

Final Year Project Report

Full Unit – Interim Report

An HCI Approach in Designing Usable Interfaces

Jatinder Sohal

A report submitted in part fulfilment of the degree of

BSc (Hons) in Computer Science

Supervisor: Dr Giorgios Koutsoukos



Department of Computer Science
Royal Holloway, University of London

December 08, 2023

Table of Contents

1 Introduction	4
1.1 Brief overview	4
1.2 Necessity of HCI	4
1.3 Project Specification	5
1.4 Aims and Goals/Objectives.....	6
1.4.1 Specific Aims	7
1.4.2 Goals to support aims.....	7
1.5 Survey of literature.....	7
2 HCI issues	12
2.1 User experience (& UCD).....	12
2.2 Colour theory	12
2.3 Accessibility	13
3 Designing interfaces (summary of work)	14
3.1 Sketches/Prototyping	14
3.2 Using HCI in design.....	15
3.2.1 Video of interface	15
3.2.2 Financial tracker website	15
3.2.3 Project Planning application	16
3.3 Current and future status.....	17
4 Software Engineering.....	18
4.1 Git (version control)	18
4.2 OOP.....	18
4.3 Testing.....	18
4.4 Technologies	19
4.5 Waterfall	19
5 Planning and time scale.....	20
5.1 Gantt chart.....	20

5.2 Diary 20

6 Bibliography 21

Appendix - Diary (Not included in word count) 23

1 Introduction

1.1 Brief overview

In this report I will be discussing my project, an exploration in designing and implementing usable interfaces using HCI principles. This report hopes to serve as a guide to my current progress and insights about HCI - a very complex and diverse subject. The project is currently halfway through and has varied significantly from my original plan from changing technologies, interface ideas and even including different processes before the development which will all be discussed.

I start the report with an in-depth discussion on the necessity of HCI, highlighting its importance in current technologies and what ignoring it in the pursuit for more functionality, can cause. Next, I discuss my specific aims and objectives for the project and how I wish to build on insights I have learned so far and what I wish to accomplish from this project. I then will discuss relevant literature, and existing research in the field. This will help to explain the context behind decisions in my project and how I was able to get to this stage. This will be followed by a quick summary of milestones that I have completed and that had to be changed.

Chapter 2 will discuss multiple HCI issues which I have researched on and have tried to implement in my own user interfaces. I will talk about the impact in not only software but also hardware as this field affects all aspects of technology. Moving on to chapter 3 I will show the work I have completed, the processes to get there and the different HCI principles that have been implemented. This section is supported in a video which can be found in the zip file. I will then explain the current status of the project and what steps I will be taking in the future.

For chapter 4 I will explain the different software engineering processes involved in generating the user interfaces, how I used them and why they are important. Followed by a timeline for the next term, a bibliography, and my diary in the appendix.

Overall, this report aims to serve as a reflective study and plan on how to develop usable interfaces using HCI, the challenges and learning that I encountered on my journey and what I wish to develop in the future using this knowledge.

1.2 Necessity of HCI

HCI is the study of designing, evaluating, and implementing technology focusing on the interaction of the human with the technology, as stated in its name. It is a deeply complex field bringing together computer science, behavioural sciences, and psychology, with the golden aim of improving user experience. It is such a significant field as it directly influences how effectively technology can be integrated with our daily lives. Good design using HCI means enhanced productivity, accessibility and satisfaction which only further the value of technology and what humans can get out of it.

When discussing HCI, there are various aspects that are important to improving the design of an interface such as accessibility, perception and cognition, usability, interaction and many more. But one of the principles that encompasses HCI is user centered design because it ensures the technology is designed for those who want to use it. It is the foundation of good HCI design that involves understanding the user and constantly refining and improving the experience depending on user feedback. This means that interfaces are more likely to meet the needs of users and allow

users to get the full functionality of the interface they are using, essentially bringing the other benefits and aspects of HCI.

The consequences of neglecting user centered design can be very major. If interfaces are difficult to use and poorly designed, it may lead to an increased learning time and while using them, will lead to more mistakes and errors. One example of this was the Hawaii false missile alert (2018) that placed a test alert next to a real one, and lead to an employee erroneously selecting the wrong option, causing widespread panic. This just shows how much of an impact a user interface can have and why HCI is so necessary in today's world where more of society is being place in technologies hands.

1.3 Project Specification

Project goal: To develop a deep understanding of HCI and its wide-ranging principles so that I can create an interface that understands and meets users needs and allows them to experience the full functionality of the interface.

Requirements:

- Design 3 UI's that are designed for different user groups and increase in difficulty
- Every aspect of the interface should be derived from HCI and must be evaluated using its principles and use of both user testing and Nielsen's principles
- Using software engineering principles to develop interfaces using industry level practices

Deliverables:

- 2 concept UI's that show different HCI principles applied and made using different methods and technologies to create them
- Conduct analysis of requirements and prototype sketches of UI design before development
- Create one final design using knowledge of design proccess and learn from any mistakes made with concept UI's
- Final report that sums research, programming over the 2 terms and user manual for last UI

Technologies (will discuss more in se):

- React for Financial Tracker Website
- Python and Tkinter library for Project Planning application
- Creating a recipe Android appication using java and android studio

Target audience:

1. Financial tracker:

- Designed for young students and workers who want to keep track of their finances in a visual manner and need guidance on how to cut down any excess expenses
- Need a streamlined and reactive UI that directs them to the functionality but does not doesn't restrict any actions
- Allow for easy access and editing as will be a lot of input and so there will be mistakes

2. Project planning application:

- Designed for non-technical users or those with little time who want to use a quick tool to keep track of current projects they are working on or their daily life
- A UI that has simple and effective options that provide feedback to their user while still being very functional
- Plan for errors and allow users to recover from them due to non-technical users that may make more mistakes than others

3. Recipe application:

- Needs to combine the streamlined and focused design of the interface with the simplicity and ease of use of second interface
- Must be responsive and adjust elements like typography, border elements, colours etc to any screen size
- Needs support for multiple languages – application dependant on text, so needs to be inclusive towards everyone and have different accessibility options compared to the desktop UIs

1.4 Aims and Goals/Objectives

This section will discuss my overall aims for the whole project and the goals that will accompany my journey to understanding HCI principles. HCI is very broad topic and so this project will focus on the key principles that are required for designing usable and effective interfaces. My primary aim of the project is to understand the balance between functionality and usability, and how to make interfaces clear and easy to use as functionality increases. This is supported by my goal of learning how to consistently create an interface that meets users' needs and delivers a near perfect user experience. I hope to achieve these targets by instilling HCI principles at every step of the design and implementation process and using the design techniques that have been optimised over time.

1.4.1 Specific Aims

- Aim 1: I aim to master the process of user centered design, by understanding user's requirements before development, and constantly refining my implementations with users in mind and the feedback I receive
- Aim 2: To design 3 very different UIs with every component and design choice influenced by HCI principles
- Aim 3: To evaluate the effectiveness of different HCI principles on enhancing user productivity and ease of learning
- Aim 4: To refactor designs of interfaces to ensure accessibility and inclusivity of all possible people. This will be done using several testing tools and user testing with various groups of people
- Aim 5: To stick the design process throughout the project and execute all processes

1.4.2 Goals to support aims

- Goal 1: Before designing an interface, I need to first analyse the requirements of the interface I will be building and then generate prototypes/sketches to act as a foundation which I can build on
- Goal 2: Constantly improve implementation with ideas on how to improve it by conducting user testing. This can include usability testing, hallway testing, interviews and even surveys
- Goal 3: Try to implement a good balance of HCI principles in each interface, focussing on a certain target audience for each interface
- Goal 4: Begin with design of finance tracker, then a project tracking application and finally a recipe application on the android operating system
- Goal 5: Complete post design evaluation with the heuristics, to find simple issues with my interface, but don't over rely only on them
- Goal 6: Use tools like WAVE accessibility to ensure aria labels are not missed out and use tools like AXE accessibility checker and colour contrast analyser to ensure interfaces are able to be used by everyone and meet standards
- Goal 7: Conduct layout and functionality testing after the implementation to ensure the interfaces behave as expected and will perform on users computers

1.5 Survey of literature

In this section I am going to discuss the different books that I read in my research while creating my interfaces. I will use content from the books later in my report, especially in the HCI issues section and so will be explaining the books main chapters underneath.

First, I will go through Alan Dix's book that layouts the theoretic foundations for HCI, Application of principles by Lidwell and then dive into the practical strategies highlighted by Schneiderman.

Dix, Alan, et al. Human-Computer Interaction. 3rd ed., Pearson, 2004

This HCI book is very comprehensive one that describes HCI from the ground up, talking about all aspects including history and hardware HCI. Because of this I decided to mainly focus on part 2 (Design process) of the book which covers a range of topics from interaction techniques to evaluation methods. In this literature survey I am going to summarise the key points from these chapters in this section, what I can apply and what isn't so practical.

The first section of part 2 begins by providing an overview of interaction design. It outlines the importance of the design process, and the role of understanding users' needs and behaviour. It layouts the application of various methods and tools not only in the design but also requirements gathering - crucial to understanding the user. This section then goes on to exploring practical applications of these design techniques in the real world and how concepts can be applied in scenarios.

Chapter 6 delves deeper in this design process expanding on how integrating user centered design inside a software engineering framework and how we can use principles of User Centered Design. It goes into detail about usability and user needs and describes the 2 ways to prototype designs. Chapter 7 takes a look at the different tools that can create this usability discussed before in actual interfaces. It shows a set of guidelines that we can use as practical advice.

The last few chapters of this section cover the translation of the development and evaluation principles to practical application of those. It details tools and systems that allow us to implement interactive systems such as UIMS. It then discusses methods for evaluating interactive system to meet users' expectations and needs. And finally details how to create diverse interactions and why accessibility is important in the design of interface.

This is a very quick summary of the books main chapters and what I got from reading it, but we can tell this book is a bit outdated especially the chapters where it talks about the software engineering. In modern development agile is always in some way included in development but this book sticks to a more structured approach which is understandable as the book is an older one. Technology has also evolved quite a lot since then and you can tell with the technology he describes and images he uses. However even with this, the core content is still very valuable and applicable today and I have used its knowledge alongside other sources to create better interfaces than before.

Lidwell, W., Holden, K., & Bultler, J. 2010. Universal Principles of Design. Rockport Publishers

This book was a great help when designing as it is a guide of 125 principles that enhances usability, improves interactions, and makes interfaces more appealing. Some of the principles are not part of HCI but still are a great help and a practical way of applying this massively broad topic in an easy-to-understand format. I will now discuss some of the HCI principles that were discussed in it and give me opinion on the book.

One principle that I had also seen in the early pages of Alan Dix's book was affordances. Its emphasis that a design should suggest how object can be used, essentially emphasizing its functionality. When its characteristics are known it means the design will be easier to use and more effective. Another principle talked about is Hick's Law which talks about the time to make decisions. The more options provided the longer this will be which means for time critical tasks we

should minimize options. There are incredibly large number of principles like this which all have importance in HCI and would take many words to summarize each one.

In fact, both of these principles are part of the usability section which covers more than 30 principles by itself showing how many principles there are. But this is where the problem lies with this book. It provides such a wide range of information but the chapters are very brief and when there is so many principles like that, it is tough to remember them later when trying to apply them. I found it a good point for being introduced to principles but relied on other content when applying and understanding.

Shneiderman, Ben, et al. Designing for the User Interface: Strategies for Effective Human-Computer Interaction. 6th ed., Pearson, 2017

This is another large HCI book and one that was actually recommended by the author of the first book we talked about, Alan Dix. This book is a great resource and one that's main focus is the practical applications of designing user interface which differs from Alan's book which is more focused on the actual theory of HCI. It has 4 main parts that cover more than 600 pages.

The first part is an introduction to the book, describing how the book will progress and then starts by going into detail about usability, which is one of the most important principles in HCI. It talks about the different goals and measures for usability and why usability is so important providing real life examples that are relevant due to the book being recent. The next chapter in this part is all about accessibility and the differences between humans that interfaces should account for. Ranging from cultural differences, cognitive and perceptual, physical abilities and especially those with disabilities. This is an important early chapter that provides context for a developer of all the different aspects that needed to be accounted for. The last chapter then talks about the different principles and guidelines involved in design and how to choose an interaction style. This is highly valuable part of the book that provides golden information about HCI and its application.

The next part then talks about the actual design process, talking about the different frameworks, method, practices and issues involved in the actual design. It talks about key phases similar to [1] starting with requirements analysis all the way to evaluation. It talks about UCD directly and even discuss agile which is something that wasn't around when [1] was made. The next chapter then talks about the user experience and the different techniques in testing to make sure it is at a good standard. It covers methods like heuristic evaluation (which I used), usability testing and user surveys. The final chapter then talks about different case studies discussing real like successful designs. Talking about the decision-making process, challenges faced, and lessons learned.

The next chapters were not too relevant to my project and so I only covered them briefly with a few small sections that apply. The last few chapters of the book talk about design issues discussing issues like colour, view management, animation, web pages and many more topics. It provides at least 2 pages of information about each topic giving key facts and applicable information that any designer could use in their own projects. It then moves onto help and documentation and the differences between the real world and screen. The final chapter is not that applicable to my project and talks about data visualization, showing various graphs and the different design choices involved with them.

This book is a great help for HCI students as it tells us how to practically apply the theory that has already been described thoroughly across multiple other sources but never go into detail of application. It gives you direct recommendations for design and tells you how to do all aspects of HCI including user testing. However, one point I noticed, was that there was not much information about the current technology like AR, VR, wearable watches and even though phones are

mentioned quite a lot, not in the same detail as web interfaces. This may be the case due to the book being made in 2017 where some of the technology like AR was in early stages, but it would be good for the publisher to release an update version to include some of these technologies especially as the book targets direct application more than general theory.

Nielsen's Heuristics

There are many types of usability heuristics to evaluate interfaces with, but the most popular and the one that I will reference a lot will be Nielsen's one. The original list was made in 1990 with Rolf Molich and then refined by Jakob Nielsen in 1994 [8]. Nielsen's heuristics are 10 principles that have been adopted by designers and researchers over the years and was based on Nielsen's research and observation of real users and problems found during their usage of interface. These heuristics have been applied to a wide range of products and applications and are used by popular companies all over the world.

Here are a three of the heuristics out of ten and a description of them based on my understanding:

1. Visibility of system status

The design of the interface should keep users informed about the interaction at every point. This should be done through feedback like visual or audio prompts and should be within a reasonable time of the interaction. This allows users to make informed decisions about how to use the interface and learn the outcome of their last interactions with the interface.

2. Consistency and standards

The interaction should remain the same from the start of the interaction to the end and words or actions should be the same throughout. The interfaces should follow platform conventions and industry rules so that users are not totally unfamiliar with the interface.

3. Error prevention

Interface design should prevent problems from occurring and remove conditions that could lead to future problems. Users should also be provided with confirmation options and error messages before users commit to any actions.

The rest of the heuristics can be read on this provided source [5].

When using the heuristics in the design and evaluation of my own interfaces I sometimes used them as crutch and almost used it like a checklist making sure that I covered all basis of the heuristics. But this is a bad approach than actually identifying the key issues with my interface and instead trying to fix everything that I have not included in this checklist. I only realised this after using other techniques like user testing that pointed out glaring issues with interface that I ignored as I had covered all 10 principles listed already. But these heuristics are still very valuable and allow designers to create effective solutions to problems with their design. They just need to be included in the whole design process and not be the only tool used.

Gestalt principles

Gestalt principles are a set of principles that describe how the mind perceives and make us visually group elements together [8]. They help us understand how patterns and relationships work and how

the mind prefers to perceive stuff more predictable and simply. The main principles were briefly discussed in [8] and these principles are good because they provide these simple descriptions like the one you can see on the right. This allows designers to use present elements in certain ways and gain an insight into human perception and cognition when implementing interface.

However, these principles ignore the fact that there are individual differences between everyone and context matters. Humans have varying level of acuity, perception of colour and different ways of recognition (due to cultural and environmental reasons), and so these principles are not a one size fit all. Another aspect that I have experienced is that it is simple to understand examples provided but when trying to apply these principles to my own interfaces I found that I was trying to fit the examples provided instead of experimenting with my own arrangements of shapes in the interface. But overall, these principles still provide a simple foundation for understanding how the human mind works and allow us to see real examples where the mind ignores or oversimplifies what it is looking at – showing what bad design can look like.

2 HCI issues

2.1 User experience (& UCD)

In HCI user experience is a central topic that ensures users can efficiently use interfaces to their full functionality and be satisfied and engaged while using it. An interface that is easy to use, fulfils user needs and is enjoyable, means users are more likely to use it and return, which in certain contexts is significant to build brand trust and be competitive with other interfaces. The more functional an interface is, the clearer and easier to use it has to be for users to be able to access it [11].

User experience has evolved quite a lot over the years and initially was based just on the usability of interfaces. However, it has developed into quite a broad topic as technology becomes more integrated into daily life and so involves not only functional aspects but to also create a meaningful and engaging experience to appeal more to users. Ranging from adapting to users' preferences, to using colour to evoke certain emotions and making sure interfaces are usable by people with all abilities and backgrounds, the user experience has advanced massively with progress of technology like AI and machine learning [2]. We will touch on some of these aspects in the 2 sections after, but how we do fundamentally create interfaces that incorporate all of these topics.

User Centered Design is a design philosophy in HCI that allows a design to tie all these unique aspects into one interface. It involves understanding who the users are, their use cases and how they will be using the interface. This is done using an iterative approach having repeated design, testing and evaluation cycles at every stage of the design process and implementation. Because it takes users feedback in at every iteration it allows for continuous changes and improvement and so ensures that users are happy with the interface being designed. This allows developers to implement functionality knowing the interface is still appealing from a user perspective. It is considered a corner stone of HCI as it brings all these benefits to interfaces like usability, efficiency and ease of use.

2.2 Colour theory

When designing interfaces using HCI, colour theory is an essential tool for several reasons and especially in improving user experience. In this section I will detail some of the main reasons for its importance, and how to utilize the application of colour in an interface.

In HCI colour is not only a visual element but can immensely influence user perception and emotion [1] allowing designers to shape interaction. Different selections of colours can evoke natural, often instinctive responses that can be traced directly to nature and cultural associations. Because of these cultural and personal experiences there is no definitive universal meaning to each colour, but the effect of primary colours is clear to see and have recognised associations and impacts.

Everywhere in industry we see these primary colours being adapted and used to influence user decisions. For example blue is used frequently in social media (Twitter, Facebook) to promote feelings of serenity and calmness, red to stimulate appetite in fast food industry (images on the right) or yellow in e-commerce to evoke a sense of happiness and optimism. There are endless examples but it just shows how much of a crucial component it can be and how it can majorly influence the



interaction and engagement of an interface.

Colour can also be used to help an interface appear more organised and less cluttered allowing users process information faster, essentially reducing the cognitive load and improving their experience. The visual distinction in different elements allows a user to quickly understand the layout of the application and what to prioritize when using the information. For example, users will be more drawn to warm colours so we can use those for urgent or tasks that need to be done first and use cooler colours for a background. Another key point to remember is that using colours consistently through a application will also help users process pages and interactions faster, allowing them to be more effective using the interface.

2.3 Accessibility

Accessibility is another key principle in HCI and is directly tied with the user experience of an interface. It is the process of designing interfaces that are usable by people no matter their disabilities or ability to use technology. It ensures that everyone can fairly use technology and is also a legal standard in most countries.

There are four disability areas in interface accessibility and one of these is visual accessibility which is accommodation of those with visual impairment e.g., by ensuring screen readers work and allowing resizable elements etc. But one area that is overlooked is the role of colour with this impairment. We focused on colour in the last section but never went into detail how those with an impairment can be impacted. If we use colour alone to show information e.g. using red to indicate errors, this puts colour blind users at a significant disadvantage leaving them unable to interact with the interface. But colour can also help those with this impairment like using colour schemes using high contrast like on the right making text more legible and overall improving the user experience.



One particularly important principle is accessibility is universal design [12]. This describes designing interfaces that are as accessible to as many people as possible, essentially ensuring that the interface covers all 4 disability aspects. It's about designing products that are usable by everyone, so that they can access all possible functionality without the need for any special adaptations. But this design principle benefits those without disabilities as well, leading to a design that is more flexible and user friendly.

3 Designing interfaces (summary of work)

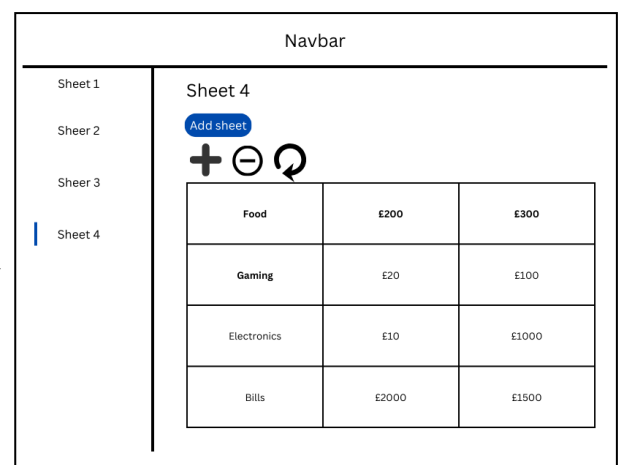
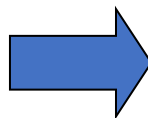
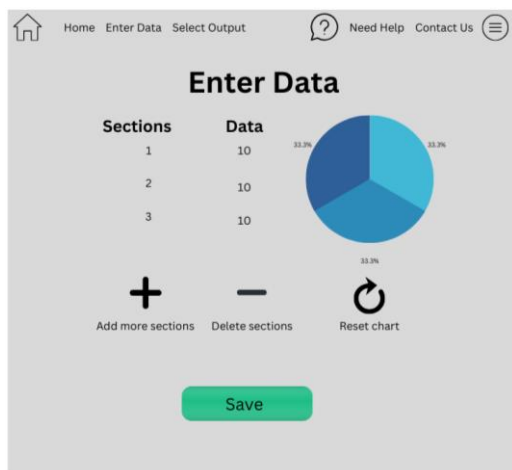
I have created 2 interfaces up to this point in the project, underneath I will go into detail about how I developed them both, how I used HCI principles in the design them and how I will progress from this point to the final submission.

3.1 Sketches/Prototyping

Before designing any of interfaces, the first step was for me to analyse the requirements of my interfaces and then create an early prototype as advised in Alan Dix's book. This prototyping would allow me to rule out assumptions in the requirements analysis so that my implementation covers all aspects of the interface.

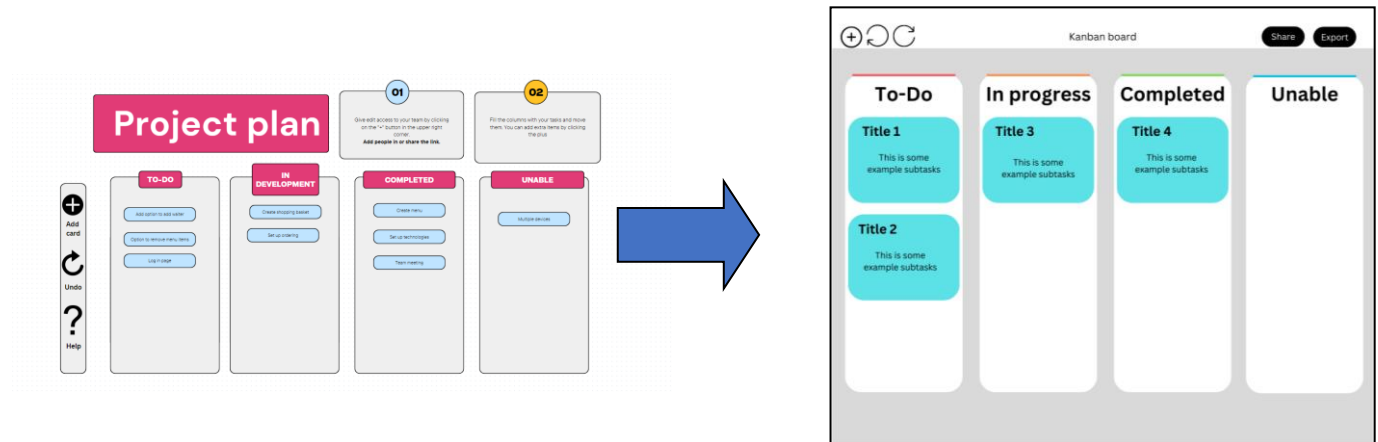
The first user interface I created was a financial tracker which is what I created when I was in the early stages of my research. At this point I had little idea about how prototyping works and what a good UI design would look like. On the right you can see my first 2 sketches for this interface, one for the main page and one for the save data page. These were the 2 pages I wanted to get right as the other pages of my interface would follow a similar design and wouldn't take so long to plan for.

When implementing the first page, I stayed on course with my prototype and create an interface that is quite similar to it. I made changes to the colour scheme and how the navbar is organized but these were changes that I thought fit the interface after the main design and ones that used colour theory more effectively. However, when creating the second page for my interface which you can see underneath, I received a lot of criticism when doing user testing and so because of this I created another prototype which you can also see next my first attempt. Here I made a mistake of not getting enough user feedback, as when I used hallway testing instead of an actual feedback process with organised questions.



For my second UI I created a project planning application, instead of leaving this for my last interface. I created the prototype, got some feedback and then start building the interface. However, when using JavaFX and Tkinter, I realized that the design wasn't appropriate and wouldn't be easy to implement to a good standard with HCI. Because of this I refactored my design into another sketch and then built it using this which you can see underneath. This mistake happened not because I didn't get enough user feedback, but because I didn't structure it well enough. I used testing similar to hallway testing and not a questionnaire type that resulted in my misinterpreting

what people were telling me about the design. And so wasted time that could've been used for more features and development, by creating a new prototype which you can see on the right and start development from scratch again.



3.2 Using HCI in design

3.2.1 Video of interface

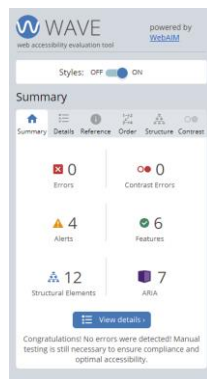
<https://www.youtube.com/watch?v=znipOFbwW70>

3.2.2 Financial tracker website

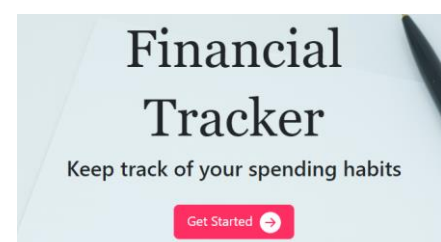
We have gone through the functionality of my interface above, in the next 2 sections I will describe how these interactions and designs were inspired by HCI and will be evaluating them using Nielsen's heuristics.

The financial tracker was designed for young working-class users who need help to see their finances in a visual manner. Because of this the website was designed to be aesthetically pleasing and flexible, so that users that are proficient with technology can effectively use it but at the same time still doesn't exclude anyone from using it.

We can see from the video that I placed a big emphasis on the accessibility in the design interface. The settings for accessibility features are available on all pages in a common place users would check. I provided a range of features for those with different visual impairments and made sure that all my elements can be read by screen readers using tools like WAVE accessibility, which can be seen on the right. The website can also be navigated using only a keyboard, removing the need for precise control.

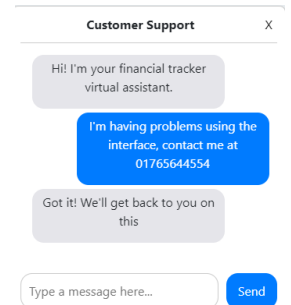


This interface was designed using user centered design, and so was designed in a simple and understandable manner for users and attempts to reduce unnecessary cognitive load. One way this was done by removing unnecessary tasks and ensuring that the interface has a clear visual hierarchy that directs the user. I implemented this differently on each page, using typography on the home page (image on right), colour on enter data and alignment on the help page to help direct users to the functionality. Also, by using the Bootstrap library it ensured the website was aesthetic and responsive which supplemented the usability of the system.



One Nielsen heuristic I tried to implement was consistency across different pages of my website. This was important in an application with multiple pages and different parts so that the interaction remains the same. So, in this website I used consistent colour, button systems, fonts and the same navbar across multiple pages so that it allows the user to become familiar with my website and know what to expect. This is further explored in the way interaction like clicking to enter data or viewing different pages happens, it all follows common conventions that allows a user to quickly adapt and access functionality easily.

One other heuristic that I spent a lot of time on was the error prevention and recovery. This is a very important aspect of an interface like this, as the user will be entering a lot of inputs and so they should be able to recover from errors easily and quickly. Some of the elements I added for this are only allowing certain values like numbers, no empty searches or additions and easy changing of any value in a spreadsheet at any time. There is also a help button in the top right of every page allowing easy access to help like FAQ's and a live chat.



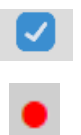
I also placed a big emphasis on visibility of system status which helps keep users informed about the past interactions with the interface and what they can expect from their next interaction. I implemented this in multiple ways for example, current section of navbar lights up with current page, sheet lighting to show which is being edited, highlighting buttons (can be seen on the right) and providing checkbox for accessibility settings. This allows for a clear experience for users avoiding them being confused about certain aspects of the interface. Some of these also have other effects such as reducing cognitive load and improving user experience.



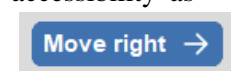
3.2.3 Project Planning application

The project planning application was designed with non-technical users in mind. And so, the interface was designed with an even bigger focus on the feedback about the interaction. This central design focus helped me to bring out the usability of the interface and help balance it against this more complex and functional interface, compared to the last. This clarity is what allows non-technical users to be able to use the interface and why I placed such a big focus on it.

The main content of this application is the lists which contain cards that allow users to keep track of tasks using individual subtasks to map out its progression. I used bold text and different colour border on top of the lists, to show the status of tasks. This was inspired by the visibility of system status principle which was also used when I added checkboxes to the tasks and a progress bar, to present the status of the cards clearly and visually. Just like how I explained for the last interface, implementing this principle provides feedback to the user and is what allows them to understand the interaction.



The cards have buttons to move between the lists which I added to allow the user control so that they can navigate through the different states of the interface. Some popular project management boards like Trello use drag and drop instead of buttons however analysing from a HCI standpoint, I thought it was more fitting to use buttons for this functionality. This is because it provides a clear action that has a single well-defined outcome whereas drag and drop can lead to confusion and even errors with accidental repositioning. Drag and drop also doesn't consider accessibility as buttons allow users who cannot have precise mouse control to still be able to use the application with a button.



In this interface I tried to use colour carefully to convey status of tasks. For example, I used grey on top of the To-do column to indicate that it hasn't started, and then used orange and green for the next 2, orange for in the middle like a traffic light and green for finished, which has the association

of go. I used a creamy white colour for the background and navbar to improve the contrast between it and buttons on the cards and navbars. I used orange on the bin logo to indicate a warning, that this has significant impact if pressed. I then used neutral colours for the other buttons that look visually appealing but are found together in nature.

I have implemented a number of other features using HCI which I will briefly go through: Used tooltips to prevent errors and act as a source of help, Undo and redo to help users recover from any errors, consistent card layouts and lists with buttons doing what you expect them to do and have designed the interface to reduce cognitive load which makes it easier to understand what to do (Recognition rather than recall). The add new card loads a new window that has a simple layout with a hierarchy that directs users to completing it. I have also provided a search bar and options to export the plan providing the user with freedom instead of restricted options. The help button was also added to provide the user with help and documentation so users understand how to use the board and can look for direction there instead of adding visual clutter for other users.

Overall, this interface was designed with a lot of thought on how to prevent user errors during the interaction, as prevention is always better than recovery. The interface may not look as visually appealing as the 1st interface but is just as usable and even more functional. It was made using User Centered Design and this is one of the reasons why the interaction is so simple and understandable and why User Centered Design is such an important design process for creating usable interfaces.

3.3 Current and future status

In the future I will finish up development of both my interfaces ensuring every component has been tested thoroughly and implementing any feedback from my interim submission. After this I will start the process of development for my last interface – a recipe tracking, android application. This will require me to research a lot over the Christmas holidays and when the term starts to thoroughly research requirements before even planning an interface.

For this interface I want to combine aspects from both of my original interfaces to create one final usable interface. This involves putting together the consistency and visually appealing look of the first with the functional and communication focus of the second interface. However, this will be a challenge when developing an android application that must take in a number of factors. Such as the different screens that will require the typography, elements, and colours to adjust, the different interaction with touchscreen and the different accessibility features that are involved with this. This will be a new challenge and one that will require a lot of research to get right.

I will develop prototypes and use an iterative feedback process develop the interface just like I did for the first term, and this will ensure that UCD is incorporated into my design. Developing the first 2 interfaces has given me experience which allows me to complete this process in a streamline fashion and learn from the errors I made for the first 2.

4 Software Engineering

4.1 Git (version control)

A revision control system is an incredible tool that helps maintain a project over time tracking its changes and allows me access from any device. I used it to split different features of a system, which permitted me to focus directly on that one feature and experiment different ideas and designs without affecting my main code in the master branch. I used features like tags to allow me to show the different releases of my interface at different times and allow for a quick comparison of changes. Overall git ensures the process of development was smooth and avoid errors that can occur.

For example, one instance was when I first used Javafx for my second application, I deleted this Javafx project inside Eclipse as I wanted to change language, but this deleted not only the second application but also my first. Without git, I would've lost my first application but instead I just had to pull from the remote branch – acting as a backup.

4.2 OOP

Object oriented programming is a crucial tool in SE that I used to break down complex projects into small modular components. This allowed me to reuse the modules, easily test them, and make changes whenever needed. We can see this in my React finance tracker, where I split nearly every complex feature into its own component. I used the Navbar component in every page and changed its colour scheme by using a clever JavaScript function. We can also see this in my python project where every separate file contains at least one class that has been thoroughly tested to see if its layout is correct and fits the requirements of my project.

This modularisation that comes from OOP also allows me to make frequent changes to any part of my code which is especially required in a HCI project like this where I will be changing the layout regularly and adding new features that improve the user interaction and experience.

4.3 Testing

My supervisor told me that TDD isn't required for this project and would be quite difficult in a project like this where we will be often changing the whole code or components regularly. And so instead I tested my code after development of a feature had concluded. This allowed me to see the element would format correctly when separate from other components and that the functionality works as intended.

As well as layout and functionality testing, I also thoroughly used user testing as described throughout the document. This is a tool that allowed me to understand my interfaces from a completely new perspective picking out weaknesses that I was looking over and could not noticed. I used multiple forms: Hallway, structured, questionnaires and these all inputted into creating my final interface.

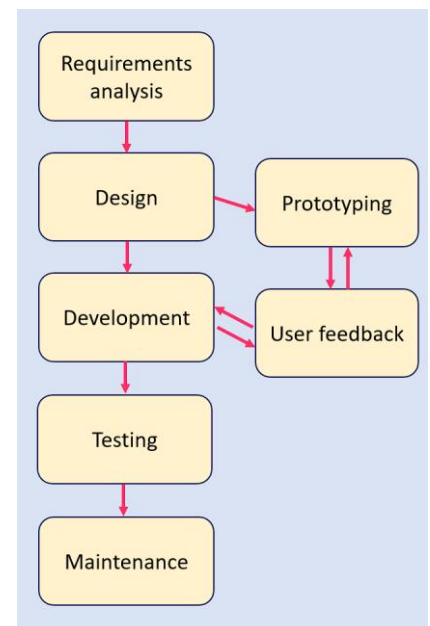
4.4 Technologies

When I first created the financial tracker I used basic html, css and javascript which can be seen from the first weeks of my commits. I used the library bootstrap as it was responsive and had stylish elements that allowed me to implement HCI rules such as consistency and accessibility with little changes. However, after a few weeks I realized that I had a lot of duplicated code, and it was being a giant mess developing all this code using html. So instead, I switched to react which provided modular components that allows easy restyling and changes at any point in the development phases.

I had a similar change for the project application as well. I was using JavaFX at first but found it very rigid and the interface was not progressing how I wanted it to. So instead, I shifted to python and a library called Tkinter that allowed me to create a good-looking application with complete freedom over every aspect.

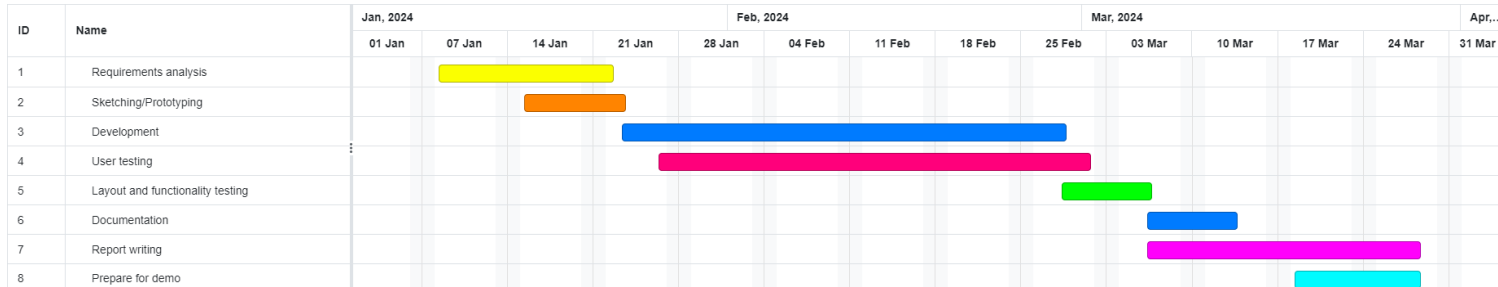
4.5 Waterfall

When designing my 2 interfaces, I developed them using structured steps first beginning with requirements analysis and prototyping as described in 3.1. Getting user feedback after this and during development of my interfaces. After that I tested the interfaces and applied maintenance patches and post development activities. This follows the general layout of Waterfall with a bit of agile integrated in, by getting user feedback at every stage of the development process. I chose this Waterfall route as it provided structure, well-defined phases which I found easier to develop software with. Sometimes I had to make changes due to feedback, but with allocated time for each part of the process, it ensures that nothing overran too much and allowed me to complete my work before the deadlines.



5 Planning and time scale

5.1 Gantt chart



The Gantt chart is self-explanatory with RA starting first, sketching starting after initial gathering and finishing at similar times. Development will start after this with iterative user testing and functionality testing after development has concluded. Documentation in the form of both user manuals and commenting code will begin towards the end of development as well as my final report.

One thing to note is that there may have to be new changes added to my old interfaces after feedback from my interim submission. And at any time, my supervisor may give me feedback which could alter this timeline. But for now, this is the plan I will be following from the interim submission to the final.

5.2 Diary

I have included the diary inside my appendix of the report which has entries split into 3 sections, completed, problems and completed. This was a key tool for me to keep track of my issues and future steps that needed to be taken, letting me to organise my thoughts from a different angle. It allowed to understand what I needed to work on when I opened my laptop and ensured I stay on course with my Waterfall timetable. The Diary was also a great help when writing this report as it allowed to remember which issues I had at the time and what steps I took to recover from them.

6 Bibliography

- [1] Basil, J. (2023, October 24). Exploring The Impact Of Color Psychology On User Experience. Retrieved from OpenReplay Blog: <https://blog.openreplay.com/exploring-the-impact-of-color-psychology-on-user-experience/#:~:text=Colors%20can%20arouse%20emotions%20and,create%20stronger%20relationships%20with%20people>
- [2] Voss RP Jr, Corser R, McCormick M, Jasper JD. Influencing health decision-making: A study of colour and message framing. Psychol Health. 2018 Jul;33(7):941-954. doi: 10.1080/08870446.2018.1453509. Epub 2018 Apr 18. PMID: 29667448
- [3] <https://www.hci.org.uk/article/the-evolution-of-human-computer-interaction-a-review-of-the-past-and-future-directions/>
- [4] Dix, Alan, et al. Human-Computer Interaction. 3rd ed., Pearson, 2004
- [5] Lidwell, W., Holden, K., & Bultler, J. 2010. Universal Principles of Design. Rockport Publishers
- [6] Shneiderman, Ben, et al. Designing for the User Interface: Strategies for Effective Human-Computer Interaction. 6th ed., Pearson, 2017
- [7] Nielsen's heuristics: <https://www.nngroup.com/articles/ten-usability-heuristics/>
- [8] PC3001 User centered design lecture notes
- [9] Nina Hollender, Cristian Hofmann, Michael Deneke, Bernhard Schmitz, Integrating cognitive load theory and concepts of human-computer interaction, Computers in Human Behavior, Volume 26, Issue 6, 2010, Pages 1278-1288, ISSN 0747-5632, <https://doi.org/10.1016/j.chb.2010.05.031>.
- [10] William Jones, Jared Spool, Jonathan Grudin, Victoria Bellotti, and Mary Czerwinski. 2007. "Get real!": what's wrong with hci prototyping and how can we fix it? In CHI '07 Extended Abstracts on Human Factors in Computing Systems (CHI EA '07). Association for Computing Machinery, New York, NY, USA, 1913–1916. <https://doi.org/10.1145/1240866.1240922>
- [11] <https://www.analyticssteps.com/blogs/human-computer-interactionhci-importance-and-applications>
- [12] Stephanidis, Constantine. "Design for All" Interaction Design Foundation - IxDF. 8 Dec. 2023 <https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/design-4-all>
- [13] Gonzalez-Holland, E., Whitmer, D., Moralez, L., & Mouloua, M. (2017). Examination of the Use of Nielsen's 10 Usability Heuristics & Outlooks for the Future. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 61(1), 1472-1475. <https://doi.org/10.1177/1541931213601853>

- [14] Still, Brian, and Kate Crane. *Fundamentals of user-centered design: A practical approach*. CRC press, 2017.
- [15] Joshi, Anirudha, Nandlal L. Sarda, and Sanjay Tripathi. "Measuring effectiveness of HCI integration in software development processes." *Journal of systems and software* 83.11 (2010): 2045-2058.

Appendix - Diary (Not included in word count)

Week 2 (2/10/23)

- Completed:
- first draft of Project plan and have sent to supervisor to get his opinion on my plan.
 - In this plan discussed some technologies like JavaFX, Django, spring boot, react
 - Have completed some basic HCI reading for my abstract description learning about why usability is so important and some basic rules.
- Problems:
- Tough to cover such a massive subject and convert this into a few paragraphs
 - Many types of technology that have their own advantages and my knowledge is especially great for any of them
- Future:
- do more research to cut down technologies to what I am actually going to use for the interfaces
 - Continue reading about HCI and add to paragraphs for final draft

Week 2 (06/10/23)

- Completed:
- Good foundation reading of HCI learning about the balance of functionality and usability, accessibility issues, basic colour theory and a few HCI principles.
 - Researched and decided the technologies I will be using: Javafx, bootstrap and Springboot
- Problems
- It was tough to decide the technologies and I still might have to change them if I can't learn or adapt to using them fast enough
 - Had to add more to nearly all aspects to my report due to feedback from supervisor
- Future:
- Begin setting up the html to run a page - will use visual studio live coding extension
 - Start adding in libraries that will allow me to create elements

Week 3 (11/10/23)

- Completed:
- Created home page for first UI adding navigation bar, background image, fitting colour schemes, buttons and text to direct the user
 - Set up framework spring boot to add some basic functionality to the website and allow it to communicate with the database
- Problems
- Haven't adapted elements for minimizing yet and so some remain fixed while others move.
 - Tough to create a design that looks good for everyone after getting feedback. Some like one design and others prefer another.
- Future:
- Get feedback from supervisor at next meeting
 - Add some functionality to home page and begin working on another page for the first user interface
 - Read up on accessibility from books and articles online

Week 3 (15/10/23)

- Completed:
- Nearly finished fixing elements for the bug discussed in problems.
 - Fixed indentation of css and jsp file
 - Completed sketch of enter data page using canva
- Problems
- Discovered bug in home page where minimizing doesn't affect all elements the same and some are off screen.
 - Sketched elements will be tough to create, for example a pie chart that updates when entering data - may need to redo sketch
- Future:
- Set up enterData page and begin coding. Need to add table, colour scheme, title, navbar again.
 - Add navigation link from home page to enter data page

Week 4 (18/10/23)

- Completed:
- Nearly finished enter data page adding features which were described last week

-Read some basic features about accessibility e.g Image annotations, logos, page readers etc

Problems: -Couldn't add a good title to the page, spent a lot of time experimenting with different fonts and sizes - looks very amateurish

-Not sure if I need to do tdd or create a uml as it says one thing on Moodle but other people are saying we don't.

Future: -Add some basic functionality to enter data page

-Begin working on other pages

-Implement researched accessibility if I have time and also carry on research on different rules.

Week 4 (22/10/23)

Completed: -Refactored design of enter data page as I wasn't happy with the design, especially the bootstrap items and so redid the code using vanilla html and css.

-Added extra features like sidebar content, more buttons for the table and improve colour scheme.

Problems: -Pushed back implementation of accessibility even more and development of other pages

-Haven't added functionality to the page as will be quite time consuming and just added dummy data to the table - not sure if I will be penalised for this.

Future: -Finally begin other pages and finishing up with this application

-Start a bit of the write up about this implementation and how I used HCI rules

-Implement accessibility features (might do after finishing application)

Week 5 (25/10/23)

Completed: -Finished help page and started display data page

-Found how to use WAVE accessibility tool and have been planning on how to organise pages to include more hci principles

Problems: -My friend got feedback from his meeting, in which he was reprimanded for using html which is what I have used.

 -Falling behind in schedule and need to start report

Future: -Learn react

 -Need to convert all my html code to react code, so that I am able to get all available marks for complexity

 -Need to finish last page of application

 -Start report

Week 5 (29/10/23)

Completed: -Learnt react in a short space of time

 -Finished converting code to react

 -Finished last page of application

Problems: -Didn't realise css pages overlapped, had to spent huge amounts of time fixing that

 -My whole timeline is a mess right now

Future: -Start 2nd UI

 -Learn swing language

 -Make good progress on report

Week 6 (31/10/23)

Completed: -Research online about different java GUIs and found JavaFX to be more appropriate

 -Set up development environment

 -Started early development

Problems: -Don't have access to last year's javafx videos, had to do research to learn how to set it up

Future: -Start development of lists and cards
-Need to add text boxes and icons
-Research what HCI from notes can be applied to interface

Week 6 (5/11/23)

Completed: -Added lists to main and set up css so that it doesn't overlap title boxes
-Add styling to all elements
-Added textboxes

Problems: -Trying to get list to not overlap the title of boxes was difficult
-Struggling to make a usable design following prototype using javafx

Future -Continue development adding help tips and icons to icon box
-Start working out how interaction will happen and how to add cards

Week 7 (9/11/23)

Completed: -Decided to switch language to tkinter as javafx was very inflexible and I didn't like the look of my UI
-Added several elements: Navbar, lists, 2 cards and buttons to move between lists

Problems: -Had to learn tkinter while implementing this UI
-A lot of problems where height and width were being controlled by another variable

Future: -Add way to track visibility of system status
-Add options to add cards, undo and redo and begin testing

Week 7 (12/11/23)

- Completed:
- Added checkboxes and progress bar to the cards
 - Added some extra menu options and icons to these
 - Started testing elements
- Problems:
- Was having problems inputting images and changing their sizes
 - Had to change what elements returned to test elements
- Future:
- Go to supervisor meeting and show uis and report
 - Continue testing and report

Week 8 (16/11/23)

- Completed
- Started adding a lot of functionality to my ui as advised from my supervisor.
 - Added moving between lists, adding new card, and options to manually control checkboxes
- Problems
- Have to spend a large amount of time on UI's
 - Tests are going to have to be redone
- Future
- Continue adding functionality.
 - Go back to old user interface and add functionality to that as well and test it
 - Keep writing report

Week 8 (18/11/23)

- Completed
- Added a scrollbar to each list so that multiple cars can be viewed
 - Refactored code using oop principles
 - Changed design of code

Problems referenced -Struggling making scrollbar, had to use internet to create code - has been

-Could not decide on colour scheme

Future -Refactor classes into their own files

-Test this UI thoroughly before moving onto old one

-Finish report to send to supervisor late next week

Week 9 (22/11/23)

Completed -Added delete button to card, realised it was missing

-Changed colour scheme again

-Fixed old tests

Problems -Had to change structure of tests a lot due to putting navbar into its own class

-Unable to change some hover colours

-Still got some sections of report to do

Future -Keep testing

-Send report to supervisor on friday

-Test all the extra functionality I added

Week 9 (25/11/23)

Completed -Refactored the main classes into their own files using oop principles

-And some tests for the layout of the ui

-Sent report to supervisor

Problems: -Took a lot of work to refactor requiring a lot of variable name changes and errors
 -Testing also required me to change what methods to return

Future: -Work on feedback from supervisor early next week
 -Start PowerPoint
 -Finish testing and change buttons of move right and left

Week 10 (28/11/23)

Completed -Thoroughly tested all classes for interface 2
 -Add extra functionality like undo, redo and help window when button pressed
 -Refactored old tests that broke with new changes

Problems: -Had a lot with tests, required a lot of refactoring and change of return values
 -Undo and redo took a lot of effort to get working - bugs with undo doing redo job
 -Subtasks where moving checkboxes around when long text added

Future: -Comment all my code
 -Move on to interface 1 - extra functionality and testing
 -Write up PowerPoint
 -Add improvements to report which were suggested by my supervisor today

Week 10 (03/12/23)

Completed -Submitted PowerPoint and recorded videos of both my interface for it
 -Commented my 2nd interface
2nd) -Added a lot of functionality to my 1st (now gone back to it after finishing with

Problems: -I have an intermediate level in react, so tough to add some complex functionality and had to fake parts

to fix -Had a lot of spacing problems causing scrollbars to appear - used overflow in css

-Need to fix report mistakes still and sent to supervisor asap

Future: -Start testing 1st ui

-Tag releases

-Send report to supervisor asap and revise ppt for presentations on Wednesday

Week 11 (07/12/23)

Completed: -Tested all UIs

-Sent report to supervisor 4 days ago

-Filmed video for report and done presentation

Problems -Had a lot of speaking errors when filming video

-Components were frustrating to debug when testing

-Couldn't add 100% support for colour blindness

Future: -Complete documentation

-Tag last release

-Fix citations in report