the problems that still exist and am aware that there is much room for improvement. For my website, my initial goal was to be able to have a website which I could distribute to future employers. The information was to be displayed in a ‘clean’ interface, which could be easily read. I also wanted to create a sense of interactivity, as a means to attract employers to me as a employee prospect. Whilst I can say that I have achieved this, I do believe that some more complex animations would have gone a long way, in certain sections. I believe that the ‘Projects’ section of my website could be viewed as quite dull and just blocks of text, which is not what I would have wanted. The lack of more animations was mostly due to the lack of skill in the area and little time I had, but still would have been appreciated nonetheless.

In terms of my mobile application, I was thoroughly pleased with my designs I completed, before the implementation. Initially I was worried that because my application was very coding heavy and I was slightly unfamiliar with the syntax, so my designs would not be brought to real life exactly how I had desired it. However, my designs were very precise and were able to be implemented exactly as how I had planned. The main problem I encountered when working on my application was my lack of experience, that resulted in quite a few issues. It became apparent quite quickly that many features which I thought would be easier to implement were in fact not the case at all. For a portion of this project, I encountered many technical issues with my mobile application which caused significant delays in other areas. During my proof of concept program stage, I do not believe that I pushed myself enough to challenging tests. If I had done this, I would have been much more apt at dealing with the syntax and some of the errors that I had encountered. Unfortunately, due to this added time I had to spend on fixing problems, it took away time that I could have spent on developing my GUI.

Of all the applications, I was perhaps the less pleased with my GUI. While I think that my designs on Photoshop followed HCI conventions and I had some good ideas for the interface, it just seemed that I was unable to effectively recreate this by coding. I believe that this was due to both my lack of knowledge, but also my choice of using JavaFX/Swing. I found it increasingly frustrating that I could not create a GUI with the look that I desired. If I were to do this project again, I would definitely research into a difference approach to designing a GUI, as I am sure there would have been a better alternative. But if I can gain anything from this experience, I can say that I was sure by the end that mobile application or website design was definitely something I was more suited and driven towards.

Having analysed all three of my interfaces I can say that, excluding my GUI for which the problem is clear, that they are all a result of a lack of time and knowledge. Perhaps the worst aspect of my project, which I take full responsibility for, is my development process. First and foremost, I do not believe that I pushed myself enough in the proof of concept program stage of the project. Had I chosen to select more challenging programs, I would have learnt more earlier on and therefore been able to implement more complex ideas into my interfaces. As a result of this, I then found that I still had more to learn in the final stages of the project, thus causing a delay in other areas. Next, I can see that my time management definitely needed rethinking. My initial train of thought was that I would dedicate an equal amount of time to each interface but perhaps now I see that this may have not been the correct approach. I have done HTML and CSS coding in the past so naturally I was much quicker with this and it did not take me nearly as long to create this interface as it did for

my mobile application. Even with my GUI, it did not take as long as my mobile application, the problem was just that I was not able to have complete freedom with the developer environment that I chose.

## Results Analysis

From working on this project, I was able to complete three interfaces on entirely different platforms. I was glad I chose to do three different platforms, as it allowed me to understand how HCI principles apply to various situations. It was very interesting to me to see the different things I had to think about when designing each interface and thinking about how users would be using it. My mobile application was perhaps one of the most easiest to design for, as phones do not generally vary in size too much. One of the things that was perhaps my biggest challenge was making sure my interfaces were adaptable to different sizes. In Android Studio, it is extremely easy to make the code adjustable for screen sizes. In my website, Bootstrap allowed me to use columns and grids to resize the elements easily. This was highly effective for me. My GUI probably proved the most difficult to make adjustable, because of the coding behind it.

Another aspect that was striking to me was the different in navigation, between the interfaces. Mobile applications contain many different screens and hidden navigation, as their displays are much smaller. Almost all of the menus are hidden, such as my navigation drawer and popup settings menu. My preferences menu even contained many embedded screens. A GUI has lots of room for navigation, but mainly focused on lots of menus. A website has lots of space for navigation and is usually targeted at filling the entire screen. This is because the larger surface area the navigation covers, the quicker it is for the user to reach it. As a result, it makes the interface much more efficient. In terms of colour schemes, I kept my mobile application very simple. Due to the small display, I found that any use of many colours would overwhelm the user entirely. With the website, I was able to branch out a little more and make more use of accent colours and large coloured backgrounds. As GUIs are more focused on the functionality of the application, the colour scheme is kept simple and monochromatic.

Throughout design and implementation I found that I was a lot more creative when it came to my website. The sheer space available and variety of CSS effects allows one to have a much more artistic outlook. During my mobile application development, I was very focused on making navigation and every single task as easy to possible. Having being a frequent smartphone user, I can relate to how frustrating a badly designed application is. My aim was to ensure that no users who used my application would have this problem. A GUI can be seen as perhaps the least ‘aesthetic’ but most ‘functional’ of the three interfaces.

## Future Goals

For my mobile application, I was pleased with the design but I do feel as though there could have been more features. In the future, I would definitely implement more for the application and minor tweaks here and there. For example, a progress bar would be a good addition for the register screen. Furthermore, more shortcuts could be added for seasoned users. In general, any addition features would be welcome for the application. A mobile application cannot have too many ‘complex’ features, as it always needs to be able to fit on a mobile device and too much clutter will decrease usability. However, a website is generally aimed at a large surface such as a monitor, meaning more room for flexibility. While the application future goals were more focused on features, I believe my website could have more animations and creative input. For example, there could be

animated backgrounds and an interesting scrollbar introduced, so make the website a lot more interactive.

My mobile application had many different screens and sections which the user could navigate to, whereas my website does not. It would be considered somewhat ‘flat’. To combat this, my future goal would be to implement separate screens or more pop ups, perhaps in the projects section. Perhaps each project card could redirect the user to a different page, where more information is available. Having viewed many creative and wonderfully inspiring portfolio web pages, I can see that my website has perhaps the largest potential, as the possibilities are endless. Unlike my mobile application, I am not confined to any ‘conventional’ Android design guidelines. Concerning my GUI, a goal would be to definitely look into some possible plugins or extensions which would aid me in creating a more pleasing design. Whilst browsing the internet, towards the end of my project, I came across something called ‘Looks and Feels’, which can be implemented with JavaFX and Swing. This is perhaps something I could delve into further, in the future. My GUI could also include more features for the note editing, such as font sizes and colours.

## Conclusion

The difference in my HCI understanding from the start of the project to now is astounding and has only increased my passion for the topic. Furthermore, I am confident that this project has made me more ready for the working world and closer to being an IT professional. Studying the professional issues, such as plagiarism, has allowed me to not just do some coding but think about the outside world and real life software. When completing some university assignments, they can be rather closed off and straightforward questions. However, completing a project is another situation entirely where one is exposed more to the outside world and what others may have to offer. This project has also helped me to understand the importance of a strict schedule and design process, one which I would definitely ensure to work harder on in the future. This again, is something which is highly important in a professional IT environment.

All in all, I learned so much from the project that I view it as quite a success. Although there are some obvious improvements to be made and some weaker areas, such as my GUI, I do truly believe that my website and mobile application can lead me somewhere promising. I have thoroughly enjoyed completing this project and have definitely gained more clarity for the direction of my future career. Had I not chosen this project, I would not have this clarity nor greater understanding of my future work area. As I am sure I will proceed with a career in this area and know I still have a lot to learn, this project at Royal Holloway will always be what I consider the starting point of my HCI career.

## Appendices

### Bibliography

- [1] Alan Dix, Janet Finlay, Gregory D. Abowd, and Russell Beale, Human-Computer Interaction (3rd Edition)

- Used when researching Art in Interfaces, had a quote which was inspiring: "HCI is a marriage of art and science"
- Contained a lot of other useful ideas about interfaces

[2] Cooper, About Face 3: The Essentials of Interaction Design, Third Edition

- Used when researching Art in Interfaces, presented some interesting ideas about characteristics of elements

[3] HCI Lecture Slides 7, Royal Holloway University of London

- From taking a HCI module last academic year, many of the lecture slides came in useful when researching Nielsen's Heuristics

[4] [https://en.wikipedia.org/wiki/Heuristic\\_evaluation](https://en.wikipedia.org/wiki/Heuristic_evaluation)

- Deeper understanding into the concept of Heuristics

[5] Rob Toledo Article, <https://usabilitygeek.com/colour-theory-introduction/>

- A lot of interesting ideas about emotions being representative in colour systems
- Background information about colour schemes

[6] Jake Rocheleau Article, <http://www.vandelaydesign.com/ultimate-guide-color-theory-designers/>

- An argument against Toledo, wherein he suggests to not rely on colour
- More in depth information about the importance of colour saturation and value

[7] Ian Storm Taylor Article, <http://ianstormtaylor.com/design-tip-never-use-black/>

- Ideas about not using pure black in interfaces

[8] Wayne Thiebaud Paintings

- Examples of how pure black is not present in everyday life

[9] Tubik Studio Article, <https://uxplanet.org/mobile-typography-8-steps-toward-powerful-ui-deaf205274c5>

- Interesting conclusions of typography considerations/characteristics

[10] Chris Bowler Article, <https://www.invisionapp.com/blog/4-tips-on-typography-in-ui-design/>

- Views on readability in typography

[11] Android Studio Tutorials, <https://developer.android.com/training/index.html>

- Tutorials to help with setting up Android Studio

[12] TutorialsPoint

- Help with Android Studio Syntax

[13] <http://www.vandelaydesign.com/ultimate-guide-color-theory-designers/>

- Background reading on colour theory

## Project Diary

18 Jan - 24 Jan

The first week of my project, my main focus for now should be reading up on heuristics and carrying out some case studies. I think I need to definitely look at some case studies for

GUIs and mobile applications to get a better idea of what will be successful. From my mobile application proof of concept I know that I did not put enough research into Android Studio and the available features. This will need research.

#### 25 Jan - 31 Jan

Carried out some research on Android Studios and the available content. Found many interesting youtube videos which will help me in my development. Started to redesign and finish some of my final designs. For my website, I seem to have a clearer idea of what I want. I need to really focus on my GUI designs as I have been spending too much time on my mobile application designs

#### 1 Feb - 7 Feb

Finalised many of my designs, in particular I am happy with my mobile application designs. I still think my GUI designs need more work.. but I don't want to get behind on my actual coding and development.

#### 8 Feb - 14 Feb

Started working on my website further and managed to get a lot done. Having some issues with spacing, I need to read up on columns and spacing that bootstrap has to offer. Also, I'm having trouble with working with css animations - I also need to read up on this.

#### 15 Feb - 21 Feb

Working on my mobile application, managed to get a lot of basic functionality done. Login screens and register, etc, were easy to complete. My main problem at the moment is some of the more complex classes with Android Studio. In particular, I am finding it difficult to work the navigation drawer and preferences. Online there are so many different ways to do it that I find it quite confusing and do not think this problem will be sorted any time soon. The preferences basic layout is done but I'm having problems with accessing and setting the summaries of some of the fields...

#### 22 Feb - 28 Feb

Managed to find a fix for setting the preference fields by making a global class which accesses the application. However, I am still having issues with the navigation drawer, which I have been trying to fix most of this week. I need to focus on my website and GUI, which I am behind on development for.

#### 29 Feb - 4 Mar

Did more work on my website and got most of the content entirely done. However, I am still having trouble with the animations. I took most of the week to study them, as it was a lot more complex than I first anticipated. I managed to do a basic animation for some icons, but I am having trouble with some JQuery functions. I can't seem to figure out how to check if an element is in the viewport, so I can start some of the animations. I need to read up on some peoples theories for this.

#### 5 Mar - 11 Mar

Got a lot of work done this week. Found a solution to my previous problem for my animations, using many ideas from the internet. Animations are now all successfully working. I've also been working a lot on my mobile application, as there are still a few annoying bugs. Mostly, I really need to start my GUI development.

#### 12 Mar - 18 Mar

Working on developing my GUI. The programming is coming easy to me but I am finding it hard to create a nice design. Even from looking at youtube videos, many of the interfaces are not

desirable and ‘clean’ as you can achieve from websites and mobile applications. A bit disappointed with my progress for my GUI, but will work on this next week also.

19 Mar - 25 Mar

This week is the first week of the project and my main focus is on my report. I need to focus on my professional issues section and software engineering process. I have previously looked at extreme programming but need to do some more reading about it. I also need to figure out the best way of submitting my mobile application and work on making some videos of my interfaces. My mobile application in particular needs refactoring.

## User Manuals

### Mobile Application

Video: [https://www.youtube.com/watch?v=Pvi\\_xb5NrT8&feature=youtu.be](https://www.youtube.com/watch?v=Pvi_xb5NrT8&feature=youtu.be)

#### Installation

To use my mobile application, simply unzip the file and import it into Android Studios.

#### How to use

To use the application, simply run it on an emulator which is provided in Android Studio. Or, you can load it onto your google smartphone. Links to the Android Studio tutorials can be found here <https://developer.android.com/training/basics/firstapp/running-app.html>

Once the application is loaded, you can either login or register. If registering, a name and email is necessary, otherwise you will not be allowed to progress. Furthermore, the password constraints specified on screen must be followed. After registration, or login, you will be taken to the main page. From here you can create a new task, change settings, or access the drop down menu. Some features that are not implemented, because the project is of a design nature, are still shown on the app. However, when they are selected, a dialog will popup to explain the situation.

### GUI

Video: <https://www.youtube.com/watch?v=TuMQ8qR4cQQ&feature=youtu.be>

#### Installation

To run this GUI, simply double click on the runnable JAR file

#### How to use

It is a similar concept to the mobile application, wherein you can either login or register and must fill in the required fields. Again, some features are not implemented. Premade notes are created, for the purpose of demonstration, which can be read by selecting them on the menu.

### Website

Video: [https://www.youtube.com/watch?v=TxpTPaDZ\\_PO&feature=youtu.be](https://www.youtube.com/watch?v=TxpTPaDZ_PO&feature=youtu.be)

#### Installation

To run this website, simply unzip all of the files and open pocwebsite.html in a web browser, preferably Google Chrome.

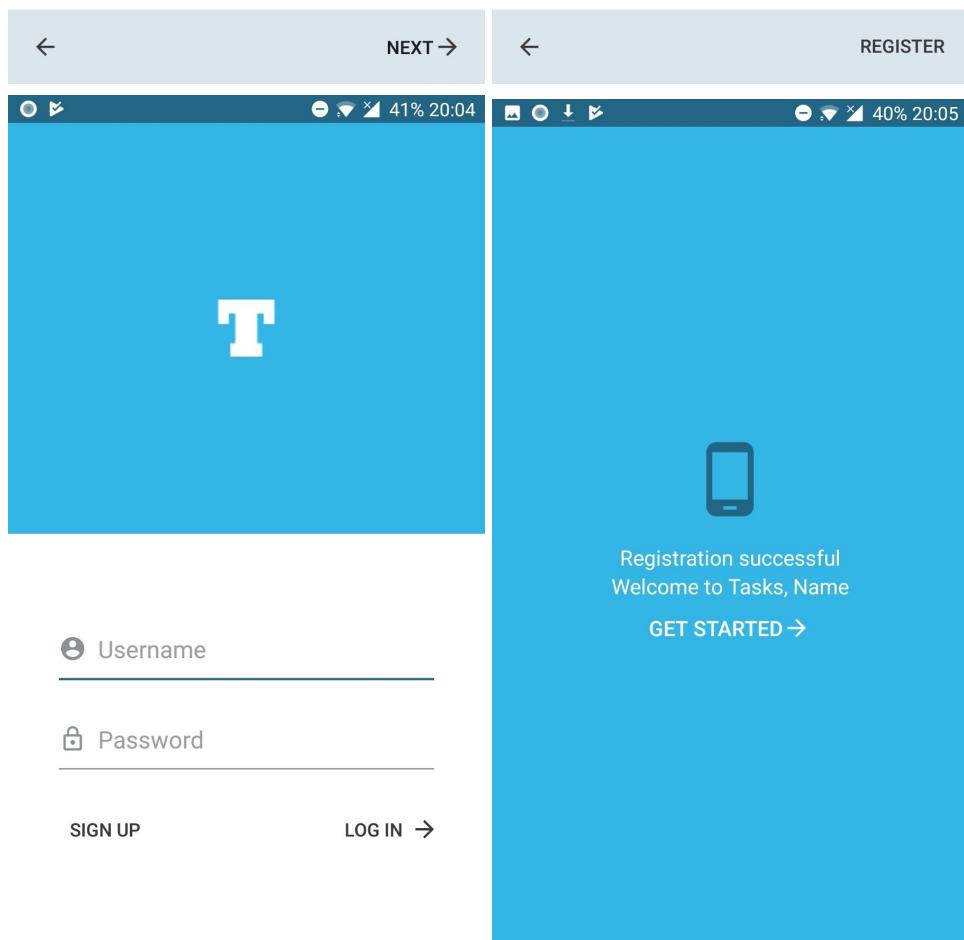
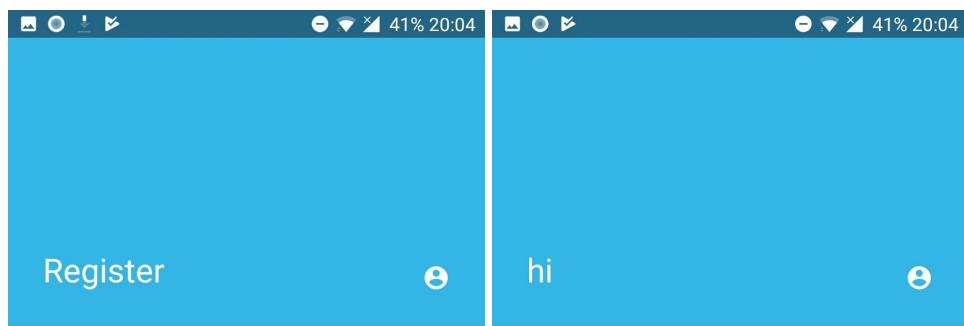
How to use

The website is quite straightforward, as it is purely visual.

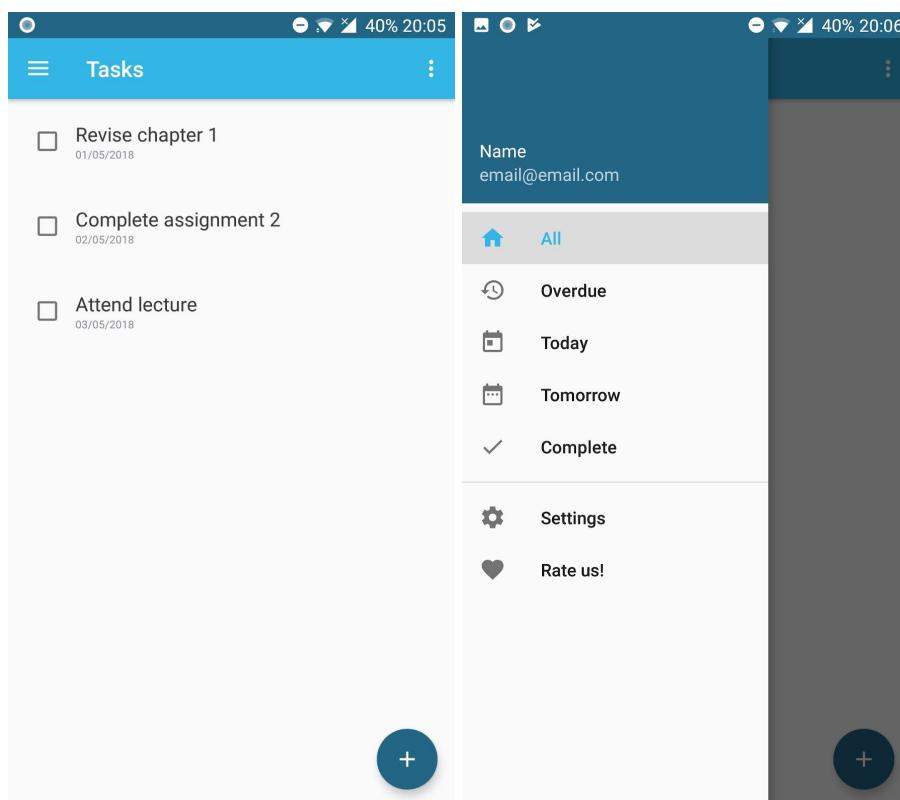
## Screenshots of working programs

### Mobile Application

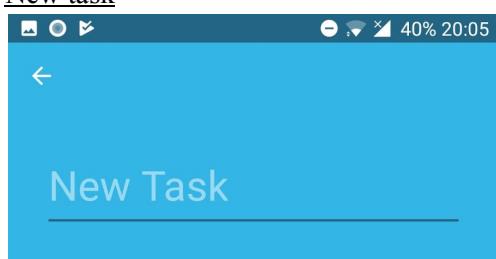
Login and register



Main page



New task



**Due date** Select a date

**Priority** Select a priority

**Tag**

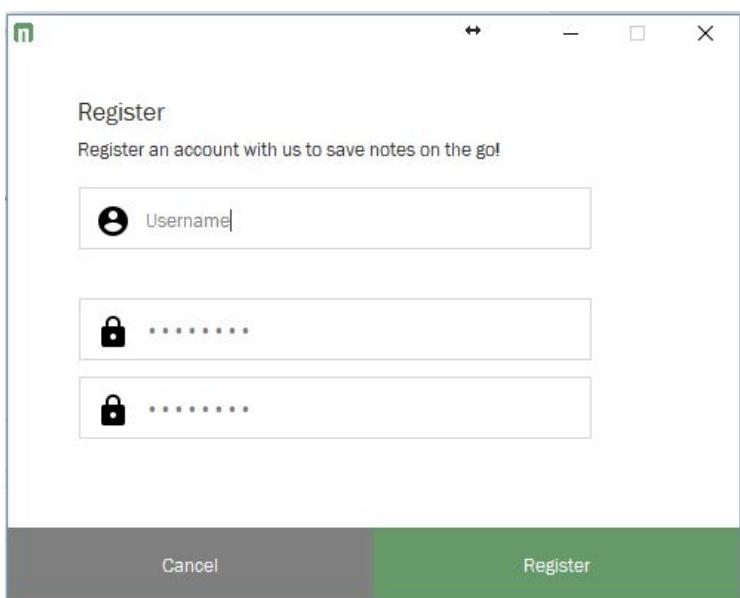
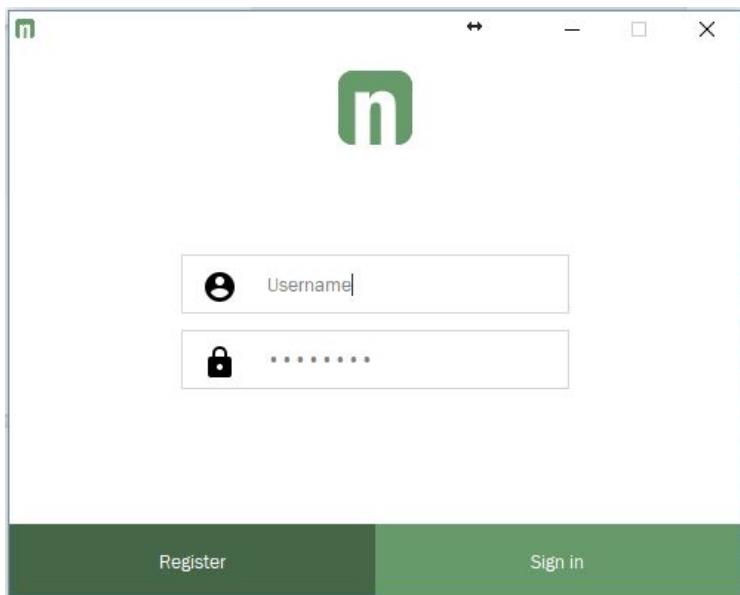


Settings

The image shows two screenshots of a mobile application interface. The top-left screenshot displays the 'Settings' screen with the following options: Account, Notifications, Help (with a sub-note 'Visit our help center'), About, and Log out. The top-right screenshot displays the 'About' screen with the following information: Social (Follow us on Google+), Send feedback (Help us improve our app!), Legal (Task mobile application created by Manjyot Sandhu, student at Royal Holloway 2018, for a final year project), Version (1.0), Private Policy, and Terms and Conditions.

The bottom-left screenshot displays the 'Account' screen with sections for Personal (Display name: Name, Email: email@email.com), Password, and Emails. Under Emails, there is an 'Email updates' section stating 'You're not signed up for email updates from us.' with a toggle switch. Below it is a 'Subscribed emails' section with the note 'Manage your email subscriptions'. A large button at the bottom says 'Email subscription OFF'. The bottom-right screenshot displays the 'Help' screen, which contains a note: 'In the case where the application is implemented by a company in a real life scenario, this page would navigate the user to their relevant FAQ web page.'

## GUI

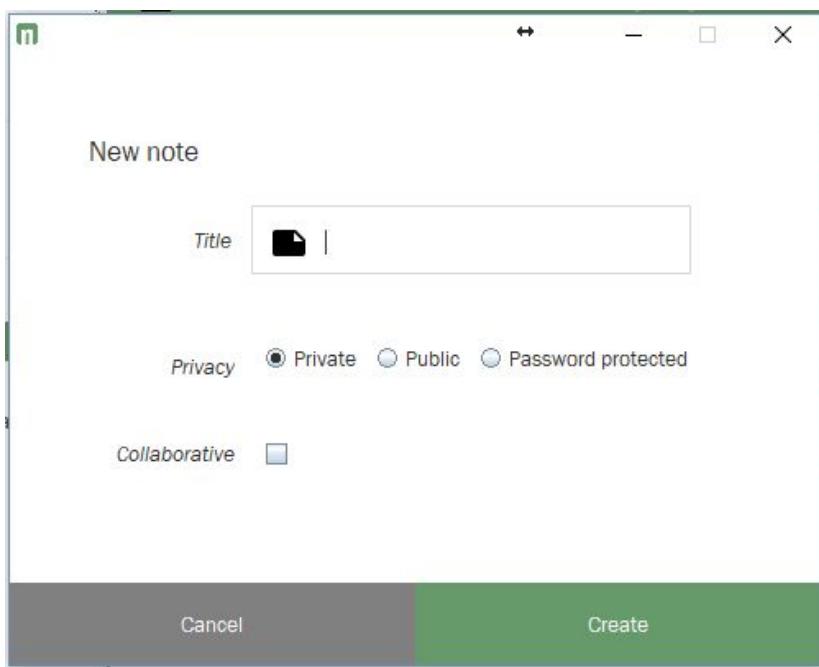


The screenshot shows a desktop application window titled 'Notes'. The main content area displays a note with the title 'Art in Interfaces'. The note's content discusses the importance of visual design principles like color and size in user interfaces. Below the note, there is a sidebar with various options: 'New', 'Save', 'All notes', and a list of recent notes ('Art in Interfaces', 'Colour Theory', 'Personal Motivation'). At the bottom of the sidebar is a 'LOG OUT' button.

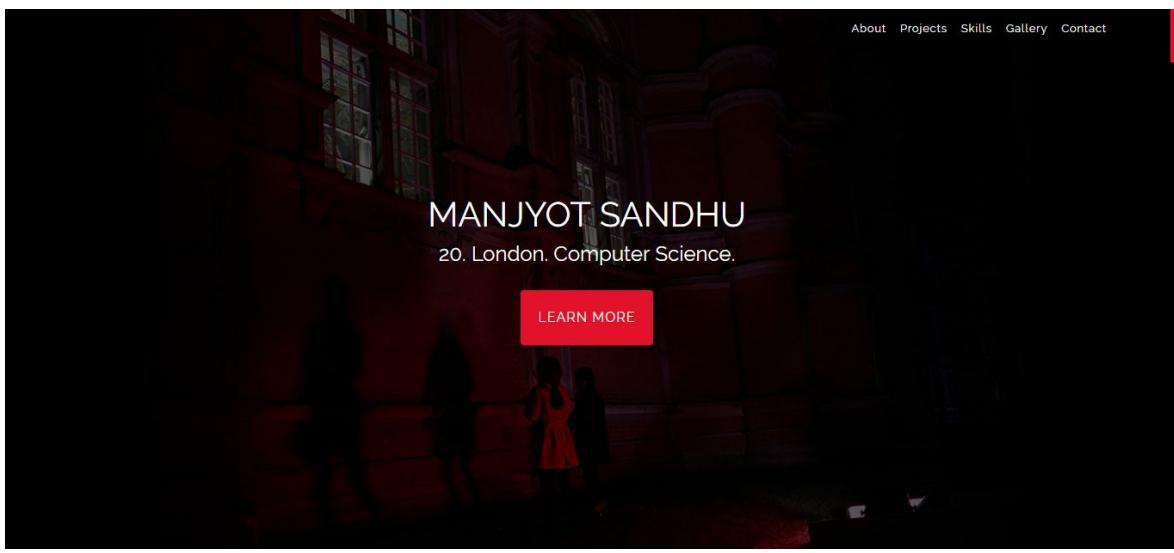
**Art in Interfaces**

As years have passed with using interfaces, certain 'ideals' have been developed, such as a line shape can be interpreted as an area for user input. It is also vital in any design to adhere to these basic standards. The size of an element can help to help the user interpret how important an element is, just from a quick glance. When a user is observing various elements, they will automatically compare the sizes and likelihood is their eye will be drawn to the largest element first. In a body of text, the notions of headings and subtitles are used to represent the importance and a certain type of hierarchy of information. In many modern interfaces, on a login page, the 'Register' button is usually the largest, as a means to encourage more users to click on the button and therefore register with the enterprise.

Colour can have a massive impact on an interface, as it is very easily distinguishable for the human eye, even for children. For this reason, it is so often utilised in learning environments. Various shades of colours and hues can be used to, as Cooper [2] referred to it as, a 'visual system', to help the user distinguish between similarities and differences. Certain colours have certain connotations attached to them, such as red has the meaning of negative, which is why it is often used to represent errors or cancel buttons. On the other hand, we can argue that we have this idea in our head because of day to day idea, such as stop signs or traffic lights. In other parts of the world, they may not have these ideals and therefore colour is not a good idea to rely on, to convey a message. In addition to this, colour blindness can mean that certain users are not able to get the full use out of a system. The orientation of an interface can help to provide the idea of a flow or direction for the user. Providing that a user is completing a form in a mobile phone application, in a portrait orientation, the user will naturally complete the form going from the top to bottom. Using icons which are pointing in a left or right direction can also help to direct the user, such as a 'Next' button. Even small aspects of a system, such as a scroll bar can help give the user a sense of direction.



## Website



## ABOUT

*Timeline*

Born  
3rd July, 1997



Primary School  
Attended Castleview School



Further interests and awards



#### Duke of Edinburgh

In progress of completing the Duke of Edinburgh awards in Silver.

[Find out more](#)



#### Maths Challenges

Achieved Silver in both Junior and Senior math challenges

[Learn more](#)



#### Interests

My other interests include piano, art and photography.

[Visit my VSCO](#)

## PROJECTS

Personal work and University projects

#### Photoshop design

Self taught photoshop for personal use. Use for designing typography, posters and various photo manipulations.

[Photoshop](#)

#### Personal photography

During the course of my art projects at A Level, I did a lot of photography and photo manipulation.

[Personal Photography](#)

#### University Group Project

In my second year, I took part in a large group project wherein I learnt how to use and implement GitHub with my work. I was involved in the design and coding of an interface for a restaurant, which was supported by a backend database.

Teamwork, JavaFX, Java, GitHub

#### JavaFX GUI

As part of my final year project, I designed and created a JavaFX GUI that serves as a notes application.

#### Android Mobile Application

I designed and created a task mobile application, as part of my final year project. I carried this out using Android Studio.

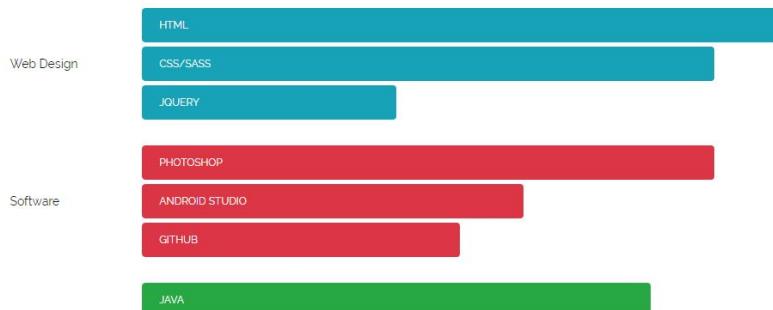
[Java, Android Studio](#)

#### Personal Website

This website was created by myself, as part of my final year project. I used Bootstrap libraries to help me with this.

## SKILLS

Languages and software engineering techniques



## GALLERY

*Photographs from around the world*



## CONTACT

*Send me a message*

Your name

Your email

Your message

Maximum 250 characters.

SEND MESSAGE



© Manjyot Sandhu 2018