SDD Logbook
Jesse Whitford

Date	**23rd -24th of March**
Time spent on the project	**4 Hours**

Work completed

Over the weekend I spent time brainstorming ideas for my project and starting on the “Defining the problem” section of the design brief. One of the ideas that I had for my project was a financial planning app where users could enter their income, expenditure, investments, and savings, to calculate a budget for them and help them invest the leftover money. Another idea that I had was a platform for students to help manage their marks and assessments, with other features such as a time table and widgets. I also thought about doing an iPhone app lockout to help students focus when studying, and a website that would help people manage and review the books they want to read, songs they want to listen to, and the TV shows/movies they want to watch.

Eventually I fell onto the student mark management program, from this I laid out and

consolidated my ideas into the Defining the problem section of the design brief.

Defining the problem

- identifying the problem - Briefly describe the program you have decided to develop. What problem does it solve and why have you chosen this.

  - The program that I have chosen to develop is a program that helps students manage their grades, upcoming assessments, and timetable. I am looking to make a web app.

    - needs of the client
      - functionality requirements - picture yourself as the client and describe what you want from the finished program. Functionality requirements encompass the various tasks, operations, and processes that the software should be able to perform to meet the user's needs. This could include user interface features, data processing capabilities, specific outputs, integrations with other systems, and user interaction mechanisms. Identify SMART goals from which you can measure success when it is complete (A well-worded goal will be Specific, Measurable, Attainable/Achievable, Realistic and Time-bound (SMART).)

  - I want to have a system where users can input all their upcoming assessments and track what they need to study for, a system where users can track all of their marks and see where they need to improve, and a timetable system where users can enter in their classes and see on the dashboard what classes they have on that day.

      - compatibility issues - this refers to the ability of the software to operate effectively with other systems, software, and hardware environments. Identify the operating system to be used (like Windows, macOS, Linux), software tools, browsers required etc.

  - Since I am planning to implement the program as a web app, there will be little compatibility issues. The program will be able to run on all desktop operating systems that support modern web browsers

      - performance issues - this refers to how well the software operates under certain conditions. This includes speed, responsiveness, scalability, and stability of the application.

  - Since web apps are lightweight, I expect the program to be snappy and responsive on most platforms. It will have a high scalability due to the use of local storage instead of server side storage

    - boundaries of the problem - What is the scope of the project you are attempting? Scope is

Date	**30th of March**
Time spent on the project	**30 Minutes**

Work completed

Today I researched competitors to my software project, to which I found that Microsoft Excel and Google Sheets were the only programs similar to what I was building. If this project

were to be for financial gain, this would have been an excellent identification of a gap in the market to fill, especially because it is such a specific niche.

Issues relevant to a proposed solution

- determining if an existing solution can be used
    - social and ethical considerations - what social and ethical issues should be considered if you are using/modifying an existing solution?

- Compliance of data protection and privacy laws
- Intellectual property rights and licencing
- Security of user information and sensitive data
- Bias of creating a fair experience for all users

    - consideration of existing software products - are there programs that already do the tasks your proposed project will perform? If so, why are you developing this product? What is the market niche?

- Currently, there are no existing “ready-made” solutions. The only somewhat similar programs are Microsoft Excel and Google Sheets, both of which would need a high level of understanding to build something similar.

    - customisation of existing software solutions - If an existing product meets most but not all requirements, customization might be a viable option. Could you modify an existing software product to meet your specific requirements.

- Microsoft Excel is highly customisable and can be changed to the user's needs provided the user has adequate knowledge of Excel and its functions.
  Google Sheets is not as customisable as Excel but still highly customisable depending on the users knowledge level.

    - cost effectiveness - This concerns the financial implications of using an existing solution versus developing a new one. Is the task financially feasible?

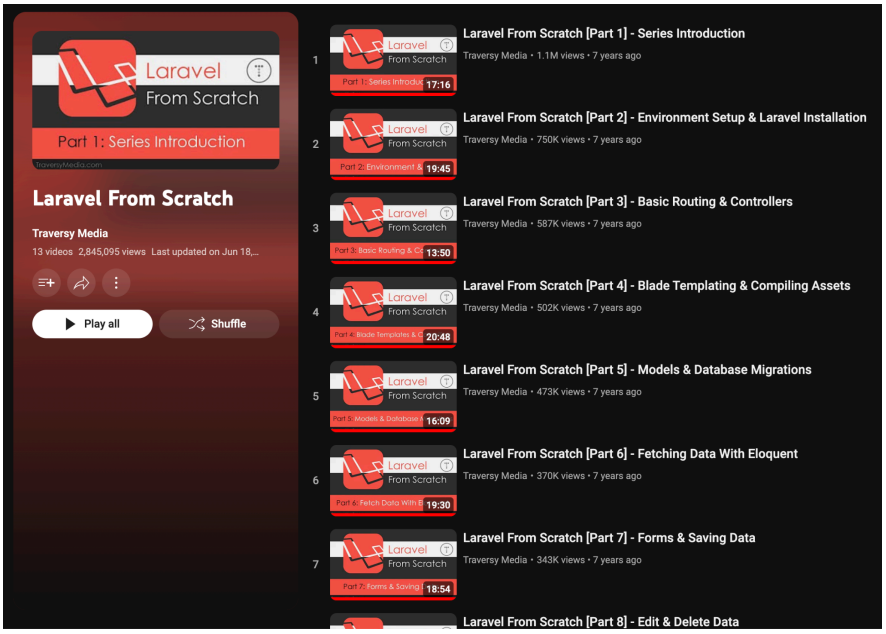
- It would be highly cost effective as users could save time and money trying to learn how to use Excel and

I also completed the Gantt Chart to help keep me on track.

Date	**15th - 17th of March (Holidays Week 1)**
Time spent on the project	**9 Hours**
Work completed	

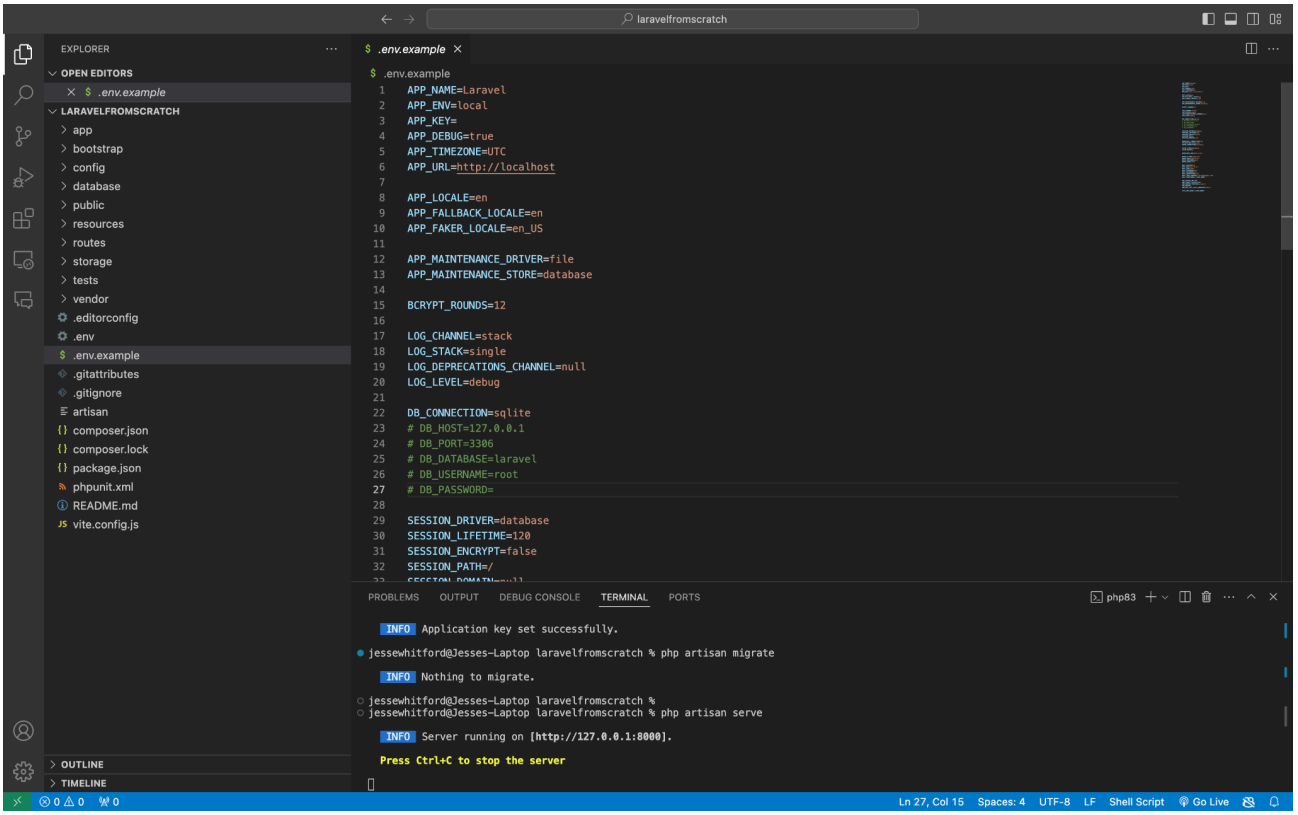
Over the past three days, I decided that my project would be a web app. I had a look at a few options for programming languages including HTML, PHP, Laravel, C++, and TypeScript. I chose to learn Laravel and build my website with it due to its highly responsive and dynamic nature, allowing the quick and easy development of websites that have a modern look and feel all while keeping the back end components tidy and decluttered.

I chose to use the Laravel from Scratch youtube series (https://www.youtube.com/playlist?list=PLillGF-RfqbYhQsN5WMXy6VsDMKGadrJ-) on the recommendation of my dad as it would help me to learn the basics of the Laravel language and framework, as well as get me going on my own project.



Throughout this experience of learning Laravel, I encountered many hurdles. For example, even though this is a good tutorial, Laravel has updated quite a bit since this has been made, so many of the commands and file locations used have changed or don't exist any more. When this happened, which was quite frequently, I would have to go looking through forums and documentation to find how to follow along with the tutorial. If I couldn't find what I was looking for, I would ask my dad as he has completed this series before and managed

to get around a lot of the issues that I had encountered.

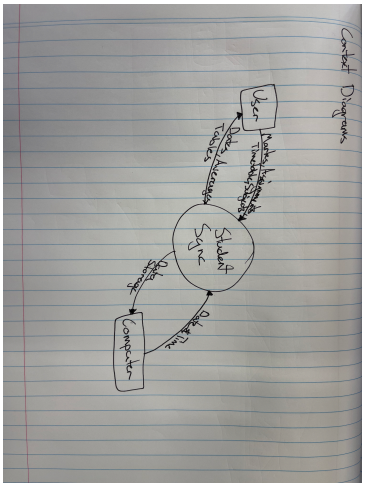


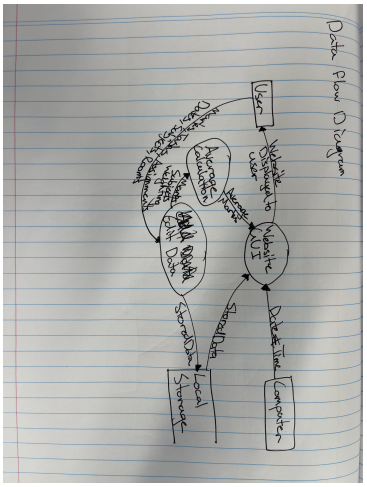
I eventually finished the series with a sizable project folder and the knowledge of how to build what I want. I won't be able to any more this week as I have driving lessons in preparation for my test this week.

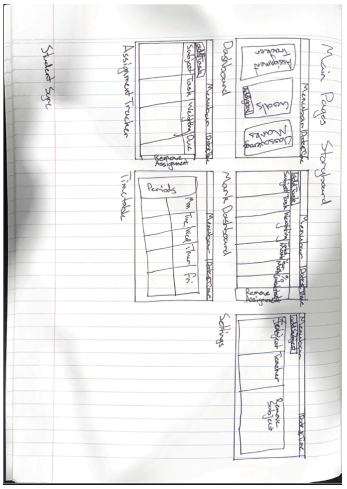
Date	**23rd - 24th of April**
Time spent on the project	**5 Hours**

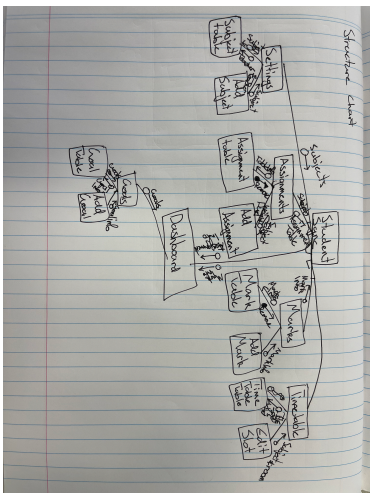
Work completed

Over the past two days, I worked on refining my idea and finishing off the design brief document. This included creating various charts and diagrams, pseudocode and data tables, as well as the information boxes that needed to be filled out.





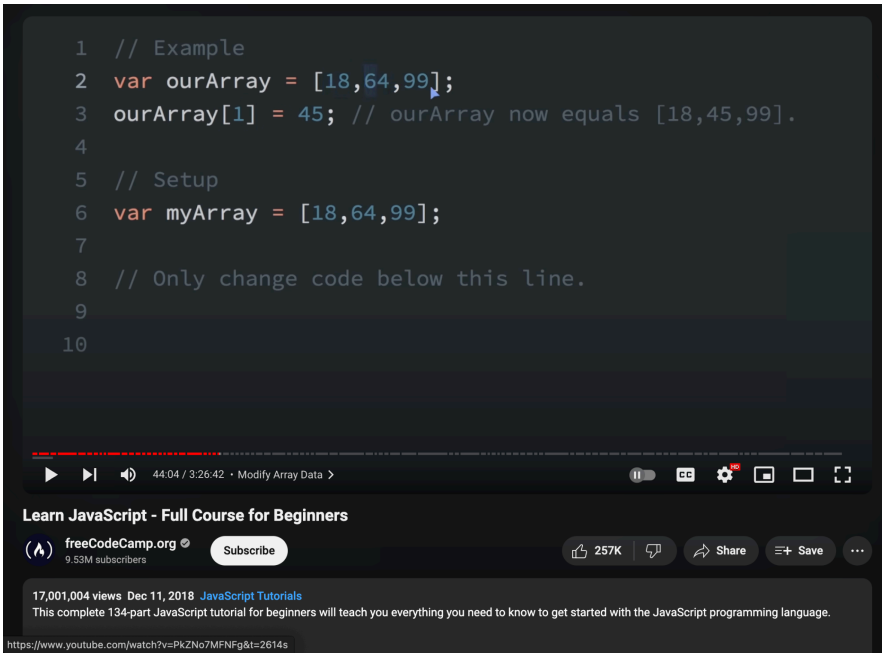




Date	**25th of April**
Time spent on the project	**1.5 Hours**
Work completed After trying for a few hours to try and get the website started in Laravel, I have decided to switch to HTML, CSS, and JavaScript for my website. After being unable to achieve what I was trying to, I learned that Laravel is way too complex of a language for what I am trying to make. It is jam packed full of features such as user handling and email management that get in the way of what I am trying to build.	

Date	**26th - 28th of April**
Time spent on the project	**6.5 Hours**
Work completed	

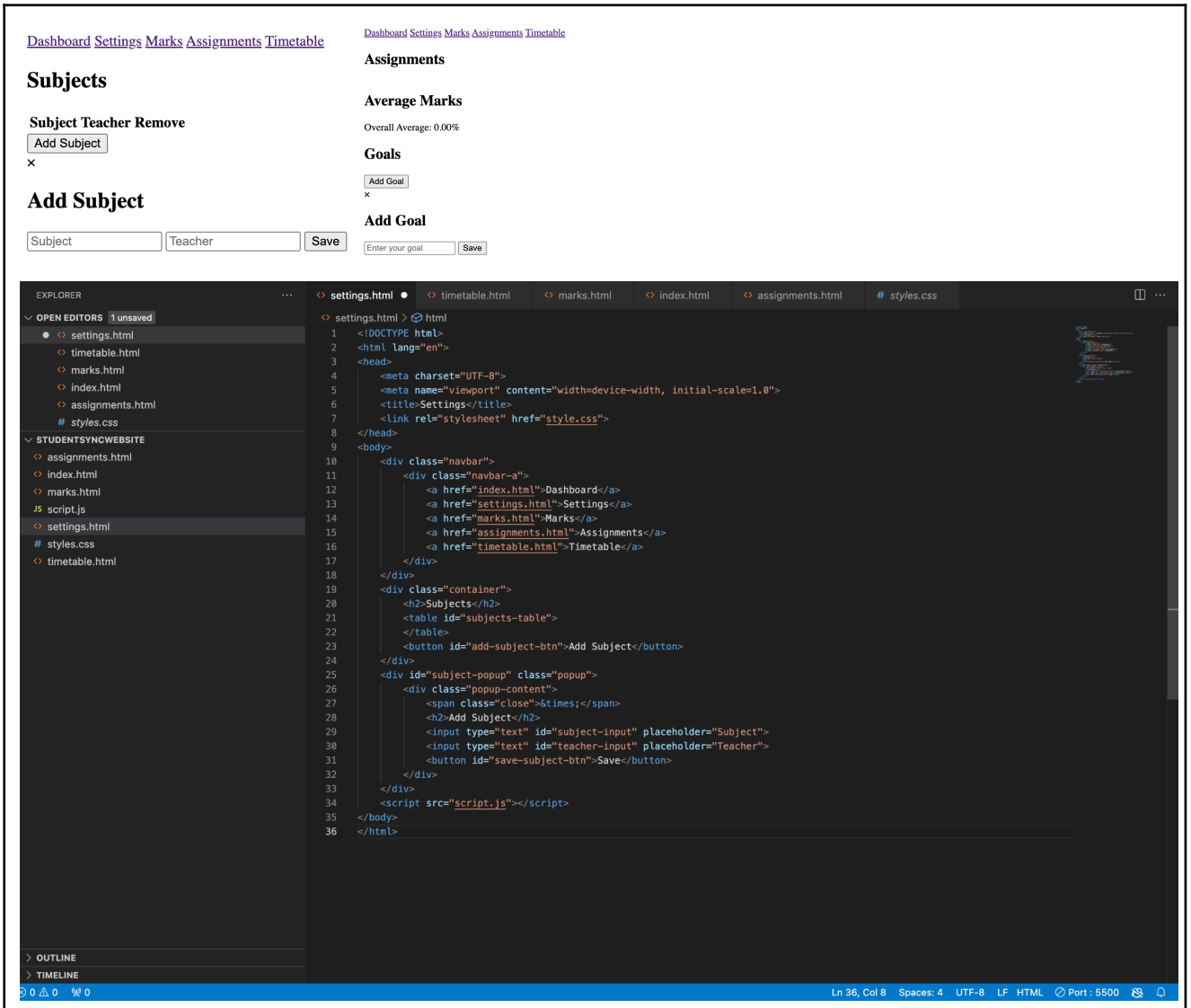
Over the past few days, I have researched what I will need to program for my website and completed the Learn JavaScript - Full Course for Beginners on the freeCodeCamp.org youtube channel (https://www.youtube.com/watch?v=PkZNo7MFNFg&ab_channel=freeCodeCamp.org). I chose to do this course as I don't have the largest knowledge of JavaScript and this will help me with producing things such as global arrays and accessing the browsers local storage in a way that it can be used by all pages on the site.



Date	**3rd - 5th of May**
Time spent on the project	**5 hours**

Work completed

Over the past three days, I have worked on creating the base HTML and JavaScript of the website. So far I have done the Settings and Dashboard pages. The main challenge that I have been having with the Settings page is trying to me the Subject field store itself in a location where it is accessible to the rest of the web pages. I managed to solve this issue by forcing the JavaScript to make the JSON file where it was being stored to become global to the rest of the pages. I also managed to create a button that hides/shows a popup allowing the user to enter some information.



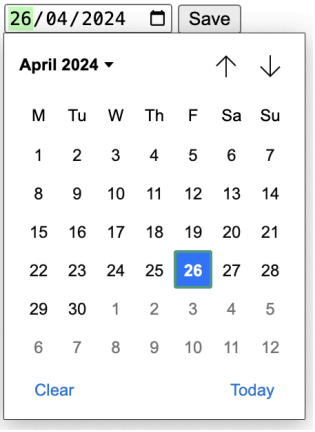
Date	**6th - 8th of May**
Time spent on the project	**4.5 hours**

Work completed

I have been highly productive over the past few days, managing to complete the Marks and Assignment pages using code from the Settings and Dashboard pages to create the data entry fields and have some of the data globally accessible to use in the dashboard.

On the Assignments page, I wanted to up my game a little bit by adding a calendar button on the due date for ease of use. After searching around for a while and not finding any

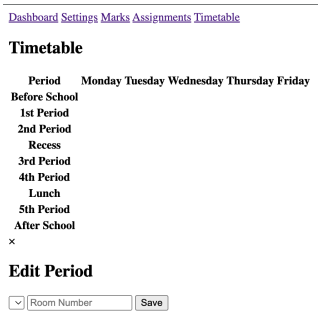
solution, I employed ChatGPT to help me implement this feature. It helped me implement this very modern and clear calendar to the assignments page with a clear entry and jump to today feature.



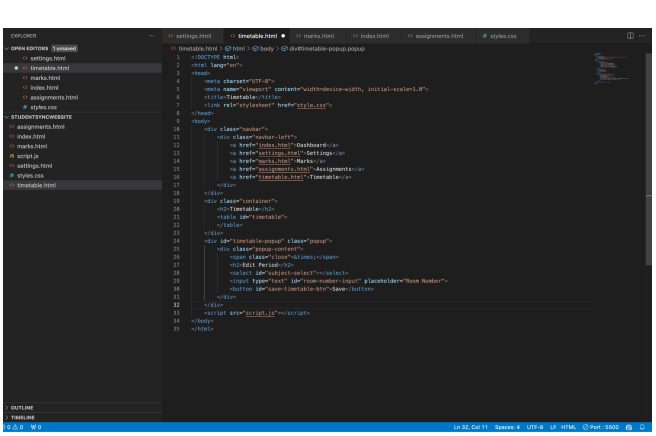
Date	**12th of May**
Time spent on the project	**3 Hours**

Work completed

Today I worked on creating the Timetable page. Although creating the table and popup were easy, it was a challenge to manage to combine the both. The way that I managed to get them working together was to make each cell of the table into a hidden button which would activate the edit timetable slot popup.

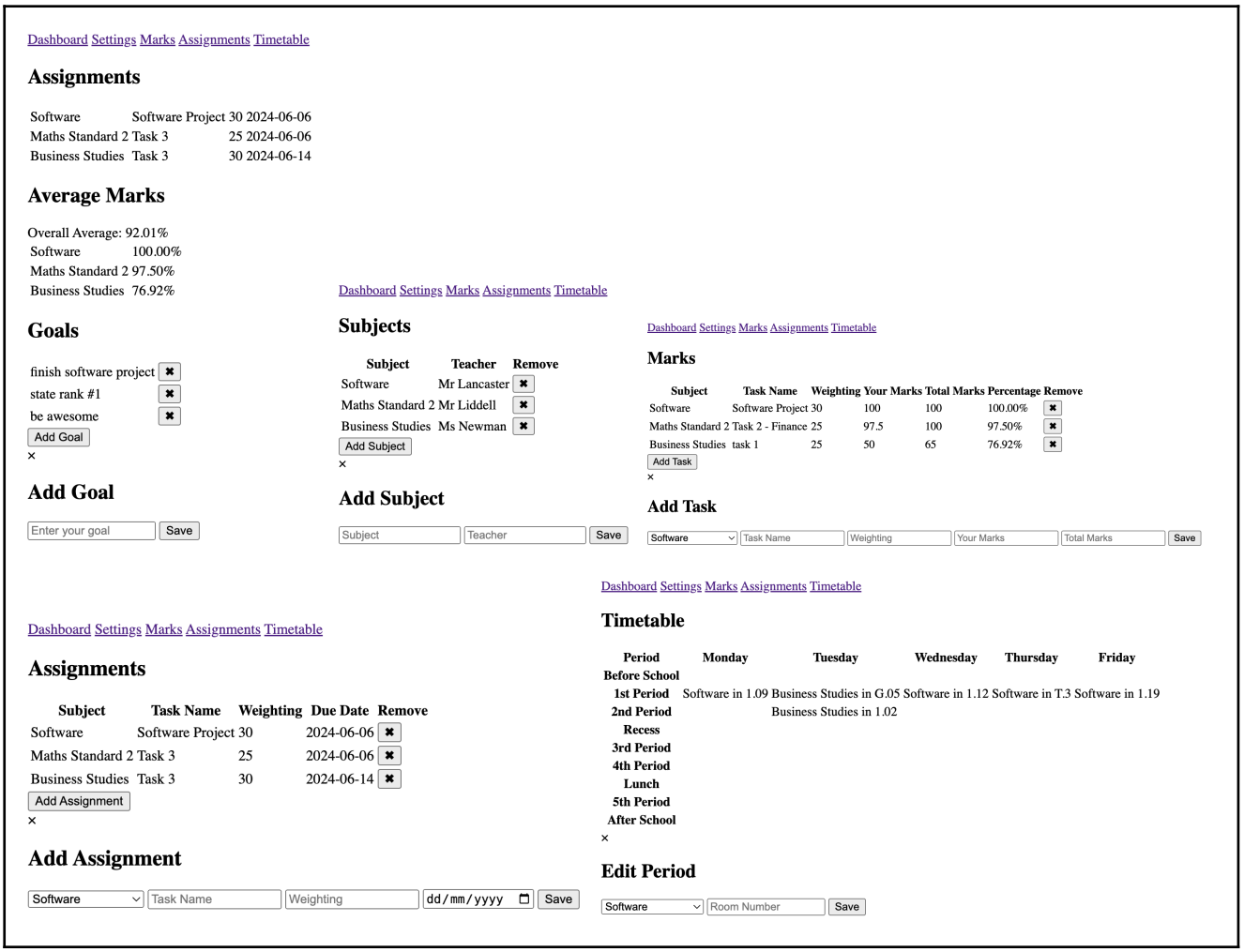


Although currently there are no dividers between the table slots, I will fix this later with CSS.



Date	**17th of May**
Time spent on the project	**2 Hours**
Work completed During this double lesson today, I decided to start laying out my user documentation and everything that needed to be put in it. This involved creating all the headings, as well as creating dot points of what needs to go under each heading. I also went back through my Design Brief and checked that it was all in order.	

Date	**18th - 19th of May**
Time spent on the project	**2.5 Hours**
Work completed Over the weekend, I have done some testing of the math in my programs for the Dashboard and Marks pages. I have also checked that all of my global data is being parsed through to the other pages and it was highly successful. All of the maths calculating the percentages and averages for tasks and subjects, as well as the global information is working correctly as shown through the below screenshots.	



Date	**21st of May**
Time spent on the project	**1 Hour**

Work completed

Today as I was trying to quickly implement some CSS before I go on camp, I started encountering the error seen below. Attempts to find a solution online have not been helpful so I will have to have a go at it once I get back from camp.

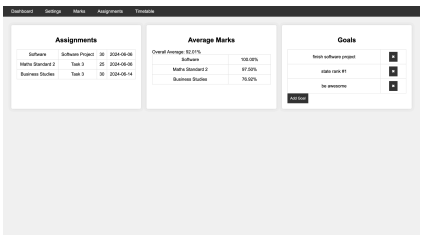
```
Refused to apply style from 'http://127.0.0.1:5500/st  timetable.html:10
yle.css' because its MIME type ('text/html') is not a supported
stylesheet MIME type, and strict MIME checking is enabled.

Refused to apply style from 'http://127.0.0.1:5500/sty  timetable.html:1
le.css' because its MIME type ('text/html') is not a supported
stylesheet MIME type, and strict MIME checking is enabled.
```

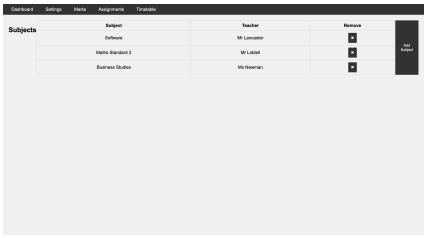
Date	**25th - 26th of May**

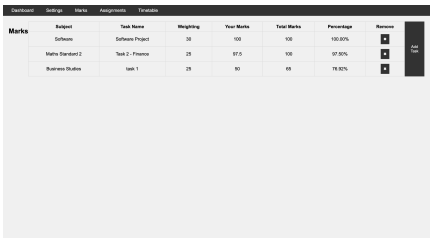
Time spent on the project	**2.5 Hours**

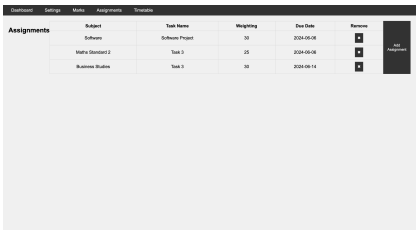
Work completed

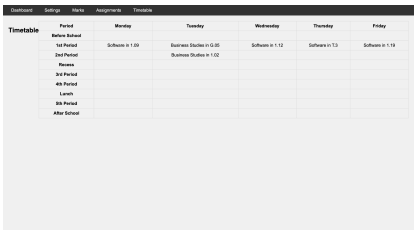
Today I was working on trying to figure out what that error meant and implement some basic CSS into the website. I finally found out that it was because I named the file styles.css, while all the pages were referencing style.css. After a quick rename of the file, I threw together this basic CSS and this is how the website looks now.



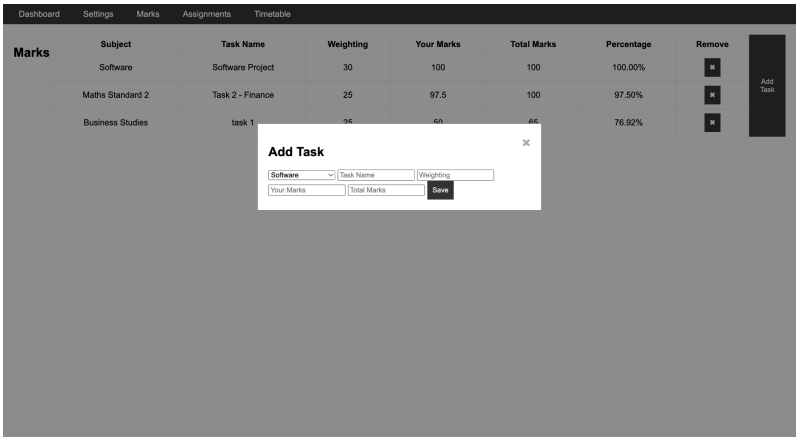








I am going to need to adjust some of the tags in the html code to fix the headings, as well as give the website a better look and feel, as well as incorporate screen design principles.
I also got all of the popup windows looking like actual popup windows, which is good as it helps separate it from the rest of the page.

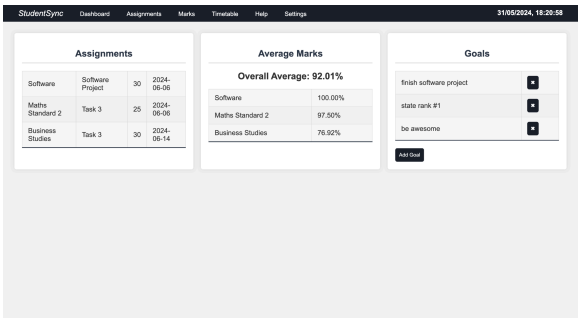


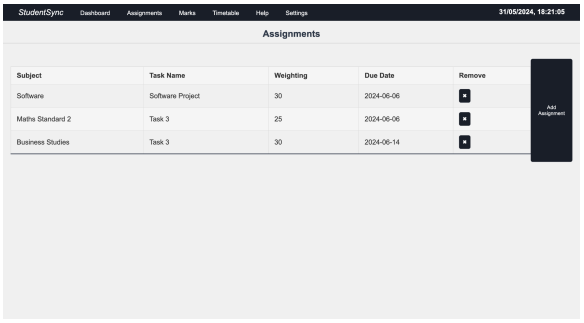
Date	**28th to the 31st of May**
Time spent on the project	**6.5 Hours**

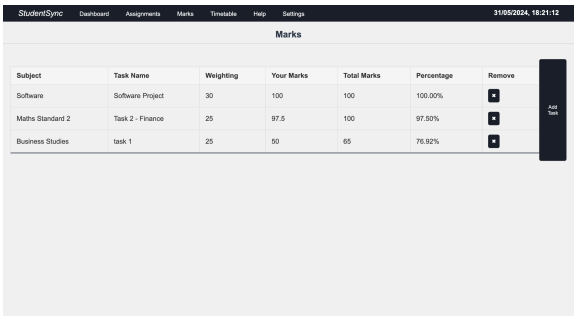
Work completed

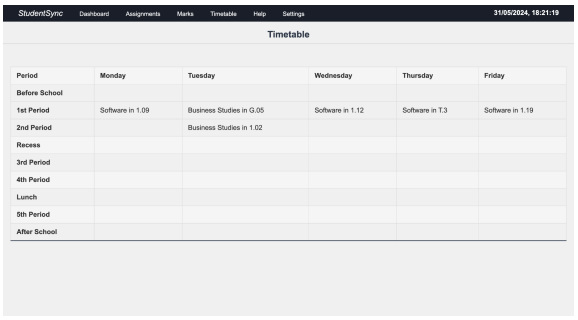
These past few days have been a bit sporadic, I have done a few hours at home and 3 hours at school. Over the past few days at home, I have managed to freshen up the website with a brand new look and feel, fixing the issues with the headings and making the website

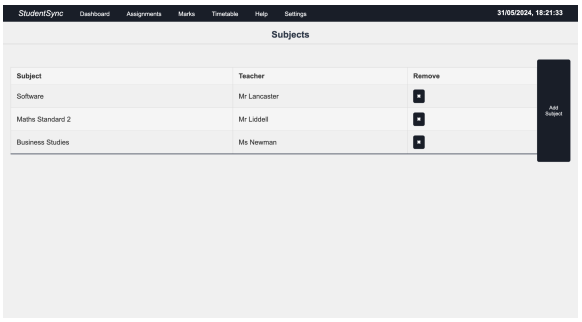
operate better as a whole while incorporating many ergonomic and screen design principles. I also incorporated a help page to put the user manual on, as well as the clock feature that I wanted to add, helping users keep track of the date and time. It uses the computer as an API and grabs the time and date every few milliseconds to stay up to date. Below are images of the new pages and menubar, complete with a logo and the time feature, as well as a refreshed popup window.

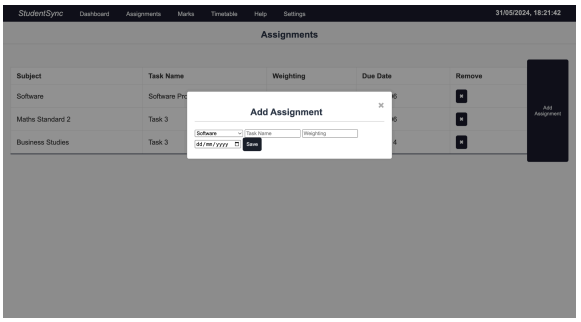












During my three hours at school, I created all of the user documentation including guides, tutorials, installation, troubleshooting, and a reference card.

Installation Guide

Since StudentSync is a web based application, minimal disk space is required to use and operate the program. It runs over the internet through the use of a web browser, with a small amount of local storage required for the storage of information (<1MB). Listed below are the minimum browser versions and hardware requirements to run StudentSync.

Browser	Minimum Version*	How To Check Version
Google Chrome	115	Settings > About Chrome
Microsoft Edge	115	Settings > About Microsoft Edge
Safari	16.5	Safari > About Safari
Mozilla Firefox	116	Help > About Firefox
Tor	12.5	Help > About Tor Browser
Opera	102	Help > About Opera
Chromium	117	Help > About Chromium

** Although this is the minimum version, StudentSync may work on previous versions. This is not recommended due to various bugs and security issues found in the programming of these browsers.*

While hardware requirements for browsers may vary, these are the hardware specifications for a device to be able to run StudentSync.

Component	Minimum Specification	Recommended Specification
CPU	Intel Core 2 Duo (or AMD equivalent)	Intel i5 7600 (or AMD equivalent)
RAM	4GB	8GB
Storage	1MB Hard Drive	500GB SSD

Reference Card

Feature	What does it do?
04/06/2024, 21:49:14	This is the clock, it appears on the menu bar and helps you keep track of the time and date down to the second.
StudentSync Dashboard Assignments Marks Timetable Help Settings	This is the menu bar. It contains the name of the program (StudentSync), as well as access to all the other pages within the website.
✖	This is the remove/cross button, it allows you to remove items from lists throughout the website.
Save	This is the save button, it allows you to save and close the popup that you clicked onto.
✖	This is the exit/cross button, it allows you to exit from a popup without saving your changes.
Subject	This is a text entry field, it allows you to enter in text relating to the information highlighted by the gray text in the background (which in this case, is the subject).
Software	This is a dropdown menu, clicking on it allows you to scroll through and pick one of your subjects.

Troubleshooting Guide

My subjects aren't appearing in the dropdown menu

- This usually happens when the computer is running out of memory or processing power. This can be resolved by doing a hard refresh on the browser. On Mac this is done via Command + Shift + R, and on Windows this is done through Control + Shift + R. If this hasn't resolved the issue, restart the computer.

An item isn't disappearing when I press the cross button

- This is usually caused when the web browser freezes up due to a lack of memory. Try to close and reopen the browser. If this hasn't resolved the issue, restart the computer.

My average mark is incorrect

- This usually happens due to the local storage cache filling up and StudentSync being unable to read the information properly. Please check the browser's user guide on how to clear your cache and cookies (this is usually done via the settings menu).

I can't make a timetable slot empty again

- This is due to a bug within our programing. Please bear with us while we try to fix this issue. Keep in mind that StudentSync is in an early access stage so not all features are working and available.

Date	**1st - 2nd of June**
Time spent on the project	**1.5 Hours**

Work completed

Over the weekend, I worked on completing a testing report for my project. This included testing the data of the fields in the program to check that they are all in working order. The testing report is available in a separate document.

Date	**4th of June**
Time spent on the project	**1 Hour**
Work completed Today I spent an hour tidying up all of my folders and the resources for the project.	

Reflection
Reflecting on my project, there are many areas which could be improved and further developed. I would have liked to expand upon the Timetable page, which is a really cool feature but could use some upgrades. I would have liked it to eventually become a calendar like function so you could see what week it was and what subjects you will have today. This feature could have linked up with the Dashboard to show the user what classes that they will have on that day. I would have also liked to create an Apple widget for this feature so that a quick look at your phone or desktop will show you what classes that you have on that day. If I were to go further with this program, I would add in a fully fledged login system and the ability to cloud sync your data. This would mean that you can log in and see it anywhere rather than just on the one browser. I also would have liked to add in some form of encryption for the users data. Currently it is protected by the browser's internal files but it is easy to access through something as simple as the built in developer tools.