



# *Cardiff Youth Service Digital platform for staff*

Final Report

CM3202 – Individual Project

40 Credits

Saul Ollett: C1989661

Supervised by: Catherine Teehan

Moderated by: Bailin Deng

## Abstract

Youth Services can be an incredibly important part of many young people's development for their later adult lives. They aim to provide young people with plenty of educational and recreational leisure-time activities to improve their well-being, social, and personal development. Youth Services need scheduling systems that can help them organise their workload and all the events they plan, they also need a scheduling system that allow them to be able to work efficiently, seamlessly, and time effectively.

The main purpose and focus of this project is to create a scheduling application that will aid Cardiff Youth Service so that all members may work more efficiently and agile at any given time or place with any device they want if they don't have access to a laptop or computer from home. I aim to achieve this purpose by creating this application so that any member may easily access their schedule on a device such as a mobile device at any given time, allowing them to work more efficiently and agile.

The following report highlights that all of the aims and objectives have been met and completed. A working prototype has been made that allows a user to add bookings of any kind and notes to those bookings, deeming this project a success.

## Acknowledgments

I would like to thank and acknowledge my supervisor Catherine Teehan for her time, dedication, and support throughout this project as well as providing me with advice through any difficulties I had faced. I would also like to thank Catherine for taking the time out of her incredibly busy schedule to answer and queries I had.

Furthermore, I would like to thank Dayle Luce, a senior youth officer at Cardiff Youth Service, for their support and feedback during the early development of this project, providing me with insight to how Cardiff Youth Service is currently operating with their scheduling system and for providing plenty of feedback on early wireframe designs for the prototype.

## Table of Contents

Abstract.....	2
Acknowledgments.....	2
1. Introduction .....	4
1.1 The Scope .....	4
1.2 Intended Audience .....	5
1.3 The Aim & Objectives .....	5
1.4 Objectives:.....	6
2. Background .....	6
2.1 Background Research .....	6
2.2 Existing Solutions .....	7
2.3 Initial User Research and Feedback .....	8
2.4 Project Justification .....	9

2.5 Project Constraints .....	10
2.5.1 Time Constraints:.....	10
2.5.2 Learning curve constraints:.....	11
3. Approach and Methods.....	12
3.1 Agile Methodology .....	12
3.2 Web Development .....	12
3.3 Data Storage .....	14
4. Design.....	14
4.1 Initial Wireframe and User Interface Drafts.....	14
4.1.1 Wireframes.....	14
4.1.2 User Interface Designs.....	15
4.2 Wireframe Designs and User Interface designs.....	15
4.2.1 Wireframe.....	15
4.2.2 User Interface.....	18
4.2.3 Wireframe Strengths and Weaknesses.....	19
4.3 Design Colour Scheme:.....	20
5. System Specification .....	21
5.1 Functional Requirements.....	21
5.1.1 Must Have .....	21
5.1.2 Should Have.....	22
5.1.3 Could Have .....	22
5.2 Non-functional Requirements .....	23
5.2.1 Must Have .....	23
5.2.2 Should Have.....	23
5.2.3 Could Have .....	24
5.3 Use Cases.....	24
5.3.1 Use Case Diagram .....	24
5.3.2 Use Case Tables .....	24
5.4 UML Class Diagram.....	26
5.4.1 UML Class Diagram Description.....	27
6. Implementation.....	28
6.1 Database Design and Storage solution .....	28
6.2 Web Development .....	29
6.3 Framework Implementation.....	29
6.3.1 HTML and CSS Implementation.....	33
6.4 Hosting the Application .....	36

7. Results .....	37
7.1 CYS Scheduling System Prototype Test Cases .....	37
7.1.1 Functional Requirements Check.....	37
7.1.2 Non-Functional Requirements Check.....	40
8. Final User Feedback .....	42
9. Conclusion.....	42
10. Future Work .....	43
11. Project Evaluation .....	44
Appendix.....	44
References.....	44

## 1. Introduction

The project was initially pitched to me by Catherine Teehan as a collaboration with CYS. CYS is a youth service organisation that aims to help young people between the ages of 11 and 25 to develop all the personal, educational, and social skills that will aid them into their later adult lives (Cardiff Youth Service, 2021). The organisation aims to achieve this by providing those young people with opportunities, activities, information support and experiences that they may have never been able to have before (Cardiff Youth Service, 2021). These opportunities are put in place to help guide the young people who attend to reach their full and unique potential. The current system that a lot of youth services seem to have makes it harder for some people to easily use the systems provided to create a simple and easy to read schedule for meetings and in some cases, certain services are still on pen and paper, whiteboards and applications such as Microsoft Excel to make their schedules, all of which can either take up too much time or can easily be lost particularly in the case of pen and paper. The goal of this project is to make some form of application or tool that CYS can use to be more efficient, effective, and agile and to make it easier for any employee or volunteer to use the system in place to create their own schedules for meetings. The main aspiration for this project is to allow anyone of any age and technical ability to use the application to its full potential so that they can easily organise a schedule with events and sessions and to also allow the employee or volunteer to make notes on said sessions and events.

For some users, especially those who may be older or lack experience in certain technologies, it can be very daunting to use an application to organise their week-to-week schedules that tend to get full of sessions and events, which can then be difficult to keep track of and in some cases cause overlaps in certain bookings, which can make things even more difficult. This causes hassle and stress for those who do struggle in these areas, so with this in mind, I was determined to make a system that would be simple for them to use, that will circumvent any overlaps, and that can be easy to keep track of daily.

### 1.1 The Scope

The scope of this project has already been determined by CYS. CYS have asked me to create an online tool or application for any employee or volunteer that will allow them to work from wherever they want efficiently and in an agile way. The application they have asked me to create for them is to be put in place of their current system they have been working with and has also been requested to allow employees and volunteers to add events and sessions to their weekly schedules and for

everybody to be able to see these changes to the weekly schedule, specifically for adding events. This new system that they are asking for, facilitates weekly schedules creation for all employees and volunteers, as currently some members struggle with having to use multiple applications on their computers and some employees just do not have access to a laptop or PC from their home. Subsequently, this project needs to be able to function on several different devices so that anyone can work with the new system at any time and at any place.

## 1.2 Intended Audience

The intended audience of this project is the employees and volunteers who work with CYS. These employees and volunteers are from all different kinds of backgrounds, and all have different understanding of current technologies in place for scheduling, so it is crucial to keep in mind these differences so that the application can be user-friendly and as accessible as currently possible. With the features that will be implemented into the system, any user will be able to easily navigate the application, add any session or event they wish to, edit their weekly schedule, and record reflections from a session. As I do not know how old a lot of the intended audience for this application will be, the overall design and functionality must be simple enough for anyone to comprehend and it also must be easy enough to use the features that will be added to the application.

## 1.3 The Aim & Objectives

The primary aim and objectives of this project is to create an application for CYS for any staff or volunteer of the service to be able to work from any location, whether they are part of the street-based team or any other team without sacrificing efficiency. I will be making this application by creating a timetabling / scheduling system that will allow the users at CYS to seamlessly record notes in any scheduled session and create any events that will get added to the weekly schedule, which should be visible to everyone. The aim of this is to make CYS more efficient when it comes to staying organised with all the sessions and events that they must schedule in each week. Another main aim is to ensure accessibility to anyone who wishes to use the system, so the prototype will have a simple, easy to read and navigate UI design so that anyone of any age or technical background is able to figure out how the system works.

Having some sort of guide / digital instruction manual will also be of great use to CYS as they will be able to manually add, change, or fix parts of the coding if they need or wish to in any future developments. This would also be very useful to a hired developer as it means that they will not get lost and will be able to understand how they could make future improvements to the application, which would mean that CYS would be able to be more self-sufficient as they would require less assistance with time.

The user should be able to:

- Edit their weekly schedule for each month
- Login or register to the application
- Record any reflections from any session
- Add sessions and events
- View any events that get added

The administrator should be able to:

- Check all weekly events and sessions
- Cancel or approve any event or session

- Login or register to the application

## 1.4 Objectives:

Below is a list of the objectives that have been identified previously in my initial report, as necessary to complete this project. In any situation where an objective has not been met or fully completed, a justification will be provided.

- To design and plan what I want the application or tool to look and function like, by drawing up a couple of quick sketches.
- To choose an appropriate environment and language to start developing the prototype application.
- To get some qualitative and quantitative feedback from employees on how they feel they could work from home easier and how they feel they could be more efficient when using the tool.
- Create a basic prototype, wireframe model, with barebone functionality to get an understanding on how it can work and the areas of improvement for efficiency.
- To start building the front end of the application, e.g., overall design, feel and functionality of the application.
- Start developing the back end of the application, where data will be stored and the general functionality development for the features required.
- To adjust front-end issues based off user feedback.
- Finalise the prototype and trial it with a focus group with employees at CYS
- Research if there could be more ways to use or implement the application e.g., an app or website.
- Review the final product and discuss what could be further done to the application if it was to be accepted by CYS.

## 2. Background

### 2.1 Background Research

Traditionally, youth services are crucial to assisting with the overall development of young people and their integration in society by providing a varied ranges of programmes in health, safety, education, economic well-being, participation and generally promote values of positive contribution to local communities and British society (McKee, Oldfield and Poultney, 2010). Within the same report by Unite the Union, there is a numerical comparison of the benefits and costs of the effects of youth work, specifically it mentions that a young person in the criminal justice system costs taxpayers over £200000 while a young person supported by youth services costs less than £50000 (McKee, Oldfield and Poultney, 2010), which shows that the benefits clearly outweigh the costs of youth services.

Having seen the above image prior to the pandemic it is important to note that due to the above benefits and more that will be analysed below, the Educational Policy Institute has included expansion of youth services and wider children's services as part of their proposal for the recovery of children's development and education after the closure of schools and other providers (Educational Policy Institute, 2020).

After the pandemic youth services have become increasingly important in reaching young people and particularly disadvantaged young people and helping them recover from the learning losses incurred during the pandemic. According to the Educational Policy there are learning losses in both

reading and mathematics for secondary education (years 8 and 9) 1.6-2.0 months whilst due to small samples they were unable to gain robust samples (Crena-Jennings, Perera and Sibieta, 2021). For schools with high levels of disadvantaged students, where disadvantaged students were defined as those receiving free school meals, learning loss was around 2.2 months whilst for schools with low levels it was around 1.5 months (Crena-Jennings, Perera and Sibieta, 2021). When we consider that attendance was on average between 80 and 85% for the autumn semester for 2020, these significant learning losses are easily explained by the data (Crena-Jennings, Perera and Sibieta, 2021)

Furthermore, young people approach youth services to get access to help for their mental health and during the pandemic there have been significant changes in the statistics for mental health related issues. According to NHS Digital, rates of probable mental disorder have increased between 2017 and 2020, for children we have 16% identified as having a probable mental disorder in 2020 compared to 10.8% in 2017 (Vizard et al., 2020). For young people aged 17-22 the percentages of such an occurrence were 20.0% in 2020 with young women at 27.2% and young men at 13.3% whilst for children the increase was similar for both genders (Vizard et al., 2020). It is important to note that 62.6% of children with probable mental disorder had regular support from their formal education provider compared to 76.4% of children who are unlikely to have a mental disorder (Vizard et al., 2020), which further shows the importance of youth services in filling this gap for those with mental health issues and for them to have the appropriate resources to do so.

Despite providing all the above benefits, youth services face significant challenges particularly a lack of funding and infrastructure both physical and digital. Before the pandemic they were already facing issues with funding considering spending per head per annum of 13-19 population was £100 whilst the needed funding would be £350 per young person for all to access youth services (McKee, Oldfield and Poultney, 2010). However, the pandemic has placed further limitations on youth services. The UK Youth report for their 2020 survey on 235 youth organisations across the UK summarised the impact of the pandemic on them and found that 64% of respondents thought they were likely to lose funding (UK Youth, 2021). Regarding digital infrastructure, the same report mentioned that 65% of respondents identified an urgent need for digital infrastructure and support whilst 31% of them said they do not have that infrastructure to be able to provide digital services (UK Youth, 2021). This particular point ties in with the project justification further on, as the Cardiff Youth Service prior to this project lacked a scheduling system and hence lacked this digital infrastructure needed to provide digital services efficiently.

Overall, from the above evidence we can see that youth services are important during normal circumstances to support schools, families and other providers with child and adolescent development in a variety of areas and to support young people with their personal journey to becoming well-adjusted and balanced adults who contribute to the local society and economy. We can also see that they are even more important in crises in this case the pandemic when educational provision and family life was disrupted considerably as they bridge educational and developmental gaps for all young people but even more so for those who are disadvantaged.

## 2.2 Existing Solutions

For a problem such as this, there are already a variety of solutions when it comes to online scheduling applications. However, a lot of these scheduling systems are often pieced together from separate applications or even using pen and paper. From what I have gathered in my meetings with CYS, I have discovered that they are using different applications for recording notes from sessions and events, and a different application for organising their weekly events and sessions. Whilst some applications are more attuned to scheduling, most of the time you will find them making notes and

having to switch to different windows which often can take a while if the user is not tech savvy. Having notes specifically on applications such as Microsoft Word means that there is a possibility that files can either become corrupted or lost which can be extremely inefficient.

One existing solution that I have managed to identify is the application Microsoft Power Apps. Microsoft Power Apps is an application that allows users to make a system that is custom made for them. An example of this would be when someone uses the application to make a schedule. This would effectively allow them to make a schedule with all the tools they need to use. The problem is with this application, though it is simple to use, it does require the user understanding how to make apps through Power Apps, which can take more time than they might possibly be allowed. Power Apps is a great tool to make your own scheduling system however that is not the aim of this project. However, it does come with several disadvantages such as the cost of the application (Bordoli, 2022). The cost of Power Apps per user annually is \$120 which depending on the amount of users on the application can make it very pricey to use, in addition to this, the Power Apps Portal costs \$200 for 100 monthly logins and then it also costs \$100 for 100,000 webpage views monthly (Bordoli, 2022). Whereas this project aims to make a free application for all users within CYS. Another disadvantage to Power Apps is that it has an item limit which is not good for a lot of companies. It can connect to data sources such as SQL or Oracle which can hold a larger number of items, so having this limitation would not benefit some companies such as CYS which could potentially have a requirement for a larger number of items to be held.

Another application that I have found that is similar to this project is Google Calendar. Google Calendar is a great tool that is used for organising your weeks and months. It has the capability to add any plans to each day and even plans that stretch multiple days like this application prototype I have been working on. Google Calendar also has the feature to colour coordinate your plans when there are different types of events happening which I have taken inspiration from as this was something I was intending to add at some point during the development. However, there are a few disadvantages to Google Calendar which would not make it suitable for CYS (McOmie and Ionescu, 2022). One of these disadvantages is that the weekly view of their calendar seems to get messy and difficult to understand (McOmie and Ionescu, 2022), which would be detrimental for CYS as they are looking for a scheduling application that is easy to keep track of and that avoids any sort of overlapping in the week-to-week schedule. Another issue with this that I have noticed, is that in some cases Google Calendar does not save reminders for upcoming events properly which means that it is possible for a user to either miss an event entirely or they can lose track of when the event is taking place, this again would be a significant problem for CYS as it means that they could potentially lose track of their bookings which is the opposite of what I aim to do with my prototype for them.

### 2.3 Initial User Research and Feedback

In the early stages of development, it was crucial to get feedback from someone who worked with in CYS to get a better understanding of what they wanted and expected from the prototype of this project. For this I held a few meetings with my main contact at CYS to discuss design and functionality ideas that they felt like are important to the overall look and feel of the application. The meetings additionally provided us with the opportunity to discuss how the youth service is currently operating and the struggles they are facing with the systems in place. This process improved my understanding of what I could add to the overall design, layout, and functionality in my prototype so that they will be able to work more efficiently and effectively. During one of these meetings, I got some feedback for the wireframe design concepts I had come up with for the application and it allowed me to get a better understanding of what design ideas they were happy with and to see if I



had missed anything in the design that could or should be added to the final product. Luckily, they were very happy with my overall design since I had only the brief that I was provided with, and I was able to come up with something that was feasible and comprehensible to any user. There were a few areas that I still had to work on, such as adding a page with a clear link to CYS as a sort of about page so that anyone who uses the application can then see what CYS aims to accomplish.

Unfortunately, I was not able to hold a meeting with any other employee of the youth service or with any volunteers who worked there, which did add to the challenges I faced, since I only had one person giving me feedback on the prototype, whereas having more employees or volunteers that could have given me feedback would have proved extremely useful as it would have given me more than one perspective to consider and I could have more easily catered the prototype to everyone who worked with the youth service. However, having a senior youth officer to give me feedback was vital as they have been working at the youth service for many years and therefore know the intricacies of the current system and why it is challenging for some members to use the current system. The majority of the feedback I did get in all of the meetings I held was very positive and it meant that I did have less stress when working on developing the prototype and though there was some feedback on what could be added or what the youth service could potentially use the system for, it was never put as a hard requirement, as they could always hire a professional developer later on to work on the prototype I have worked on, and this meant that I did not have any additional weight on my shoulders at the time.

## 2.4 Project Justification

During the meetings with my main point of contact, it was highlighted that they currently have multiple challenges with their current system they use for scheduling, and it also became very clear that they needed a new system that would work for everyone. In our discussions, it was noted that whilst they do have a system in place to make schedules for all their sessions and events, in some cases people were still using pen and paper which is not as reliable as computer systems. Whilst CYS was aware of other solutions and they were using excel and notepads for making their schedules, it was clear that they wanted a system that bespoke to all their needs and to ensure that it was easy for all their members to use. Whilst, I do not have all of the insider knowledge of most youth services as I have not had the opportunity to work with any other service, I was told by my main point of contact that they do have an understanding of how other youth services do currently operate with scheduling systems and they noted that they were not the only youth service that struggles with using a system that will allow them to make a week-to-week schedule. So, this helped justify why I was making this system as it would be versatile for other services, and it would improve how they currently operate.

For this project to be seen as a success by both the CYS and I, this application would need to include:

- Mobile and other devices accessibility to enable any employee or volunteer to work from wherever and whenever they can. The user must be able to have full access to the application and all its functionality.
- A design that is simple, easy to read and navigate that is understandable to anyone of any age and technical background to use to its full potential. Something that is easy to read and understand on all devices.
- Functionality that allows the user to make bookings, either sessions or events, easily and without much hassle. Performing as the main scheduling application for easy organisation of week-to-week plans.

- A system in place that prevents any type of overlapping on an individual's weekly schedule to ensure that no booking is forgotten. Alerting the user when there is a booking in place.

## 2.5 Project Constraints

As part of my initial report, I set out the task of making a work plan for the development of my project so that I could highlight any challenges that I might possibly face and how I could overcome them so that I may continue my work as efficiently as possible. In the below sections of the report, I have written what I believe to be the constraints I consistently had to face during the development of my project.

### 2.5.1 Time Constraints:

One of the biggest constraints that I faced several times throughout the development process of the project was having a time constraint. Whilst we officially started to work on the projects at the start of February and we did have until the middle of May to get the prototypes created and our dissertations written, there was a lot to fit into and sort out during this three-month period. Part of the reason this was an issue was due to the amount of preparation work I had to do for the project. One of the biggest parts in developing the prototype for CYS was having to either refresh my knowledge on specific coding languages such as JavaScript, which proved to be time consuming and did take up a couple days in the first initial weeks of development. There was also the issue with learning newer languages such as C# and starting fresh with MySQL. Another element to the time constraint was having the Easter Period during the development of the project, since whilst it was not a big issue, it did mean the occasional travel here and there to see family. This time that I had to spend travelling meant that there were some days where I was either completely unable to work on the project, or there were days that I could only slot in a few hours, which in hindsight did slow me down over this three-week period. However, these time constraints only slowed me down a couple of times and I was able to constantly move forward with the project and therefore not get lost behind in my own schedule.

To tackle any time constraint issues, I may end up having to face at any given moment, meant that I would need to have some sort of plan in place to find a way around the problem at the time. So, for this I created a Gantt Chart for my initial report as that is where I had to make some sort of plan to tackle the development of this project. Using a Gantt Chart allowed me to precisely pin what tasks needed to be completed when and by what date. Since this was my first time taking on a project of this scale, it was crucial that I was able to break down each task into sizeable chunks each week and it meant that I would not go off track too often. Since the duration of this project was roughly 14 weeks I had to be careful not to spend too long on specific sections of the development process but to also ensure that I was getting enough done each week, so this meant that I would need to make sure that the tasks for each week were feasible and attainable otherwise I could end up spending too long in certain areas and ending up with an unfinished prototype which is not what I wanted in the slightest.

Weeks / dates	Deliverables	Documentation	Milestones
Week 1 / 31 <sup>st</sup> Jan- 7 <sup>th</sup> Feb	Background research on Cardiff Youth Service for the initial project.	Initial plan / report	Initial Plan submitted
Week 2 / 8 <sup>th</sup> Feb – 14 <sup>th</sup> Feb	Create the basic look and feel of the application with quick sketches Research what environment and languages would be best. Have a meeting with supervisor to sort out any unclear areas of the project	Draw up lightning and quick sketches of the look and feel for the prototype. Also have a storyboard in place for how it should ideally operate.	General look and feel of project covered and completed. Cleared up any uncertain aspects of the project so I can get on.
Week 3 / 15 <sup>th</sup> Feb – 19 <sup>th</sup> Feb	Get in touch with Cardiff Youth Service to brief them on the project. Finish independent research for what language and environment to use Gather data from Cardiff Youth Service for quantitative / qualitative models.	Write down what language and environment I will be using and why. Data from Cardiff Youth Service in either a qualitative model or quantitative model.	Language and environment decided on and sufficient data collected to work on.
Week 4 / 22 <sup>nd</sup> Feb – 26 <sup>th</sup> Feb	Start creating the wireframe model of the application and start a basic prototype to get a good sense of how it can work.	Wireframe model along with any brainstorming or notes made for it. Start of a basic prototype model.	Decision of wireframe model I think works best for the project.
Week 5 / 28 <sup>th</sup> Feb – 5 <sup>th</sup> March	Begin work on front end design by using the wireframe I decided on in the previous week Look into other examples for inspiration for my application	Write the beginning front end code for the application.	Beginning initial look for the prototype completed.
Week 6 / 8 <sup>th</sup> March – 15 <sup>th</sup> March	Start initial back-end coding for the prototype. Get reviews on the overall design so far from a focus group at Cardiff Youth Service	Begin writing the back-end coding for the prototype and review answers on initial design.	Start of the back-end coding and feedback given for the design
Week 7 / 16 <sup>th</sup> March – 21 <sup>st</sup> March	Fix any design issues stated from the focus group and carry on working on back-end coding Get feedback on the changes made to front-end design	Further back-end coding and redesigned front-end (if changes are required)	Progression with the back-end of the prototype and a more finalised design for the prototype with feedback
Week 8 / 23 <sup>rd</sup> March – 1 <sup>st</sup> April	Finalise the front-end of the prototype based on the feedback received.	Finalised front-end of the prototype.	Finished the overall design / front-end code of the prototype.

Figure 1: Initial timeframe and plan

## 2.5.2 Learning curve constraints:

Finally, there was a learning curve restraint that I faced numerous times in the development of the prototype. This constraint appeared due to me not spending enough time prior to this project to develop my knowledge and strength in some languages that I ended up using. One such example of

this was when I had to refresh my knowledge in JavaScript. Though I have used JavaScript numerous times throughout my years at university, I did not spend time over some of the break periods, such as summer, to hone my skills in the language and therefore better prepare myself for this project. Another reason why I had a learning curve restraint, was due to some languages either not having been used before or very little to the point where I had completely forgotten how to work in that language. This was the case when I had used C# for program and start-up of the program. I had not used C# since I was in college, so this was a language that seemed very foreign to me, and it was quite daunting when having to use it again. But I did spend some time practicing the language at the very beginning of the project which meant that when it came to the point where I had to use it, it was not too difficult to figure out how it should be used.

So, to tackle this constraint, I made sure that I was not trying to learn any new languages with no prior experience or so little experience that I would not know where to start. This meant I was able to focus all my time and effort in strengthening the languages I have selected and to not waste time being confused for a whole week. Moreover, I was able to more easily identify the languages that I would be able to work through quickly and effectively and it helped to narrow down the areas I did not feel confident in, which helped me to plan to ensure that I was confident in my skill and knowledge with the certain languages that required more of my time and energy.

### 3. Approach and Methods

#### 3.1 Agile Methodology

For this project I decided to follow the agile methodology for the overall development process. There are multiple reasons for choosing the agile methodology for this project, the biggest being that I had to work on my own as this is an individual project. The agile methodology permitted me to break down my workload into multiple phases throughout development, making it a lot easier on me as I am working without a team. With the timescale, some of the restrictions, and problems I faced, using the agile methodology meant that I was able to shift around the workload safely with as much flexibility as I needed, it also meant that I had better controls of the issues that arose. The agile methodology also was the easiest method for me to apply as alternatives would have meant that I would be putting too much pressure on myself, therefore making more frequent mistakes and errors throughout the development.

In addition, it empowered me to use the Gantt Chart in my planning. The purpose of the Gantt Chart was to easily assign my work for the project into chunks for each week, which enabled me to devote my full attention to specific tasks highlighted for the week. This enabled me to have a more concise plan for each week and therefore improve my time-management by allowing me to consider whether perhaps I was spending too much on a specific section of the development. The Gantt Chart also enabled me to ensure I have written down dates of when I would be holding progress reviews with my supervisor, which was very useful as she was able to let me know if I was still on track to complete the project in time. As a result, I was able to see if I had achieved any of the milestones, I had set out to accomplish in the beginning phase of development. Having the Gantt Chart also meant that I was able to give myself time to fix any issues I faced with developing the prototype as it can be a timely process and it could have meant that I could fall behind if I was not careful.

#### 3.2 Web Development

At the beginning of the development for this project, I already knew that it would be preferable for this application to be web-based instead of being an application made for IOS or Android, because during my meetings I learnt that they were looking for something anyone would be able to

access without having to worry about what type of mobile device they had. Another reason I wanted to make this a web-based application was because I have some experience in making web-based applications prior to this project and therefore felt more comfortable developing it for the web. If I had gone down the route of making an application for mobile devices, I would have faced much greater challenges as I have had no prior opportunities or experiences in developing any other sort of app. Developing this project as web-based first will allow future developers to easily access and continue development on the application, rather than having it be made for a mobile app to begin with and then for CYS to want to use the application through other mediums. This therefore led me to deciding what languages and frameworks I could potentially use for this application, and whilst there are a lot of options for me to have picked from that ultimately would have ended in a similar end product, I ultimately decided to use JavaScript as my main framework to develop the application in. This of course would be possible to create most of the application using JavaScript as my back end and HTML and CSS as the front end of the application, however, I believe that it was in my best interest to develop the application using C#, JSON, and MySQL, since each of these frameworks could be used in various ways for data storage and to implement controllers. I already had some pre-existing knowledge with JSON and C# as I have used them for some projects prior to this which meant that implementing them to make the prototype work to its highest capability wasn't too much of an issue.

Nonetheless, as I was deciding upon which frameworks I could potentially use within my prototype, there were other options that I considered. The primary alternate framework I considered using was Django. Django while being an extremely popular framework amongst a lot of developers for web-based applications, I simply did not have as much experience or knowledge with the framework which could have potentially caused me a lot of issues and I believe that I wouldn't have managed to finish the prototype to my best ability. I did although want to highlight why I decided to pick my frameworks over Django. The main reason primarily being my experiences, whilst I have had to use Django before in other projects, I have had a lot more time to learn and understand JavaScript. Django overall is an easier to use framework when you fully understand how to use it, however, it is a very daunting framework to work with for someone who has little to no experience with it before, and for someone who has used other frameworks for the majority of my experiences. The other reason I decided to use JavaScript instead of Django was because it is a simpler language and framework to work with in my case and I believe that future developers that could work on this would find JavaScript easier to wrap their heads around from the get-go.

Since I already had a lot more existing knowledge with JavaScript, it meant that getting started on the development of the application was relatively easy. I initially started with the front-end coding to being the development process as I believe front end coding is a lot easier for me to make significant process and it would only need some changes when developing the back end of the application. Whilst refreshing my knowledge on languages such as HTML and CSS I began to get through developing the initial web pages very quickly as I knew that there was not going to be a lot of pages necessary for CYS as they were only after a scheduling system that they could store events and sessions on, along with the notes of those sessions. Since I came across very few issues when implementing the JavaScript to the HTML, I did not have to waste too much time trying to research solutions for any errors that came up which was very useful as it meant that I could constantly work through developing the application with little to no setbacks. Whereas, if I used Django, I think that I would have come across a lot more issues that would have meant that I would need to spend time trying to research and look for solutions that could help with any errors. This means that, I was able to work through the development of the application at a pace that I was happy with and felt comfortable being able to create a scheduling system to the best of my ability.

### 3.3 Data Storage

In the development of the web-based application prototype, I knew that I would have to consider how any sort of data was going to eventually be stored within my system, and I knew that the data would have to be secured in a safe environment which could work with JSON with little to no constraints. I knew that JSON and any SQL format are compatible with one another, and they tend to work seamlessly so knowing this I decided to use MySQL as my data storage. Using MySQL meant that I was able to create my own database for all the data that would be added to the prototype which was great as I knew that I was able to constantly monitor the database to see if there were any issues that had arose. The only issue I had when using MySQL was that I did have to make changes to the database quite frequently which in hindsight did slow me down at some points, however, it did not cause me too many issues and therefore meant that I was able to progress using MySQL. MySQL also enabled me to transfer any data from the database to the application with little to no trouble which meant that any changes I did need to make to the database would be very quick and then any new data would be show quickly in the application which also meant that I did not have to waste too much time with fixing and sorting any data I would use. Additionally, I was more confident implementing MySQL as I have a little bit of experience with it before and I was not aware of too many other data storage solutions I could use, so this ultimately meant that I didn't have much more choice in the data storage solutions I could have used. If I had used any other storage solution it would have meant that I would need to learn and understand how it worked and how it could be implemented into the application, all of which would mean that I would have been spending too much time trying to figure out what to do and therefore not develop the prototype as efficiently or effectively as I have done.

## 4. Design

When it came to the initial designing phase of this project, I decided to look at a variety of different youth service applications and websites to get a very good idea of what the common features were and to see how I could make a coherent and easy to navigate application for anyone of any age. The initial designs for this application were inspired mostly by the CYS's existing website as I wanted to make sure that it would have a similar feel to their aesthetic. As I got through different iterations of the design, I wanted to ensure that it did not change too drastically to the point where it would no longer feel like it is bespoke for the CYS, However, the final design I ended up using means that it has the possibility to be altered and therefore used for other youth services if they needed an application like this.

### 4.1 Initial Wireframe and User Interface Drafts

#### 4.1.1 Wireframes

The first drafts of the wireframe designs for the application were created by using Draw.io where I was able to create the individual pages of what the application would look like. Doing this allowed me to pinpoint specific areas on each of the pages that I wanted to edit or alter entirely. Draw.io was also easy to use which meant that I didn't need to spend a long time on creating the first drafts of the application, it also meant that I was able to easily copy elements from one design that I liked and then add it to a newer design, this meant that new iterations could be made within minutes of a previous one and therefore speed up the designing process of development and I would not have to waste too much time trying to figure out changes that could be made. These initial wireframe designs were essentially used for me to get a grasp on what a basic level scheduling system would look like and to get a feel of how easily someone would be able to navigate such a design. I also used these designs to compare it to how other applications look and function to see if I



could improve on my original concepts. This proved extremely useful to me as it meant that I was able to work on the wireframe designs very quickly and not spend too much time in my weekly planned schedule on those areas that could potentially be very tedious.

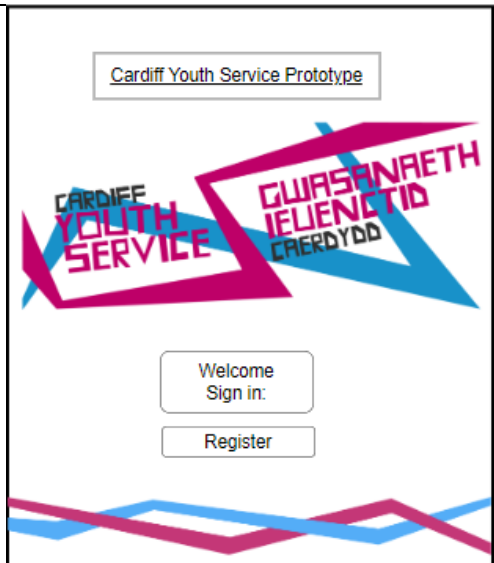
#### 4.1.2 User Interface Designs

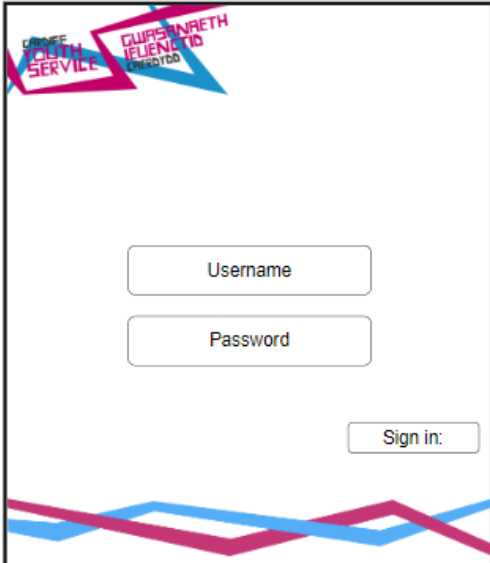
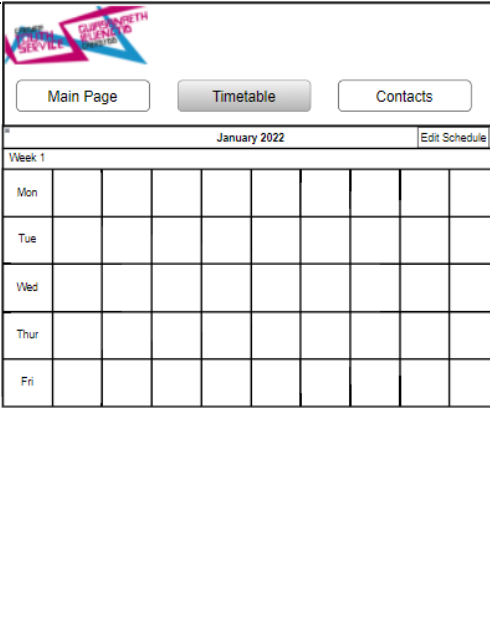
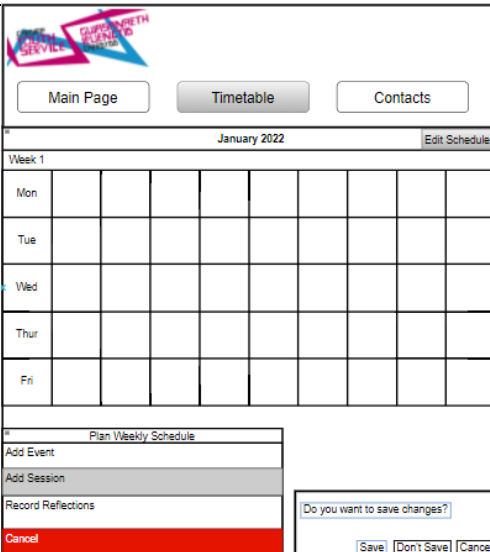
When I held my first meeting with CYS, I had created an example UI (user interface) design and wireframe design as mentioned above, so that they could understand what I had already researched and done and what I had hoped they would like the design of the application to roughly be like. The feedback I received from the senior youth officer was good and they appreciated that I incorporated a lot of design elements that makes it easily recognisable as something CYS would use. For this I made sure that I included the same colours that can be found on their website, and I also made sure to include their already existing logo (Cardiff Youth Services, 2021). The feedback I received also let me know that they liked the simplistic and easy to navigate design I was going for. For this I decided it would be a good idea to use certain principles from Nielsen's Heuristics (Nielsen, 2020) which included a minimalist design and user control and freedom, meaning that a user has full access and control over an application when navigating it and that nothing is restricted from them. Having these principles in mind I was able to make a UI design that meant that I was able to keep functionality and overall design user friendly enough that anybody should feel comfortable using the system and that they should not get confused with any of the functionality of the page.

#### 4.2 Wireframe Designs and User Interface designs

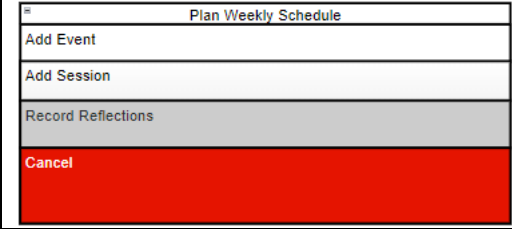
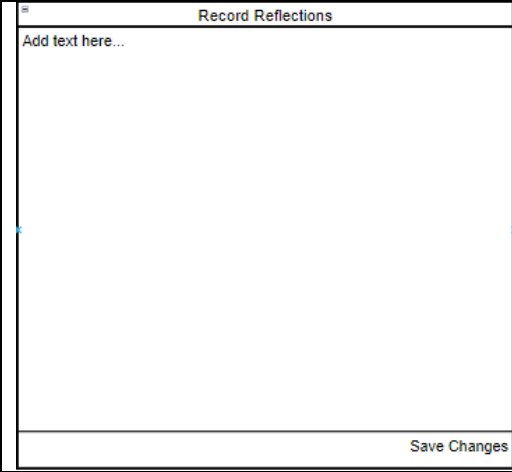
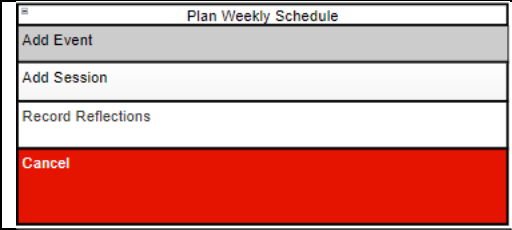
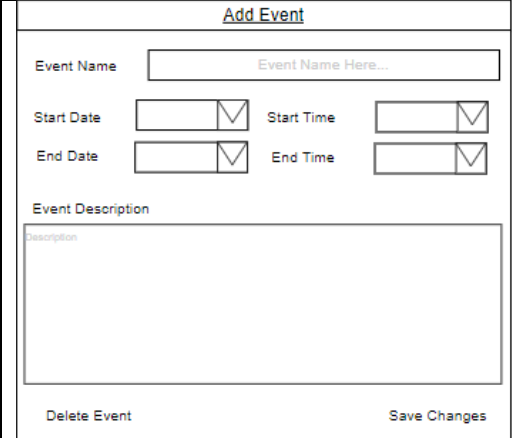
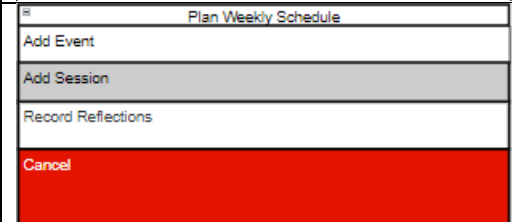
Below you will find several tables with all my final wireframe and User Interface designs I decided to use for the final product.

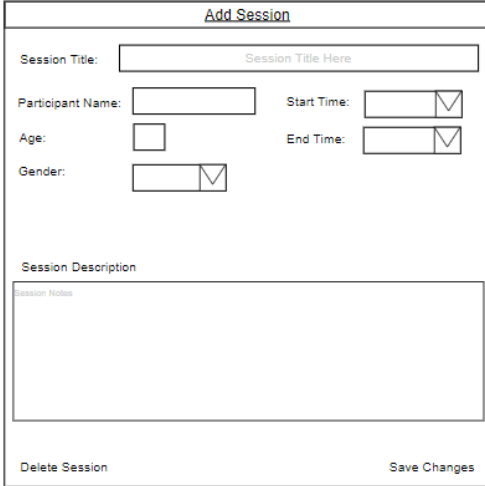
##### 4.2.1 Wireframe

Wireframe	Description
	<p>Sign in and register page: This is the final wireframe design I created on Draw.io for the prototype. This page shows that you should be able to sign in or register to make an account.</p> <p>Pressing the sign in button will take you to the login page where you will be able to enter your username and password.</p> <p>The register button was initially put in place to allow anyone to create an account for the system, however this was difficult to implement into the final product so I unfortunately had to remove this feature until future development can be made to the application.</p>

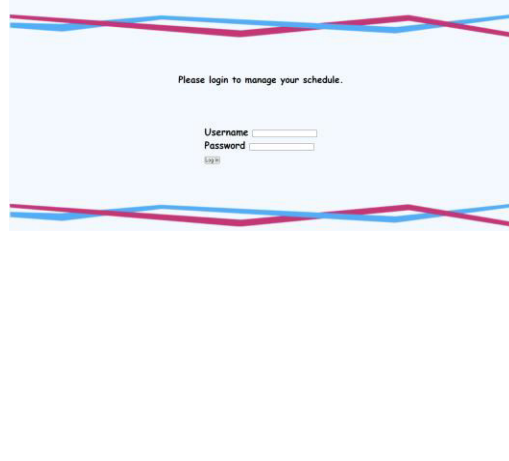


 <p>The login page features a header with the 'GURPREETH REFLECTIONS SERVICE' logo. It contains two input fields labeled 'Username' and 'Password', and a 'Sign in:' button at the bottom right. The page is decorated with a colorful abstract graphic at the bottom.</p>	<p><b>Login page:</b> This is the next page the user gets sent to after pressing the sign in button.</p> <p>It allows the user to enter their username/email into the username field and password into the password field.</p> <p>Pressing sign in after that will take you to the main page which contains the calendar feature of the final product.</p>
 <p>The main page has a header with the same logo and three navigation buttons: 'Main Page', 'Timetable', and 'Contacts'. Below the buttons is a section for 'January 2022' with an 'Edit Schedule' link. It features a weekly calendar grid for 'Week 1' with days from Monday to Friday, each with a corresponding row of time slots.</p>	<p><b>Main Page:</b> The main page of the final product acts as the central hub where users will be able to look at their calendar and edit their schedule.</p> <p>In order to stay within the principles laid out by Nielsen, I made sure not to add too many tabs to the page design, make everything clear and easy to read, and easy to navigate through.</p> <p>The overall design also meant that it is easily transferable to other devices such as mobiles and tablets as the screen would be able to adjust to the format of those technologies. This was important to include as it means that any user will be able to use the application on any device wherever they are.</p>
 <p>This screenshot shows the main page with the 'Edit Schedule' link highlighted. Below the calendar grid, there is a 'Plan Weekly Schedule' section with buttons for 'Add Event', 'Add Session', and 'Record Reflections'. A red 'Cancel' button is at the bottom left. A modal dialog box is open at the bottom right asking 'Do you want to save changes?' with 'Save', 'Don't Save', and 'Cancel' options.</p>	<p><b>Main Page / Edit Schedule enabled:</b> This is the same main page as the one shown before, however in this image, you can see that the user has selected to edit their schedule.</p> <p>Pressing "Edit Schedule" would bring up a drop-down selection menu that the user will be able to add sessions, events, and record reflections they had in those sessions.</p> <p>The drop-down menu was decided so that the user could easily see what they were doing when editing their schedule for the week.</p>

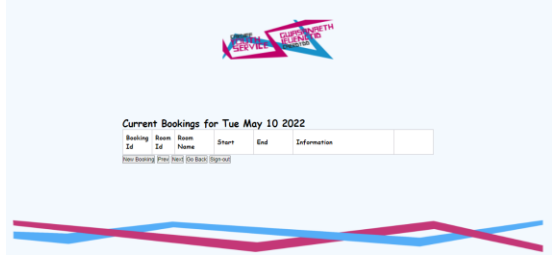

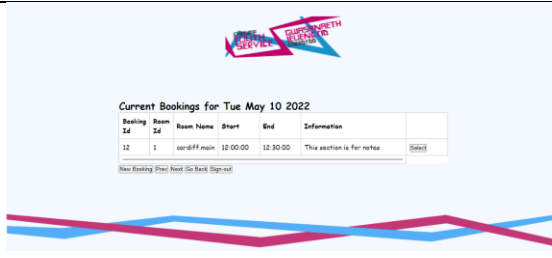


	<p>Drop-down menu:</p> <p>This is the drop-down menu that appears when the user selects edit schedule.</p> <p>It allows the user to select one of the options on the menu and then edit the schedule.</p>
	<p>Record Reflections option:</p> <p>The record reflections option will bring up a new screen on the page. The screen is a box that allows the user to enter text into a text field.</p> <p>The user is able to save the changes made to the Record Reflections section and that will then be visible to see for the user.</p> <p>There should be a delete button, but as I have changed a significant amount in the final product, it was no longer needed.</p>
	<p>Drop-down menu (again):</p> <p>The same drop-down menu as before, nothing is different with functionality or design.</p>
	<p>Add Event option:</p> <p>The Add Event option works similarly to Record reflections option as it brings up another screen which is a box containing options and text fields.</p> <p>The additional options allow the user to add a start date and time, and an end date and time.</p> <p>This screen is used so that the user is able to make an event on their schedule and write an event title and description so that everyone knows what it is.</p>
	<p>Drop-down menu (again):</p> <p>The same drop-down menu as before, nothing is different with functionality or design.</p>

 <p>The 'Add Session' form includes fields for Session Title, Participant Name, Age, Gender, Start Time, and End Time. It also has a large text area for Session Description and buttons for 'Delete Session' and 'Save Changes'.</p>	<p><b>Add Session option:</b></p> <p>The Add Session option is more or less the same as the Add Event option as it brings up another screen which is a box containing options and text fields.</p> <p>The additional options allow the user to add a start date and time, and an end date and time. They can now also add the participants name, age, and gender if they allow the user to add that information.</p> <p>This screen is used so that the user is able to make a session on their schedule and write an event title and description so that everyone knows what it is.</p>
--	--

#### 4.2.2 User Interface

 <p>The login page features a light blue background with a central text prompt 'Please login to manage your schedule.' Below this are input fields for 'Username' and 'Password', followed by a 'Login' button.</p>	<p><b>Login Page:</b></p> <p>This is the login page on the final prototype user interface.</p> <p>The page allows the user to enter their username and password and the progress on through the application.</p> <p>Unfortunately, I was unable to get a register option working properly as it was unable to save the user's data properly and it would cause issues trying to get into the application altogether.</p>
 <p>The user profile page displays the text 'test1', 'test1@test.com', 'First1', and 'Last1'. At the bottom, there are two buttons: 'Bookings' and 'Sign-out'.</p>	<p>This is the following page after a user enters. Ideally it will show the user's first and last name, their email address and username, however, since I could not get a registration system working, it means that there wasn't enough information that would get passed through and it wouldn't fill in all the required fields so until there is further work done to the prototype this will have to be used as a bit of a stand in.</p>
 <p>The main page features a calendar for May 2022. The calendar is a grid showing days of the month. Above the calendar, there are some decorative elements and a 'Dark Mode' toggle switch at the bottom right.</p>	<p><b>Main Page / Calendar:</b></p> <p>This is the main page of the calendar and I managed to keep the design largely the same as the prototype with a few adjustments made.</p> <p>The calendar works by allowing the user to double-click on any day they want, and it will then proceed to take them to the bookings page.</p> <p>I kept the design largely simplistic and easy to</p>

	<p>navigate as it means that the user will be able to easily use the application how it is intended to be used and it also means that they won't get confused with too many functions all being on the same page.</p>
	<p><b>View Bookings Page:</b> The View Bookings Page appears when a user double clicks on a day of the week they want.</p> <p>It will take the user to this page where they will be able to see all of their bookings, whether that be sessions or events.</p> <p>Here the user will be able to add a new booking, delete any booking and sign out of the application as a whole. The edit button became very complicated when trying to implement the functionality so the user for now will have to add and delete bookings altogether instead of editing the ones they have.</p>
	<p><b>New Booking Page:</b> This is the creating a bookings page that allows the user to book in a slot in any given room.</p> <p>It allows them to change the slot in the day the session or event will be taking place on.</p> <p>It will also allow the user to add any notes they want for the session or event. Whether that is the event description and name, or the session reflection notes.</p>
	<p><b>View Bookings Page (again):</b> This is the same View Bookings Page as shown previously.</p> <p>The page now displays any bookings that the user has made and the notes / information the user has added about the event or session they have booked.</p>

### 4.2.3 Wireframe Strengths and Weaknesses

The biggest strength of my wireframe design for the CYV scheduling system prototype is with the overall consistent design and layout choices I have made for each of the pages, and the minimalistic feature design I have implemented into the wireframe. I believe that the reason why the consistency of the design layout for each of the pages is one of the biggest strengths for the wireframe is because having a sense of familiarity when using the system would be very important when using it on different devices. Consequently, the consistency means that a user will not get confused when they switch from one device to another and they will be able to just as easily navigate it on one platform to another. Moreover, having a minimalistic design within the wireframe helps to make the system accessible to anyone who wishes to use it which also assists with maintaining consistency

across all platforms as they would share all the features as the base system and there would not be any deviation from this. Another strength for my wireframe was the colour scheme. The colour scheme I chose to include has colours which are easily identifiable to anyone, and I was aware that there would be a possibility that someone with a colour blind disability could potentially use the system too, so taking this into consideration I opted to use colours that are less frequently associated with colour blindness, which meant not using colours such as greens, reds, oranges, yellows, and browns, and therefore allowed me to use colours such as the blues and purples that can be found on the wireframe.

One weakness that I have identified with the wireframe design was the use of the drop-down menu. Whilst initially it seemed as though it would be a good idea as it is a feature commonly seen amongst other websites, I realise that some people have issues controlling their mouse or mobile device screen effectively, which created an issue of how easily someone would be able to use the drop-down menu. Some websites that include a drop-down menu rely on the user to hover over the tab that contains the drop-down menu which can be an issue as some users may not fully understand how the mouse on a laptop or other PC system works properly and therefore may only be able to get the menu to appear very briefly which means they are unlikely to select an option on the menu. Similarly, when using a mobile device, some users have difficulties pressing on small tabs and options boxes in menus, so I ultimately decided to get rid of the drop-down menu, which can be seen in the final UI design in the above table, instead opting for a function that would allow the user to simply select a date and for that to take them to a separate page, whilst this did mean that it may affect a user's efficiency when using the system, I believe that it would be best to remove the drop-down menu in favour of comfort when using the system. Another weakness that I identified was the overall lack of error prevention when editing a user's schedule, which was identified in both the final wireframe design and final UI design. I did attempt to resolve this by implementing a warning system that would alert the user when they are about to delete or edit a booking that they have made, however, I was unable to effectively implement this system and therefore decided to remove it to the final product as it created more issues than were necessary. This is an area I would like to see improved upon in any form of future works done to the system as it would make the system more effective to use and it would mean that users are less likely to make those mistakes in the future.

During the development of the design phase for the project, I did hold numerous meetings with the senior youth officer (at CYS) to get as much feedback as possible on the design as I developed it. The feedback received provided me with a greater understanding of what they were happy with in the design so far and if there were any changes or things that needed to be removed entirely from the wireframe design. One instance of this was the idea to add CYS's main logo to the system's pages, so that the system would have a clear link to the organisation it was made for and so that it was easily identifiable to anyone that this system is used for CYS. This was a design choice that was changed almost immediately of the design phase, and it was also simple enough to implement to the final product as well.

#### 4.3 Design Colour Scheme:

Using my final wireframe design and after looking into CYS's website, it became very apparent of the colour scheme that I would need to implement into the final product and UI design. CYS's website is a bright and colourful website that easily draws the user attention with a lot of sharp and vibrant colours, so I wanted to use this as my main source of inspiration for the overall UI design of the final product. I also took inspiration from other previously mentioned applications such as Power Apps for how the UI should look and feel in general when looking at scheduling systems.

There are two primary colours on the UI design that I have decided to use. The first colour being a bright purple that leads into being magenta. This colour was a very bold look on CYS's website, and I used the same banners and logos on my system so that a user would have a sense of familiarity and would be drawn into the web page. The bright purple colour is usually associated with harmony and emotional balance (Color Column: Magenta, 2022) which was what I believed to be a very important and key factor of all youth services, to help younger people to find their own inner harmony and emotional balance as they develop through their younger years into their adult years. This colour stands out on the page and can be seen on the calendar feature of the main page as it shows what the current date is, the colour magenta was easily one of the best decisions to include in the final UI design as it is a soft colour that makes a user feel more comfortable and it also makes the experience of using the application far more enjoyable. The other main colour on the page was close to primary blue. Most blue colours are generally perceived to be non-threatening and associated with calmness and tranquillity (Cherry, 2020). This is another example of something I believe to be very important to all youth services as they aim to make younger people feel calm and peaceful as they want to help those younger people develop into their adult lives with the best personal, social, and educational development and skills. The primary blue also stands out on the main page which features a blueish grey background colour for the remainder of the page which helps to draw the user's attention to the primary feature of the page they are on.

Another reason why I decided to use these colours on the final product's UI design was because of the dark mode I had implemented into the system. These two main colours I picked were still very easy to see on the dark mode version of the web page which was crucial as it still draws the user's attention to the page. These colours also helped the user when navigating through the dark mode as the sense of familiarity meant that the user would still be able to identify where everything was without getting lost as the screen had changed colour.

## 5. System Specification

### 5.1 Functional Requirements

#### 5.1.1 Must Have

- Requirement 1: The application must allow the user to add session bookings when they have selected a date.

Acceptance Criteria:

- User can double click to select date and be taken to view bookings page
- User can select new booking and be taken to new bookings page, then edit the session booking they want to have
- User is able to save or delete session booking and then be taken back to view bookings page

- Requirement 2: The application must allow the user to add event bookings when they have selected a date.

Acceptance Criteria:

- User can double click to select date and be taken to view bookings page
- User can select new booking and be taken to new bookings page, then edit the event booking they want to have
- User is able to save or delete event booking and then be taken back to view bookings page

- Requirement 3: The application must allow the user to record reflections / make notes when they have created a booking.

Acceptance Criteria:

- User can select the text field provided
- User can enter any text they wish and to be able to save that information
- User is able to see the notes or recorded reflections when they are taken back to the view bookings page

- Requirement 4: The application must allow the user to select a date on the calendar.

Acceptance Criteria:

- User can double click to select any date in any month or year they want
- User can be taken to view bookings page with the correct date select showing in the information above the fields the user can use
- User is able to save any bookings on that date and then be able to see all bookings on said date whenever they look back on it

#### 5.1.2 Should Have

- Requirement 5: The application should allow the user to edit their schedule for both sessions and events.

Acceptance Criteria:

- User can double click a selected date and be taken to view bookings page
- User can select any previously made booking and be taken to new bookings page, then edit the session or event booking they want to have
- User is able to save any edits to the session or event booking and then be taken back to view bookings page
- User can see the edits made to the session booking

- Requirement 6: The application should allow the user login or register into the application

Acceptance Criteria:

- User can enter their login details on the sign in page or user can create new account
- User can be sent to the login confirmation page
- User can either go to the schedule page or sign out

#### 5.1.3 Could Have

- Requirement 7: The application could allow the user to delete their account on the application.

Acceptance Criteria:

- User can enter their login details on the sign in page or user can create new account
- User can see their account info and select to delete their account and any information and data that is related to the account on the database

- Requirement 8: The application could allow the user to edit their account information.

Acceptance Criteria:

- User can enter their login details on the sign in page or user can create new account

- User can see their account info and select to edit their username, password, personal info and contact information

## 5.2 Non-functional Requirements

### 5.2.1 Must Have

- Requirement 1: The application will be web-accessible on any device.

Acceptance Criteria:

- Users can access the application either on laptop / PC setup and any other mobile device
- All features of the application will be fully functional on any device
- Users can still see and access any and all bookings they have made

- Requirement 2: The application will be easy to navigate with easy-to-read design and easy to use functionality to minimise any confusion.

Acceptance Criteria:

- User can easily navigate and use all functions on the application without any help from outside sources
- Design must be easy to understand to all users
- Functions must be simple and quick for users to execute
- Users can find and understand all functions on the application

- Requirement 3: All features of the final product must work on laptop, PC setup, and mobile devices.

Acceptance Criteria:

- User can easily access the application on all platforms, e.g., laptop or mobile device
- The user can use the same features on the mobile device as they can on a laptop
- The features will work just as quickly on mobile devices as they do on laptops

### 5.2.2 Should Have

- Requirement 4: The application should provide users with help and instructions through the use of guides from other resources provided

Acceptance Criteria:

- User can read the instructions clearly and understand the step-by-step processes
- The user can access the other resources easily

- Requirement 5: The final version of the prototype should be in a satisfactory state where future development would be easy to accomplish

Acceptance Criteria:

- Future users will be able to easily access all files and code without any issues
- All code will be written with little to no errors, bugs, or warnings to allow for easy development at a later date
- Code is formatted correctly and easy to read so that all developers can understand how all current features work.

### 5.2.3 Could Have

- Requirement 5: The final version of the prototype could have guidelines in place to aid future developers work on the application.

Acceptance Criteria:

- A guide can be included within the code or in a separate format that aids future developers when working on the application
- Code can be commented on so that developers can see what certain functionality is for

### 5.3 Use Cases

Below you will find a Use Case Diagram to highlight how a user may go about using the application and all of its features. All of the use cases shown in the diagram and in the tables below that are from the “Must Have” section of the functional requirements as these were the tasks and features CYS had set out in the brief they gave me and this to meet those requirements.

#### 5.3.1 Use Case Diagram

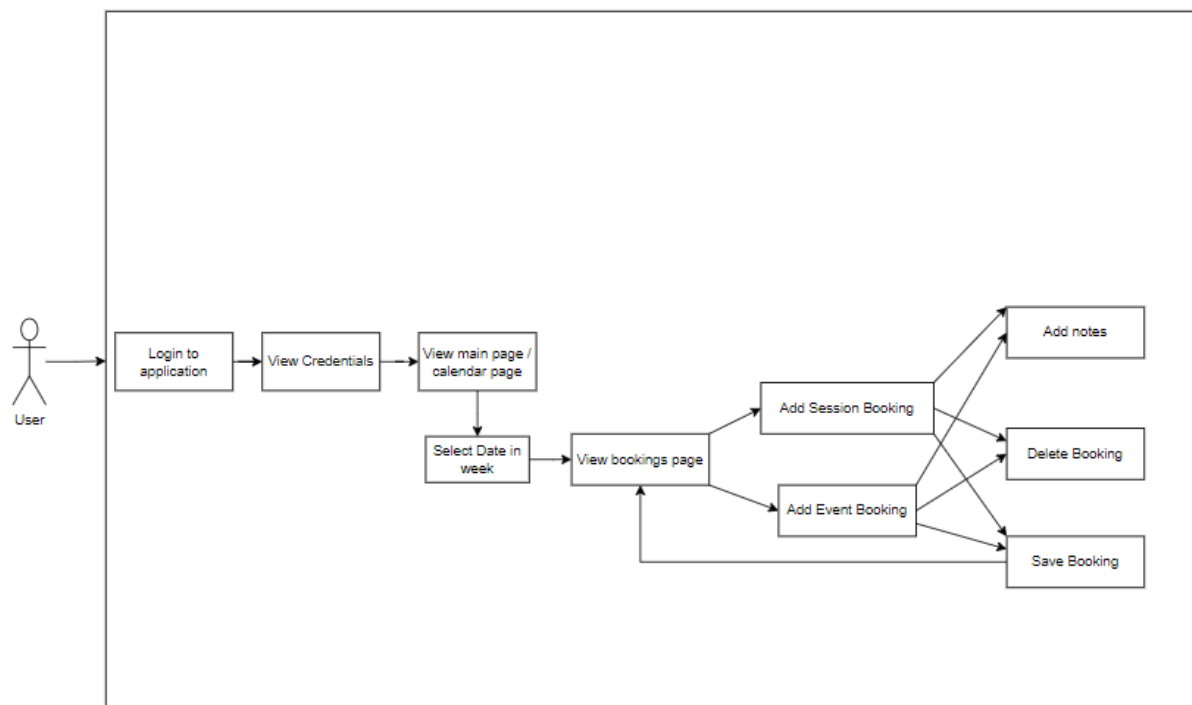


Figure 2 – Use Case Diagram demonstrating how a user would use the application and all of its features

#### 5.3.2 Use Case Tables

This is the Use Case Table section where it shows deeper detail on each of the use cases shown on the Use Case Diagram. For this section I am showing how the user will go about navigating the application and all of its features.

##### 5.3.2.1 Use Case Table 1

Use Case:	The application should allow the user to login
Goal:	The user is able to login to the application and then view their credentials on the following page.
Basic Flow:	1. The user logs onto the application when they have filled in their username and password



	<ol style="list-style-type: none"> <li>2. The user is shown their credentials</li> <li>3. The user is then directed to the main page</li> </ol>
Alternative Flow:	No Alternative route possible for this use case

5.3.2.2 Use Case Table 2

Use Case:	The application must allow the user to select any date on the calendar, on any week, month, and year.
Goal:	The user is able to go on the calendar to select any date they want from any month and any year. It allows them to go from this to the view bookings page so they can edit their schedule.
Basic Flow:	<ol style="list-style-type: none"> <li>1. The user logs in enters their username and password and is taken to view their credentials</li> <li>2. User selects bookings on the view credentials page and is directed to the calendar page</li> <li>3. The user selects a date from the week shown on the calendar</li> <li>4. The user is taken to the view bookings page</li> </ol>
Alternative Flow:	<ol style="list-style-type: none"> <li>1a. The user presses the sign out button and is taken back to the login page</li> <li>3a. The user selects the months tab</li> <li>3b. The user can select any month they want</li> <li>3c. The user is shown the new month on the calendar and selects new date</li> <li>3a.i. The user selects the change year button</li> <li>3a.ii. The calendar changes the year</li> <li>3.a.iii. The user can see the new year and can then select a date</li> </ol>

5.3.2.3 Use Case Table 3

Use Case:	The application must let the user make a new booking whether that is an event or session
Goal:	The user is able to select any given date on the calendar and is taken to the view bookings page which will allow them to create a new booking which can either be a session or an event.
Basic Flow:	<ol style="list-style-type: none"> <li>1. The user logs in to the application and is taken through the credential page and then directed to the main page with the calendar</li> <li>2. The user selects a date on the calendar</li> <li>3. The user selects new booking</li> <li>4. The user is directed to the new bookings page where they can select which room they would like to use</li> <li>5. The user selects what slot in the day they would like to set the booking for</li> <li>6. The user selects save booking</li> </ol>
Alternative Flow:	<ol style="list-style-type: none"> <li>3a. The user selects go back and is then taken back to the main Page</li> <li>3b. The user selects delete booking and the application removes any booking that has been saved there</li> </ol>

#### 5.3.2.4 Use Case Table 4

Use Case:	The application must allow the user to enter notes / record reflections on any session or event.
Goal:	The user is able to add any notes / record reflections on any bookings that they make and for that information to then be displayed.
Basic Flow:	<ol style="list-style-type: none"><li>1. The user logs in to the application and is taken through the credential page and then directed to the main page with the calendar</li><li>2. The user selects a date on the calendar</li><li>3. The user selects new booking</li><li>4. The user is directed to the same new bookings page where they can create a session or event</li><li>5. The user then enters any information they want into the text field provided.</li><li>6. The user selects save booking</li></ol>
Alternative Flow:	<ol style="list-style-type: none"><li>3a. The user selects go back and is then taken back to the main Page</li><li>3b. The user selects delete booking and the application removes any booking that has been saved there</li></ol>

#### 5.3.2.5 Use Case Table 5

Use Case:	The application must let the user delete any bookings made
Goal:	The user is able to delete any booking they want when on the view bookings page of the application. The application must allow the user to delete any booking that they want from their own schedule.
Basic Flow:	<ol style="list-style-type: none"><li>1. The user logs in to the application and is taken through the credential page and then directed to the main page with the calendar</li><li>2. The user selects a date on the calendar</li><li>3. The user selects new booking</li><li>4. The user is directed to the same new bookings page where they can create a session or event</li><li>5. The user is able to create as many bookings as they want.</li><li>6. The user is then able to select any previously made booking and delete them from their schedule.</li></ol>
Alternative Flow:	<ol style="list-style-type: none"><li>3a. The user selects go back and is then taken back to the main Page</li></ol>

### 5.4 UML Class Diagram

The following figure is a UML diagram I have created also using Draw.io that is used to portray the basic structure of the prototype and to show all of the functional classes that must be there in order to pass all the functional requirements.

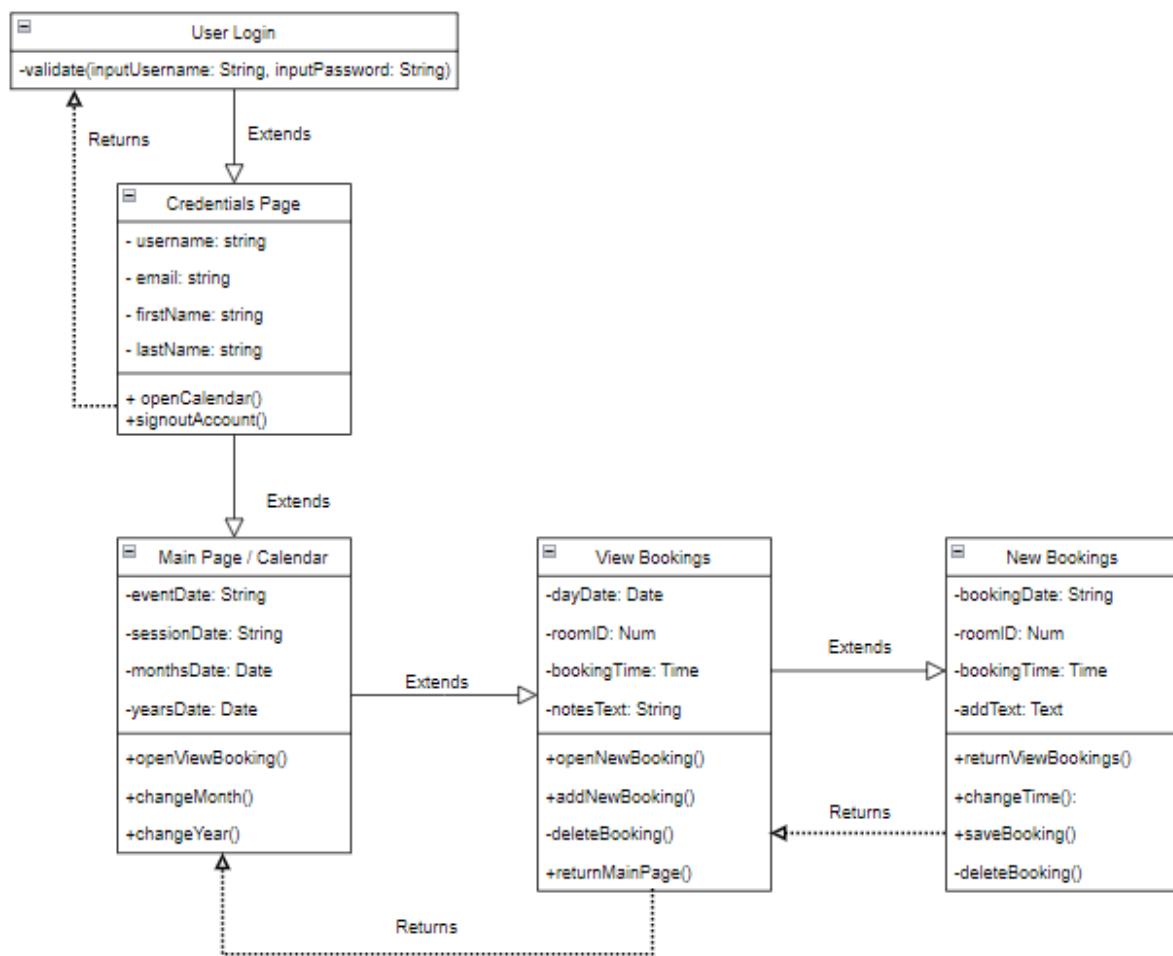


Figure 3: UML class diagram to illustrate how a user would go through the application

#### 5.4.1 UML Class Diagram Description

The first class that shows the basic structure of my application is “User Login”. This class is where the user is able to log into the application and then get directed to the rest of the prototype. This class then directs the user into the next class which is the “Credentials Page” which is the page that appears before allowing the user to go onto the main calendar page. This page will show the user their full credentials so, username, email, first and last name as a form of verifying the user and showing the user if these are the correct credentials.

Going through those first two classes will then extend and take the user to the primary class of the CYS scheduling app which is the “Main Page / Calendar”. This class acts as the main hub that the user will navigate through in order to get to the other classes that will allow them to edit their week-to-week schedule. The “Main Page / Calendar” class also extends then into the “View Booking” class. The “View Booking” class is the class where the user will be able to make complete changes to the overall schedule for their week by being able to add a new booking or deleting any previous bookings that have been added to the schedule. The user from this class will be able to get extended onto the “New Booking” class which is where the user is able to edit the specific details of the bookings being made. These details include the actual time of the booking, the room the booking will take place in, any notes / information on the booking being made. The “New Booking” class doesn’t extend any further as when the user saves a booking or deletes a booking it automatically returns them to the “View Booking” class. The “New Booking” class allows users to make bookings for both sessions and events as since these are roughly the same, there wasn’t any changes needed.

Whilst the UML class diagram is relatively small, this was partly the intention as I only needed to make a scheduling system with limited functionality that would work for CYS, adding more classes would only mean that I would be creating unnecessary features and functionality in the final product which could have complicated a lot of areas of development which would have slowed me down significantly.

Using the Use Cases and UML diagram from this section I was able to fully implement most of these classes from the UML Class Diagram and functional requirements highlighted previously. This allowed me to develop the UI design and functionality of the final product with relative ease as I was able to keep most of functional and non-functional requirements simple enough as this application didn't need too many features for CYS.

## 6. Implementation

This next section of the report is to show how I implemented the different frameworks into my final prototype for CYS. I will show the key components of each of the frameworks, how and why they work the way they do.

### 6.1 Database Design and Storage solution

Using JSON and C# primarily for the back-end portion of the programming, I decided it would be best if I used a data storage solution that was compatible with these frameworks and that I knew would be efficient and secure enough to hold a basic amount of data in there for the time being. MySQL unfortunately wasn't the best choice in storage solution for me as I still had a lot of learning to do when implementing it and in general I wasn't always 100% certain with what I was doing when it came to storage solutions, showing that it is still an area for me to improve upon, however, it did the job, and I was able to make the application store a basic level of data that would be needed to be used when passing through certain parameters in the application. The figures below show the solution I came up with when it came to adding users and allowing them certain privileges when using MySQL as I wanted them to be able to have a better level of control over the application as a whole.

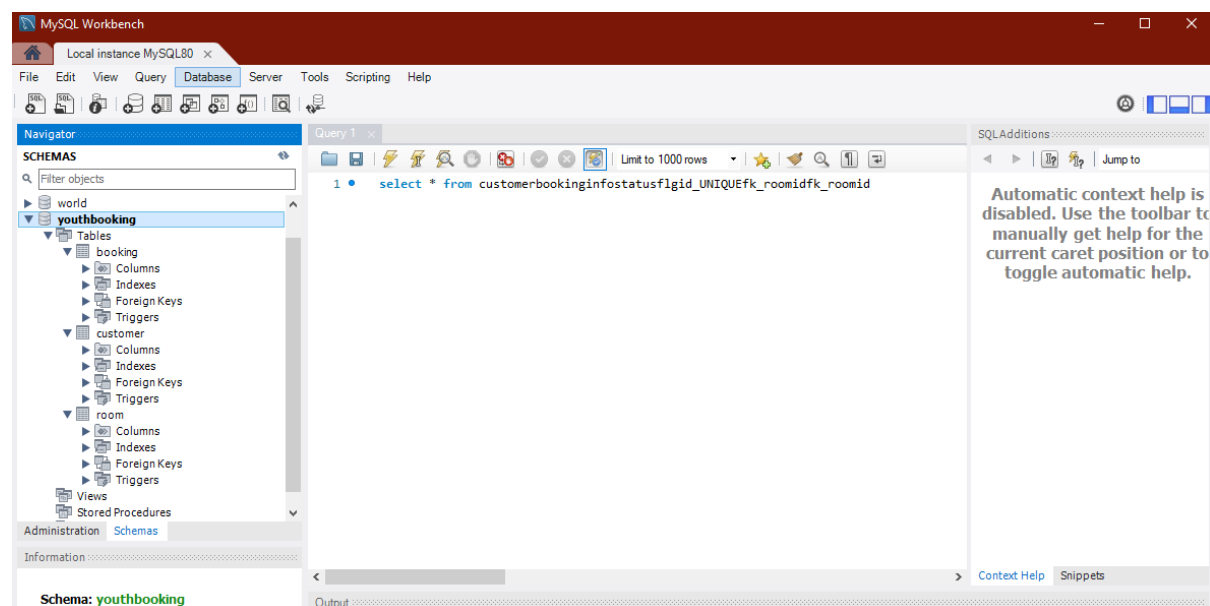


Figure 4 MySQL solution for storage and database design

## 6.2 Web Development

Since I was developing an application for Cardiff Youth Services, I had to ensure that I was developing this prototype by using the industry standards best practices. Since this was for a youth service and could potentially be implemented by other youth services, I had to be certain that all the code I was producing for the application was going to be maintainable by future developers if they wanted to do any sort of future work to it.

To ensure that I was following the best industry practices I made sure that I had laid out a clear and concise plan when it came to developing the application. Since I had been given requirements that I was expected to meet by CYS, I had to plan how I would make each framework work to best meet these requirements. In order to help make this as maintainable as possible for CYS I also had to make sure that all of the different frameworks were connecting and that nothing was being stored in any folders outside my main solution folder where all of the HTML pages had been stored. This was to make sure that it was easy for CYS and any future developers to find all of the code needed to make the application work. The image below shows the folder I managed to keep everything stored in. The below figure shows my folders I used to contain all of the frameworks and code in.

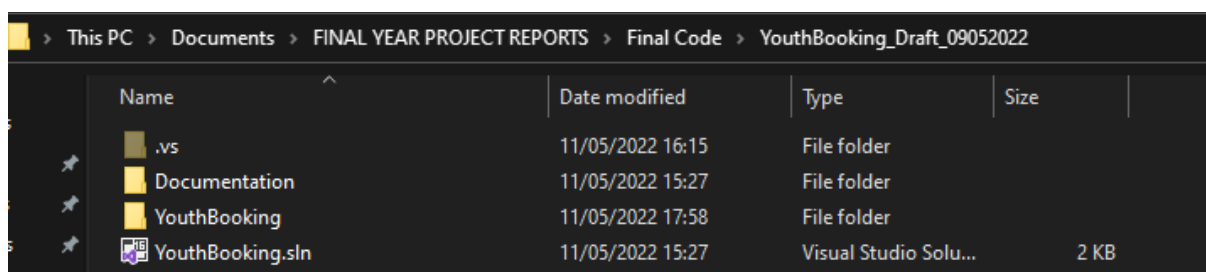


Figure 5: Folder storage of all code

As I was using the agile methodology, I was able to break down all of the programming into smaller chunks that would allow me to make any improvements to small areas without causing any bigger errors and bugs.

## 6.3 Framework Implementation

Implementing everything for the final product did prove to be a challenge so in this section I will be showing a number of figures to explain how and why they have been implemented into the program. These figures will show the key areas of functionality in the final program.

```

http://json.schemastore.org/launchsettings.json
{
  "$schema": "http://json.schemastore.org/launchsettings.json",
  "iisSettings": {
    "windowsAuthentication": false,
    "anonymousAuthentication": true,
    "iisExpress": {
      "applicationUrl": "http://localhost:51904",
      "sslPort": 44378
    }
  },
  "profiles": {
    "IIS Express": {
      "commandName": "IISExpress",
      "launchBrowser": true,
      "launchUrl": "swagger",
      "environmentVariables": {
        "ASPNETCORE_ENVIRONMENT": "Development"
      }
    },
    "YouthBooking": {
      "commandName": "Project",
      "dotnetRunMessages": "true",
      "launchBrowser": true,
      "launchUrl": "swagger",
      "applicationUrl": "https://localhost:5001;http://localhost:5000",
      "environmentVariables": {
        "ASPNETCORE_ENVIRONMENT": "Development"
      }
    }
  }
}

```

Figure 6: JSON implementation

This first figure shows how I have implemented JSON into the application. Here you can see that the primary reason why JSON has been implemented into the code is to allow Swagger to work since I have made use of APIs in the programming. JSON was used along with swagger in order to add an additional layer of authentication so that not simply anybody could access the application and prototype. You can see in a number of the lines of programming that 'launchBrowser' has been used so that when the application has been launched on Visual Studio, the user will be taken directly to a new tab on their browser which will show the Swagger interface. There is the additional option to turn off Swagger altogether by simply adjusting 'launchBrowser: true' to false. This will stop swagger from working and the user should still be able to fully use the application.

```

U references
public BookingController(IConfiguration configuration)
{
    Configuration = configuration;
    var server = Configuration["MySQLConnection:Server"];
    var userid = Configuration["MySQLConnection:UserId"];
    var password = Configuration["MySQLConnection:Password"];
    var databasename = Configuration["MySQLConnection:Database"];
    _connectionString = $"server={server};userid={userid};password={password};database={databasename}";
}

```

Figure 7: C# implementation.

In the figure shown above we can see an example of why the C# was best implemented as an additional framework when building this prototype. C# was used in order to create controllers for the application that could allow certain data to be handled from MySQL. Essentially it has been used in order to determine what response to send back to the user (MySQL in this case) when a specific browser request has been made by a user on the application. There have been multiple controllers implemented with C# in order to make sure that all of the browser requests that can be predicted by the application user, can be properly handled.

```
// CHECK LEAP YEAR
isLeapYear = (year) => {
  return (year % 4 === 0 && year % 100 !== 0 && year % 400 !== 0) || (year % 100 === 0 && year % 400 === 0)
}

getFebDays = (year) => {
  return isLeapYear(year) ? 29 : 28
}

let calendar = document.querySelector('.calendar')

const month_names = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']

let month_picker = document.querySelector('#month-picker')

month_picker.onclick = () => {
  month_list.classList.add('show')
}

update_current_date = (chosen_ybdate) => {
  var storedData = sessionStorage.getItem("customer_session_data");
  if (storedData) {
    logged_customer_data = JSON.parse(storedData);
    logged_customer_data.cursor_fromdate = chosen_ybdate;
    sessionStorage.setItem("customer_session_data", JSON.stringify(logged_customer_data));
  }
}

date_to_ybdate = (ainput) => {
  // return ainput.toISOString()
  //return ainput.toISOString().slice(0, 19).replace('T', ' ');
  return ainput.toDateString();
}
```

Figure 8: JavaScript Implementation 1

This figure shown above shows that the JavaScript framework was partly implemented in order to check for a leap year in the calendar. This was to ensure that the program would be able to handle real-world dates and years so that the future development of the application wouldn't have any mistakes when there was a leap year. This can be a very easy thing to miss as it isn't always at the top of any developer's mind when trying to make a calendar.

```

generateCalendar = (imonth, year) => {
  let calendar_days = document.querySelector('.calendar-days')
  calendar_days.textContent = ''
  calendar_days.innerHTML = ''

  let calendar_header_year = document.querySelector('#year')

  let days_of_month = [31, getFebDays(year), 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]

  let currDate = new Date()

  month_picker.innerHTML = month_names[imonth]
  calendar_header_year.innerHTML = year

  // javascript date takes the month by 0-index and not the real month
  let first_day = new Date(year, imonth, 1)

  for (let i = 0; i <= days_of_month[imonth] + first_day.getDay() - 1; i++) {
    let day = document.createElement('div')
    if (i >= first_day.getDay()) {
      var loopDate = new Date(first_day);
      loopDate.setDate(loopDate.getDate() + i - first_day.getDay());
      day.classList.add('calendar-day-hover')
      day.id = date_to_ybdate(loopDate)
      day.addEventListener('dblclick', function (e) {
        var xtarget = e.target
        var fromXDateParam = xtarget.id
        //alert(fromXDateParam)
        update_current_date(fromXDateParam)
        location.replace("daybookings.html")
      }, false);
      day.innerHTML = i - first_day.getDay() + 1
      day.innerHTML += ' <span></span><span></span><span></span><span></span><span></span>'
      if (i - first_day.getDay() + 1 === currDate.getDate() && year === currDate.getFullYear() && imonth === currDate.getMonth()) {
        day.classList.add('curr-date')
      }
    }
    calendar_days.appendChild(day)
  }
}

```

Figure 9: JavaScript Calendar generation

```

let month_list = calendar.querySelector('.month-list')

month_names.forEach((e, index) => {
  let month = document.createElement('div')
  month.innerHTML = `<div>${e}</div>`
  month.onclick = () => {
    month_list.classList.remove('show')
    curr_month.value = index
    generateCalendar(curr_month.value, curr_year.value)
  }
  month_list.appendChild(month)
})

document.querySelector('#prev-year').onclick = () => {
  --curr_year.value
  generateCalendar(curr_month.value, curr_year.value)
}

document.querySelector('#next-year').onclick = () => {
  ++curr_year.value
  generateCalendar(curr_month.value, curr_year.value)
}

let currDate = new Date()

let curr_month = { value: currDate.getMonth() }
let curr_year = { value: currDate.getFullYear() }

generateCalendar(curr_month.value, curr_year.value)

```

Figure 10: Continuation of JavaScript implementation.

These above figures show another reason why I decided to implement JavaScript into the final product. Using JavaScript meant that I was able to do both front-end and back-end work that would ultimately generate a calendar with multiple functionality features. This overall section of the JavaScript specifically was used in order to make sure that the calendar that will be shown when the



application is running, would be fully functional and that the user wouldn't have to be confused if they weren't able to click on any of the dates or tabs shown on the web page.

### 6.3.1 HTML and CSS Implementation

As there are multiple different HTML pages and a CSS style sheet, I thought that it would be best to Visual Studio in order to maintain good organisation and proper functioning of the overall web-page development aspect.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="Cardiff Youth Service Scheduling System" content="width=device-width, initial-scale=1.0" />
6      <title>
7          Calendar
8      </title>
9      <link rel="stylesheet" href="style.css" />
10 </head>
11 <body class="light">
12     
13     
14
15     <div class="calendar">
16         <div class="calendar-header">
17             <span class="month-picker" id="month-picker">
18                 February
19             </span>
20             <div class="year-picker">
21                 <span class="year-change" id="prev-year">
22                     <pre></pre>
23                 </span>
24                 <span id="year">2022</span>
25                 <span class="year-change" id="next-year">
26                     <pre></pre>
27                 </span>
28             </div>
29         </div>
30     </div>
```

Figure 11. HTML calendar page

```

32 <div class="calendar-body">
33 <div class="calendar-week-day">
34 <div>Sun</div>
35 <div>Mon</div>
36 <div>Tue</div>
37 <div>Wed</div>
38 <div>Thur</div>
39 <div>Fri</div>
40 <div>Sat</div>
41 </div>
42 <div class="calendar-days">
43 <div>
44 1
45 <span></span>
46 <span></span>
47 <span></span>
48 </div>
49 <div>2</div>
50 <div>3</div>
51 <div>4</div>
52 <div>5</div>
53 <div>6</div>
54 <div>7</div>
55 <div>1</div>
56 <div>2</div>
57 <div>3</div>
58 <div>4</div>
59 <div>5</div>
60 <div>6</div>
61 <div>7</div>
62 </div>
63 </div>

```

Figure 12. Continuation of HTML calendar page.

The above HTML shows how I managed to create a calendar that works with JavaScript. This code was mainly used so that a user would be able to pick the month, day, and year they wanted to add events and sessions onto with the calendar.

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta charset="utf-8" />
5 <title>Bookings</title>
6 <link rel="stylesheet" href="style.css" />
7 <script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.js">
8 </script>
9 </head>
10 <body class="light">
11
12 
13 <div ng-app="myApp" ng-controller="myCtrl">
14 <div>
15 <h2>Current Bookings for {{displayDate}}</h2>
16
17 <div class="box">
18 <table bgcolor="white" class="center">
19 <tr>
20 <th width="20">Booking Id</th>
21 <th width="20">Room Id</th>
22 <th width="100">Room Name</th>
23 <th width="100">Start</th>
24 <th width="100">End</th>
25 <th width="255">Information</th>
26 <th width="100"></th>
27 </tr>
28 <tr valign=top class="1" ng-repeat-start="x in myBookings">
29 <td width=20>{{x.id}}</td>
30 <td width=20>{{x.roomId}}</td>
31 <td width=100>{{x.roomName}}</td>
32 <td width=100>{{x.startDateTime}}</td>
33 <td width=100>{{x.endDateTime}}</td>
34 <td width=255>{{x.info}}</td>
35 <td width=100><button ng-click="selectRow($index, x.id)" type="button" class="btn-select">Select</button></td>
36 </tr>
37 <tr class="1" ng-repeat-end><td colspan="6"><hr style="height:2px;border-width:0;color:gray;background-color:gray"></td>
38 </tr>
39 </table>
40
41 </div>

```

Figure 13. View Bookings field of application

This figure shows how I was able to show the user that when they have double clicked on a date, they can see any bookings that have been made on that date, along with the room ID of where the session or event would take place along with the notes to inform the user of the event, or the recorded reflections if it was a booking for a session. This meant that the user was able to easily identify if there were any times in the day when a booking would be taking place so that they could avoid booking something at the same time. However, booking something else at the same time would pop up with an error message allowing the user to see that this action wasn't going to work.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>View Youth Booking</title>
  <link rel="stylesheet" href="style.css">
  <script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.2.1/angular.js">
  </script>
</head>
<body class="light">
  
  

  <div ng-app="myApp" ng-controller="myCtrl">
    <h2>Create Booking on {{titleStartDate}}</h2>

    <h2>Room Id: {{roomId}}</h2>
    <h2>Room :
      <select ng-model="roomId">
        <option ng-repeat="x in myRooms" value="{{x.roomId}}">{{x.roomName}}</option>
      </select>
    </h2>
    <h2>
      Start Time: {{displayStartDateTime}}
      <button ng-click="goPrevSlot()">Prev Slot</button>
      <button ng-click="goNextSlot()">Next Slot</button>
    </h2>
    <h2>
      End Time: {{displayEndDateTime}}
    </h2>
    <h2>
      Information: <textarea ng-model="info"></textarea>
    </h2>

    <!-- You need to assign event handlers in the angularjs app scope-->
    <button ng-click="goPrevSlot()">Prev Slot</button>
    <button ng-click="goNextSlot()">Next Slot</button>

    <button ng-click="saveBook(roomId, startDateTime, endDateTime, info)">Save Booking</button>
    <button ng-click="goBack()">Cancel</button>
  </div>
  
```

Figure 14. New booking section of application

```

var app = angular.module('myApp', []);
app.controller('myCtrl', function ($scope, $http) {
    var myObj, x, txt = "";

    var bookings = null;
    var dayoffset = 0;
    var currentDate = new Date();
    var logged_customer_data = null;
    var baseUrl = window.location.protocol + "://" + window.location.hostname + ":" + window.location.port;

    function getParameterByName(name, url = window.location.href) {
        name = name.replace(/\[\]/g, '\\$&');
        var regex = new RegExp('[?&]' + name + '=(?=[^&#]*)|&#|$', '');
        results = regex.exec(url);
        if (!results) return null;
        if (!results[2]) return '';
        return decodeURIComponent(results[2].replace(/\+/g, ' '));
    }

    function getFirstWord(words) {
        var wordList = words.split(" ");
        return wordList[0];
    }

    function getCustomerSession() {
        var storedData = sessionStorage.getItem("customer_session_data");
        if (storedData) {
            logged_customer_data = JSON.parse(storedData);
            currentDate = new Date(logged_customer_data.cursor_fromdate);
        }
    }

    function updateCustomerSession() {
        var storedData = sessionStorage.getItem("customer_session_data");
        if (storedData) {
            logged_customer_data = JSON.parse(storedData);
            logged_customer_data.cursor_fromdate = FormatYBDate(currentDate);
            sessionStorage.setItem("customer_session_data", JSON.stringify(logged_customer_data));
        }
    }

    function addMinutes(date, minutes) {

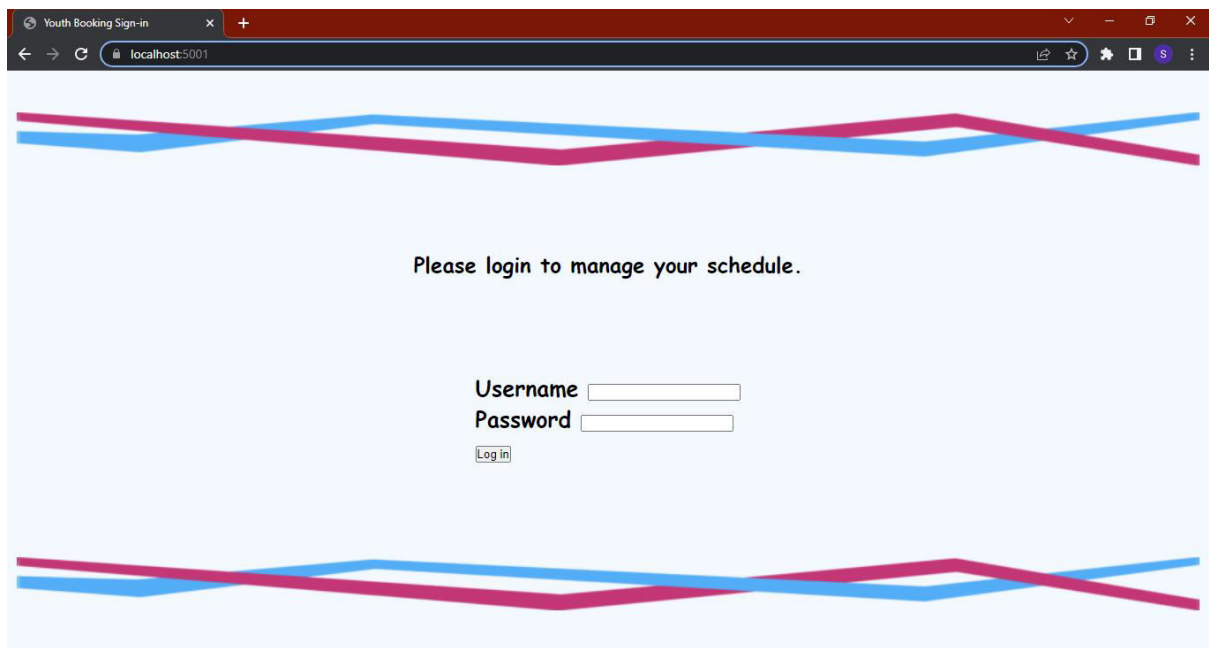
```

Figure 15. Continuation of new booking section of application.

The two figures above, show how the user was able to make a new booking after they had selected the new bookings button on the view pages section of the application. These specific sections of code allow the user to make a new booking on any date they had previously selected. This area was meant to allow the user to pick the specific slot in the day they would like to add a booking to, and it was also implemented to allow the user to make notes / record reflections on the event or session they had booked.

#### 6.4 Hosting the Application

In order to successfully launch the application once it had been completed, I thought it would be best for future work and developers if I implemented Swagger. Swagger was mostly used as a way for me to describe the overall structure of the application as I made some use of APIs throughout, in order to successfully launch the application and allow all of the features working. Swagger also acted as an extra layer of security as it means that a user would be able to launch the application from visual studio and it would come up with this (<https://localhost:5001/swagger/index.html>) in the top search bar of the web page. What a developer would then have to do to gain full access to the prototype would be to remove the /swagger/index.html line in the address and it should say (<https://localhost:5001/>) which would then direct the user to the main application. A full demonstration on how to access the application can be seen in my appendix where I have included a guide on how to properly use the Swagger interface so that any user can effectively use the application.



## 7. Results

This section of the report is dedicated to show the results I received when making the test cases that check that all of the requirements in the “Must Have” functional requirements section of 5.1.1 in the report.

### 7.1 CYS Scheduling System Prototype Test Cases

To test my CYS Scheduling System Prototype, I believed it would be best to create test cases in order to show whether I had met all of the requirements Cardiff Youth Service had asked of me. In the test cases you will see that I tested all of the “Must Have” functional and non-functional requirements, I have tested some of the “Should Have” functional and non-functional requirements, but unfortunately due to some time constraints I was unable to include the “Could Have” functional and non-functional requirements which was ultimately an okay decision as it wasn’t expected of me to include any of these features.

#### 7.1.1 Functional Requirements Check

##### 7.1.1.1 Test Case 1

Test Case Number: 1	Requirement Number: 1
Test Description	User is able to add session bookings when they have selected a date
Prerequisites	User has an account and is logged onto the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Navigate the calendar and double click a date</li> <li>2. Get onto “View Bookings” page</li> <li>3. Select new bookings</li> <li>4. Get directed to new bookings</li> <li>5. Change the room ID</li> <li>6. Change the time of day the booking will be slotted in</li> <li>7. Add any notes in the text field for the session</li> <li>8. Save session and be redirected to the previous page where they can view the new session</li> </ol>
Expected Result	The user will be able to go onto the “View Bookings” page and select the new bookings button, from there the user will be able to select a time of day to slot the booking for and what room it will be in, and

	they will be able to add notes to this session from the “New Bookings” page.
Actual Result	As expected, the user was able to access the “New Bookings” page and they were able to add a session, book the time slot, and book the room, along with notes
Comment	N/A
Pass/Fail	Pass

#### 7.1.1.2 Test Case 2

Test Case Number: 2	Requirement Number: 2
Test Description	User is able to add event bookings when they have selected a date
Prerequisites	User has an account and is logged onto the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Navigate the calendar and double click a date</li> <li>2. Get onto “View Bookings” page</li> <li>3. Select new bookings</li> <li>4. Get directed to new bookings</li> <li>5. Change the room ID</li> <li>6. Change the time of day the booking will be slotted in</li> <li>7. Add any information in the text field about the event</li> <li>8. Save event and be redirected to the previous page where they can view the new session</li> </ol>
Expected Result	The user will be able to go onto the “View Bookings” page and select the new bookings button, from there the user will be able to select a time of day to slot the event in, the user will also be able to add information on the event and then save and view the booking on the previous page
Actual Result	As expected, the user was able to access the “New Bookings” page and they were able to add a session, book the time slot, and book the room, along with notes
Comment	N/A
Pass/Fail	Pass

#### 7.1.1.3 Test Case 3

Test Case Number: 3	Requirement Number: 3
Test Description	User can add notes to any booking, whether that be a session or event
Prerequisites	The user has an account and has logged into the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Navigate the calendar and double click a date</li> <li>2. Get onto “View Bookings” page</li> <li>3. Select new bookings</li> <li>4. Get directed to new bookings</li> <li>5. Add any information in the text field about the event or session</li> <li>6. Save event and be redirected to the previous page where they can view the booking information</li> </ol>
Expected Result	I expect the application to allow the user onto the “New Bookings” page where they can then enter text into the text field provided, press save booking and for that information to then be displayed when redirected back to the previous page



Actual Result	As expected, the application allowed the user onto the “New Bookings” page where they were able to successfully able to enter anything into the text field and it then displayed after the user saved the booking and was redirected to the “View Booking” page.
Comment	N/A
Pass/Fail	Pass

#### 7.1.1.4 Test Case 4

Test Case Number: 4	Requirement Number: 4
Test Description	The user can select any date on the calendar, change month and year
Prerequisites	The user has an account and has logged into the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Logs into the application</li> <li>2. Selects the bookings tab on the “Credentials Page”</li> <li>3. The double clicks on any date to get directed to the next page</li> <li>4. Select the months tab and choose any month shown</li> <li>5. Select the year tab and scroll through the years</li> </ol>
Expected Result	I expect the system to be able to allow the user to select any date on the calendar once they get onto the main page. For the system to allow the user to select the month tab and then change month if the user selects a different month than what they are on, and for the application to allow the user to select the years tab and pick the year they would like
Actual Result	The application allowed the user to double click on the day of the week which then took them to the “View Bookings” page, the application also allowed the user to easily select the year and month tabs and change the year and month to whatever they liked
Comment	N/A
Pass/Fail	Pass

#### 7.1.1.5 Test Case 5

Test Case Number: 5	Requirement Number: 5
Test Description	The user is able to edit their weekly schedule
Prerequisites	The user already has an account and is logged into the system
Test Procedure	<ol style="list-style-type: none"> <li>1. Navigate the calendar and double click a date</li> <li>2. Get onto “View Bookings” page</li> <li>3. Select new bookings</li> <li>4. Get directed to new bookings</li> <li>5. Select edit schedule</li> <li>6. Edit any event or session</li> <li>7. Get redirected to the previous page where they can view the booking information</li> </ol>
Expected Result	I expect the application to allow the user to select the edit bookings button on the “View Bookings” page and for it to allow the user to select the booking they want to edit. Then for it to allow the edit to be saved
Actual Result	The actual edit button wasn’t able to function, however, the user could still edit the schedule by removing a booking they had made when they pressed the delete booking button, and then they were

	able to make the changes as a new booking
Comment	Whilst the edit schedule button itself didn't work; the user could still make changes to their schedule so this wasn't a complete fail hence why I will be marking it down as a pass
Pass/Fail	Pass

#### 7.1.1.6 Test Case 6

Test Case Number: 6	Requirement Number: 6
Test Description	The user can login to the application or make a new account
Prerequisites	The user is able to enter details for a new account, or the user has an account already with credentials they can use
Test Procedure	<ol style="list-style-type: none"> <li>1. Open the application</li> <li>2. Select username and password</li> <li>3. Fill in the fields</li> <li>4. Progress onto the next page</li> </ol> <p>Or</p> <ol style="list-style-type: none"> <li>1. Open the application</li> <li>2. Select create an account</li> <li>3. Fill in those fields</li> <li>4. Progress onto the next page</li> </ol>
Expected Result	I expect the application to be able to remember a user's login details or new account information and to direct them through the rest of the application
Actual Result	The was able to login to the system, however there is no create an account feature that would let a user enter new details that would then be stored onto the database
Comment	This feature is something that has been highlighted in need of before but as this was only a prototype it only needed to be able to pass test credentials to access the application, something that would need to be fixed in future work
Pass/Fail	Pass

### 7.1.2 Non-Functional Requirements Check

#### 7.1.2.1 Test Case 1

Test Case Number: 1	Requirement Number: 1
Test Description	The user can access this application on any browser on any device
Prerequisites	The user has some sort of mobile device or laptop they can use to enter the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Open browser on laptop</li> <li>2. Enter application address information</li> <li>3. Enter the system</li> </ol> <p>Or</p> <ol style="list-style-type: none"> <li>1. Open browser on mobile device</li> <li>2. Enter application address information</li> <li>3. Enter the system</li> </ol>
Expected Result	I expect the application to be accessible through the use of any device that a user chooses.
Actual Result	The application works perfectly both on a laptop and proper PC setup which was a great start, however, due to some time constraints I was



	unable to test if the application works fully on a mobile device.
Comment	N/A
Pass/Fail	Fail

#### 7.1.2.2 Test Case 2

Test Case Number: 2	Requirement Number: 2
Test Description	The user is able to easily navigate the application
Prerequisites	The user has an account and has logged into the application
Test Procedure	<ol style="list-style-type: none"> <li>1. Login to the application</li> <li>2. Select bookings on "Credential Page"</li> <li>3. Select any date on calendar</li> <li>4. Double click on a date</li> <li>5. Get directed to "View Bookings" page</li> <li>6. Select new booking button and get directed to "New Bookings" page</li> <li>7. Make a booking and save the booking</li> <li>8. Get redirected to "View Bookings" page</li> </ol>
Expected Result	I expect the application's design to allow the user to easily navigate through all of the features and functions of the prototype
Actual Result	As expected, the application was easy enough for the user to work through, they managed to select a date, add a new booking and then view that booking when redirected to the "View Bookings" page
Comment	N/A
Pass/Fail	Pass

#### 7.1.2.3 Test Case 3

Test Case Number: 3	Requirement Number: 3
Test Description	The application should be left in a state where it is functional, and a future developer can make further developments on it
Prerequisites	A future developer is able to access the application and login to it
Test Procedure	<ol style="list-style-type: none"> <li>1. Login to the application</li> <li>2. Select bookings on "Credential Page"</li> <li>3. Select any date on calendar</li> <li>4. Double click on a date</li> <li>5. Get directed to "View Bookings" page</li> <li>6. Select new booking button and get directed to "New Bookings" page</li> <li>7. Make a booking and save the booking</li> <li>8. Get redirected to "View Bookings" page</li> </ol>
Expected Result	I expect to be able to access the application through the use of a separate system so that a future
Actual Result	I was able to access the application on another laptop which meant that a future developer would be able to do the same
Comment	N/A
Pass/Fail	Pass

Out of all 9 of the test cases only one had failed which was the ability to use the system on mobile devices which meant that I had a success rate of 88.98% which was to me means that my application

can be deemed as an overall success. The only other area which could have been improved was the ability to use the edit a user's schedule when clicking on the edit schedule button, however, they were still able to delete old bookings and then add a new one with any changes they wanted to make on it, so it wasn't a complete failure regarding that test. Regarding both the actual fail and partial fail they were both due to some time constraints and with a bit more time I may have been able to implement the edit schedule button properly and also test the application to see if it would work on a mobile device.

## 8. Final User Feedback

When I had completed this project, I held a meeting with my main contact to demonstrate and explain how the application prototype is currently working and what I would like or what I could see being added in future work done to the application if they were to do any future developments to it with a more professional developer. Holding this meeting with the member of CYS allowed me to get live feedback from a senior youth officer which was largely beneficial as it means they have a better idea of what the youth service may be looking for. This meeting was also useful as it allowed me to have a better understanding of how someone who may not fully understand a lot of the background work that goes into building an application will perceive the functionality of the prototype. Having live feedback also meant that I was able to change things quickly or get notes from them as to what they would like to see changed or added to the prototype.

The overall final feedback for this project was overwhelmingly positive and the main point of contact seemed happy with what I had managed to achieve throughout the development of this project. Though I would like to see some future work done to the application, as outlined in the future works section below, I believe that the prototype is easy enough for any employee or volunteer of Cardiff Youth Service to use efficiently, agile, and effectively.

## 9. Conclusion

To conclude, this project delivered a successful bespoke application to the standards requested by CYS after having completed significant research to find the best solutions for both the front end and back-end development whilst taking into consideration the intended audience and their particular characteristics. After having conducted background research into the youth service sector both before and after the pandemic I gained a good initial understanding before the meeting with my main point of contact, which allowed me to understand the requirements and needs of CYS. Furthermore, my meetings were crucial to the development of this project particularly for the aesthetics of the application and its functionality as I was able to get meaningful feedback to update the application. For the aesthetic aspect of the application, I conducted research in the colour and design schemes of the CYS and assessed the strengths and weaknesses of the wireframe to make the best decisions possible. For the front-end aspect of this application, I researched different languages for web development and after assessing benefits and disadvantages I decided on using JavaScript, HTML and CSS. For the back-end aspect of this application, I researched and compared different languages to use for development I used JSON and C# whilst also considering different solutions for data storage and data security where I decided to use MySQL and Swagger. Continuously, I tested the application to ensure it is functioning effectively and correctly. Throughout this process I have faced challenges which did not impact the results of the projects but lead me to re-evaluating the development process which was facilitated by the fact that I was using the agile methodology and Gantt chart to ensure that the application is developed in the most efficient way possible. Lastly, I evaluated what could have been improved if there were no constraints or at least constraints and identified aspects for further development. Overall, the project and the application were a success

and met the hard requirements of functionality whilst also being user-friendly and accessible to the intended audience.

## 10. Future Work

I have managed to complete this project to a high enough standard that I am very pleased with. I have met all the requirements that were set out by CYS, and I believe that the prototype that I have finished is a success and a good representation of what a good scheduling system could look like for CYS and other youth services. However, I do believe that there could be more work done to this prototype to make it the most effective and efficient scheduling system CYS could use. This would be done by adding additional features and building upon the pre-existing features so that they could work better.

The first area I believe could have a big impact on this system in the future would be possibly to integrate to more relevant languages such as Django Python. This was an option for me at the start of this project, however, there was a much steeper learning curve for Django Python than there was for MySQL, JavaScript, JSON, and the other languages I decided to use. Deciding to learn how to use Django Python would have seriously hindered the amount of time I had to work on the prototype as I am very unfamiliar with the language and therefore, I would not have been able to get this prototype to the standard I have it in with the languages I decided to use.

Another improvement that could be added in any sort of future work would be adding more and improving the functions I added to the prototype. One of the features I would like to have improved upon is the editing the schedule area. Whilst making any sort of booking whether that be for sessions or events does work and the prototype is able to save any data the user enters, whether that be with a booking of any sort or notes, I did end up struggling to implement an editing function to the calendar which meant that users would not be able to edit an event or session that is already booked but instead they would have to delete a session or event and then have to add the session or event back into the calendar, which of course means that it is not as streamlined as it could be as this would have made the system a lot easier to use. One of the features I would have like to have added would have been some sort of tabs functionality at the top of the calendar. This would be much like a lot of other websites that have multiple tabs but as this was not a needed requirement, however, I believe adding some additional tabs and buttons to the HTML pages would have made navigation a lot easier for some users to navigate seamlessly in the application. One of the other areas I would have liked to have seem more fleshed out is my dark mode feature. This feature was implemented as a large percentage of people now prefer using dark mode as it can be more comfortable on the eyes. Whilst the dark feature on my application is functional and you can see the difference it makes to the page, it isn't available to use on all of the HTML pages I have created which means that when a user selects the feature and it activates, as soon as they go to make a booking, the page will go back to light mode which could be jarring for some people.

Overall, for any sort of future work, I would like to be able to make the application more easily accessible on other devices such as mobiles and tablets. Though the application can function on a mobile or tablet through web browsers, it isn't as refined as it could be which could make it harder to use on such devices. Allowing the user to use the application on a mobile or tablet would mean that they would be able to work more seamlessly and on the move which can be very beneficial to the user's time and schedule they may have in the day.

## 11. Project Evaluation

This project has been one of the most difficult projects I have worked on to date, however, this has still been one of the most enjoyable experiences I have had throughout my university years. This project has given me a much better understanding on what goes on with making schedules and routines within any sort of company and it has also given me a great experience in the development cycle that any developer would have to go through for most types of clients. The process of developing everything throughout this project has taught me the importance of making and keeping up with a strict work plan and schedule and it also has taught me the importance of doing my research before doing any sort of project.

Throughout the development of this project, I have come across many challenges that I wasn't fully aware that I would have to face. An example of this being, having to refresh my knowledge on a lot of different languages that I have implemented into the prototype, especially MySQL. This wasn't something that I was fully prepared for, and it certainly came as a shock when I did have to refresh my knowledge, but that was a welcomed challenge, and I may to overcome it in a fairly short period of time. I did have some prior knowledge of MySQL, JavaScript, HTML, CSS, and C# which did come in handy for a lot of the work I have implemented into the prototype. However, this was the first project I had ever done where I had to implement so many different languages in order to effectively design and build the prototype for CYS and the scale of this project wasn't quite what I was expecting. In hindsight, I have gained a better appreciation of the work and time that actually goes into building a lot of the applications and systems within the industry that get used either extremely regularly or every day, and this is because I realised how much stress can accumulate while trying to make a product for a client as best you can as you want to satisfy all their requirements for said product.

## Appendix

Appendix\_1 includes Setup of Development Environment on Windows 10

Appendix\_2 includes how to use the Swagger interface

## References

1. Bordoli, J., 2022. *What are the top Microsoft Power Apps limitations?* [online] TechTarget. Available at: <<https://www.techtarget.com/searchcontentmanagement/tip/What-are-the-top-Microsoft-Power-Apps-limitations>> [Accessed 10 May 2022].
2. Cardiffyouthservices.wales. 2021. CYSs. [online] Available at: <<https://www.cardiffyouthservices.wales/>> [Accessed 30 January 2022].
3. Cherry, K., 2020. *How the Color Blue Impacts Moods, Feelings, and Behaviors.* [online] Verywell Mind. Available at: <<https://www.verywellmind.com/the-color-psychology-of-blue-2795815>> [Accessed 6 May 2022].
4. Nix Colour Sensor Ltd. 2022. *Color Column: Magenta.* [online] Available at: <<https://www.nixsensor.com/blog/color-column-magenta/#:~:text=Magenta%20is%20the%20color%20of,%2C%20happiness%2C%20contentment%20and%20appreciation.>> [Accessed 6 May 2022].
5. Crena-Jennings, W., Perera, N. and Sibieta, L., 2021. *Education recovery and resilience in England.* [online] Education Policy Institute, pp.7-13. Available at: <[https://epi.org.uk/wp-content/uploads/2021/05/Education-Recovery\\_EPI.pdf](https://epi.org.uk/wp-content/uploads/2021/05/Education-Recovery_EPI.pdf)> [Accessed 5 February 2022].

6. Educational Policy Institute, 2020. *Preventing the disadvantage gap from increasing during and after the Covid-19 pandemic*. Proposals from the Educational Policy Institute. [online] Educational Policy Institute, p.5. Available at: <<https://epi.org.uk/publications-and-research/disadvantage-gap-covid-19/>> [Accessed 13 March 2022].
7. McKee, V., Oldfield, C. and Poultney, J., 2010. *The Benefits of Youth Work*. [online] Unite the Union, pp.13-28. Available at: <[https://www.cywu.org.uk/assets/content\\_pages/187799973\\_Benefits\\_Of\\_Youth\\_Work.pdf](https://www.cywu.org.uk/assets/content_pages/187799973_Benefits_Of_Youth_Work.pdf)> [Accessed 12 March 2022].
8. McOmie, A. and Ionescu, S., 2022. *Google Calendar review*. [online] TechRadar. Available at: <<https://www.techradar.com/uk/reviews/google-calendar>> [Accessed 10 May 2022].
9. Nielsen, J., 2020. *10 Usability Heuristics for User Interface Design*. [online] Nielsen Norman Group. Available at: <<https://www.nngroup.com/articles/ten-usability-heuristics/>> [Accessed 15 April 2022].
10. UK Youth, 2021. *The impact of COVID-19 on young people & the young sector*. [online] UK Youth, pp.3-8. Available at: <<https://www.ukyouth.org/wp-content/uploads/2021/01/UK-Youth-Covid-19-Impact-Report-.pdf>> [Accessed 13 March 2022].
11. Vizard, T., Sadler, K., Ford, T., Newlove-Delgado, T., McManus, S., Marcheselli, F., Davies, J., Williams, T. and Leach, C., 2020. *Mental Health of Children and Young People in England, 2020 (Wave 1 follow up to the 2017 survey)*. [online] NHS Digital, pp.9-11. Available at: <<https://digital.nhs.uk/data-and-information/publications/statistical/mental-health-of-children-and-young-people-in-england/2020-wave-1-follow-up>> [Accessed 8 February 2022].