Documentation

# Introduction

This report presents the project plan for the COMP1004 module coursework. The project has a focus on self-management and the use of time. Within this document, the software development lifecycle is discussed with an explanation of how it was applied to this project. Following the plan for the project is a design document.

# Software Development Lifecycle

This section discusses the Software Development Lifecycle (SDLC) and describes how it has been used in this project.

The planning phase for this project took place over a week. Multiple ideas were written down and researched to see how useful they would be, and whether if they’d been solved before how successful that implementation was. Notes were taken for the requirements and analysis, and design phase here.

The next phase to be carried out was gathering requirements for the project and writing these down in the form of user stories. This shows what the user should be able to do with the project once completed. Continuing from the requirements phase, a plan of sprints was written down with rough times required to complete them.

Following this stage is the design phase. In this phase, the project vision was created, this outlines the purpose of the project; some background of the project was written down; user stories were written down here; wireframes were created, these provide an idea of what the site should look like when complete; UML diagrams were written down, these provide a template for classes and how data is stored in the program.

The next phase is the implementation phase. This phase took place over many weeks, following the sprint plan so that the project is completed accurately and on time. The site was created, and then the JavaScript file was created that allows the user to interact with the site, as well as a NodeJS server so that the website can communicate with the server to write data back to the server.

# Design

## Project Vision

The idea of this project is to allow users to manage the tasks that they need to complete and assign them to days that they can do them. This will allow users to manage their time efficiently so that they complete all they need to in the time that they have.

## Background

People have been using calendars for years to manage when certain tasks need to be done, and a digital calendar will allow users to manage their time from any device anywhere. This will allow users to be organised and get tasks completed before the deadline.

Consideration must be taken regarding the accessibility of this project when the interface is designed. If the site is not accessible then the target audience is limited, and would go against the UK Equality Act of 2010. The Equality Act states that reasonable steps must be taken to make a website more inclusive.

## User Stories

* The user should be able to log into the system.
* The user should be able to register for an account.
* The user needs to be able to add projects that need to be done.
* The user needs to be able to add a deadline for tasks.
* The user should be able to add a predicted time required to complete the task.
* The user should be able to undo previous actions.
* The user should be able to redo previous actions.
* The user needs to be able to mark tasks as complete.
* The user should to be alerted when a task is due to be started.
* The user should be alerted when a deadline is reached for a task that is not completed.

## Wireframes

This section provides wireframes for the project. They demonstrate an idea for what the project could look like.

A screenshot of a calendar

Description automatically generated A screenshot of a login screen

Description automatically generatedA screenshot of a registration form

Description automatically generated

## UML diagrams

This section shows UML diagrams for the project, and how data can be stored.

A screenshot of a computer program

Description automatically generated A close-up of a computer code

Description automatically generated A screenshot of a computer

Description automatically generated

# Sprints

## Sprint 1

* Create introduction for documentation.
* Create product vision.
* Research background.
* Create wireframes.

## Sprint 2

* Create UML diagrams.
* Program the template for the application
  + Nav bar
  + Calendar
  + Log in modal
  + Register modal

## Sprint 3

* Create functions for navigating backwards and forwards through the calendar (previous week, next week, and this week)
* Highlight today in the calendar

## Sprint 4

* Create NodeJS function so that the server can check if the user exists
* Create JS function so that the user can log in
* Create NodeJS function so that the user can register an account

## Sprint 4

* Create JS function so that the user can register an account
* Create NodeJS function for creating a new reminder

## Sprint 5

* Create JS function so that the user can create a new reminder
* Create HTML pop up with a form so that the user can create a reminder

## Sprint 6

* Create NodeJS function to send users reminders to client
* Create reminder template and position within table

## Sprint 7

* Create JS function to receive reminders and sort them into a list
* Sort reminders so that they appear in the correct place in the table

# Use Case Descriptions

|  |  |
| --- | --- |
| Name | Log In |
| Short Description | Allow user to log in |
| Precondition | User exists and is logged out |
| Post Condition | User is logged in and can see their reminders |
| Error Situations | User does not exist |
| Actors | User |
| Triggers | User needs to log in |
| Standard process | 1. User enters log in details 2. System checks to see if user exists 3. If user exists check passwords match 4. System confirms the user and logs in |
| Alternative Process | 1. System responds that log in credentials are incorrect |

Register

Create reminder

Amend reminder

# Notes

* Data needs to be saved to a JSON file so that the user can come back to it later.
* Notifications could be visual on the screen or use external devices to indicate (e.g. Philips Hue lights change colour depending on the status of different tasks).