
CMP-5012B Software Engineering

Requirements and Design – Report

Study Planner Application

Team ID: 5

Team Member Name	Email address	Student ID
Felix Meulenkamp	pdw19yuu@uea.ac.uk	100250515
Dom Sierotowicz	cqt19quu@uea.ac.uk	100285541
Misza Pawlicki	cks19ctu@uea.ac.uk	100289045
Daniel Holmes	kmr20bqu@uea.ac.uk	100305675
James McGowan	tps19ucu@uea.ac.uk	100277997

Date: 11/03/22

1. Introduction and Project Ideas [10% marks]

1.1 Main objective and overview

The aim of our Study Planner app is to help students keep track of their study profiles, allowing for detailed planning as well as prioritisation and tracking of tasks. A student can do this by creating a study profile for the upcoming term by selecting the appropriate semester identifier and then using a personalised file provided by the hub. The file will contain information about an individual's coursework, exams, weighting of assessments and their upcoming deadlines. Specific information can be accessed by selecting a module page inside the semester profile. This page will display module specific activities, milestones, and deadlines. A module organiser will be able to change deadlines when required.

Users will be able to define study tasks and milestones which contribute towards larger assessments. These will be specific to events contained inside the modules. Study tasks may have dependencies or be done in parallel. A user can track their study progress by defining which activities they have completed alongside a quantity of hours spent on said activity. These activities will contribute towards a user's tasks and milestones. All of these will be viewable holistically on a dashboard which will allow the user to judge their progress during the semester. Visual aids, such as progress bars, will help the user better understand how they are progressing on each deadline.

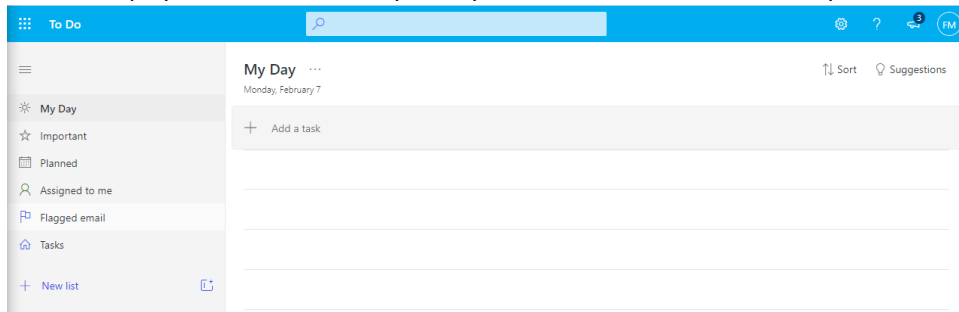
1.2. Analysis of similar systems (*minimum of 5*)

We analysed the following similar systems, identifying their major features, operating environments as well as the advantages and disadvantages of each system. Our systems are all closely related to our project scope of a study planning application.

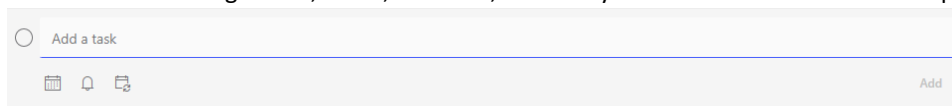
System Name: Microsoft to do

General Overview and Highlights:

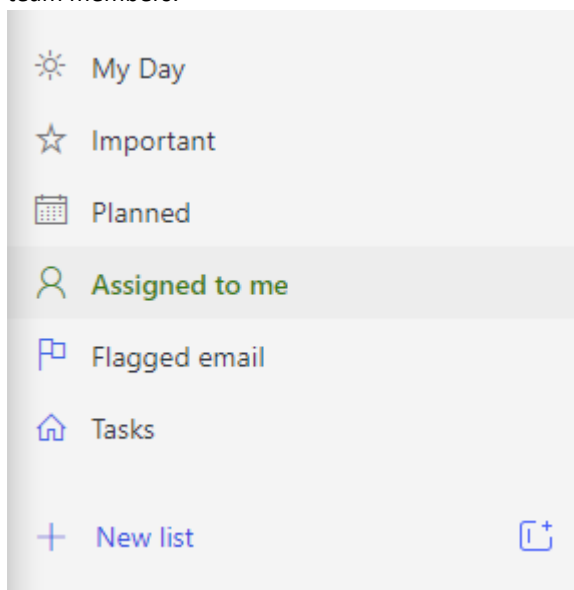
Microsoft to do is a timetabling system where you can add tasks for yourself or assign tasks to others. Tasks are then displayed on a day-to-day basis on when they are to be completed.



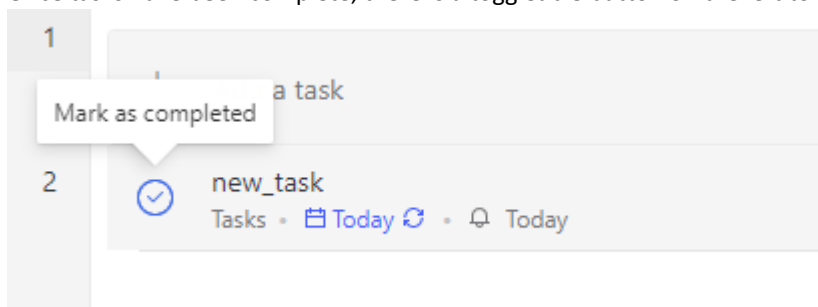
When tasks are being added, a title, due date, whether you want to be notified and a repeat option are required.



There are separate tabs for different types of tasks and the system can be integrated with your email so events are added automatically. As well as this you can make new lists which can be aimed towards projects with other team members.



Once tasks have been complete, there is a toggleable button on the left to mark the task as completed.



New tasks can have additional details added, including steps, notes, and files.

The screenshot shows the Microsoft To Do app interface. At the top, there is a task list with three items: 'g' with a star icon, 'vg' with a close icon, and 'Next step' with a blue link icon. Below the list, there are several options: 'Add to My Day' with a sun icon, 'Remind me' with a bell icon, 'Add due date' with a calendar icon, 'Repeat' with a circular arrow icon, 'Pick a category' with a tag icon, 'Add file' with a paperclip icon, and 'Add note' with a notepad icon.

Major Features:

- 1) Adding tasks: adding tasks which need to be completed on certain days which can give you notifications and can be repeated.
- 2) Assigning tasks: assigning tasks to others if you were completing group work for example.
- 3) Creating new lists: can create new lists which tasks are linked to, this can also be shared amongst a group of people.
- 4) Marking tasks as complete: completing tasks so they do not notify you/get shown as marked in the list
- 5) Login: logging into your Microsoft account which all your tasks are linked to.
- 6) Flagged emails: Shows emails which are linked to tasks, so new tasks can be added from here.
- 7) Important tasks: Tasks which have been flagged and need to be completed.
- 8) Adding steps to tasks: Add steps to tasks which are required for the task to be complete.
- 9) Add files to tasks: add files which can be edited tot asks.

Operating Environment: H/W, S/W environment

- 1) Platform: Android, iOS, Windows, web
- 2) Required HW: computer, Wi-Fi, keyboard, mouse
- 3) Required SW: OS
- 4) Any other requirements: Microsoft account

Pros:

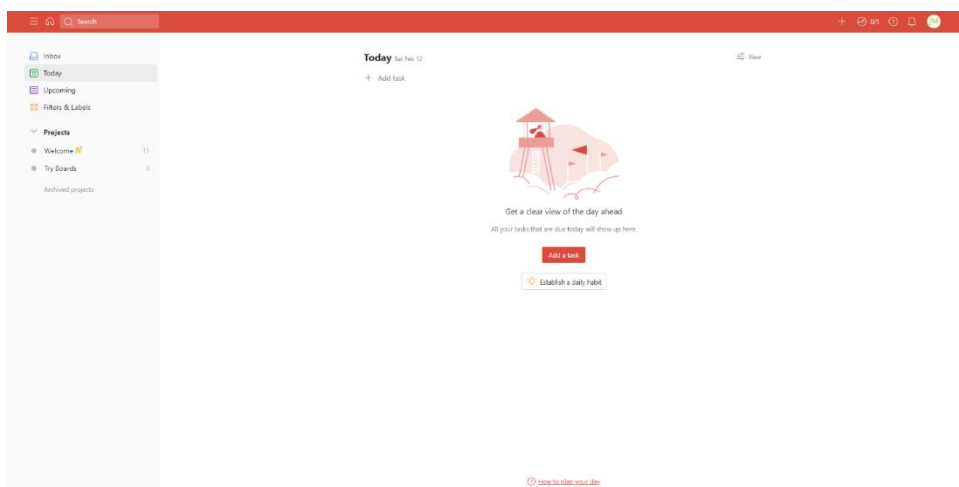
- 1) Interlink ability with other users
- 2) Linking with your email to automatically identify new tasks
- 3) Ability to add new lists and share with other users
- 4) Clear layout, so is easy to identify tasks
- 5) Daily planner to show which tasks are due/occurring today
- 6) Notifications for tasks

Cons:

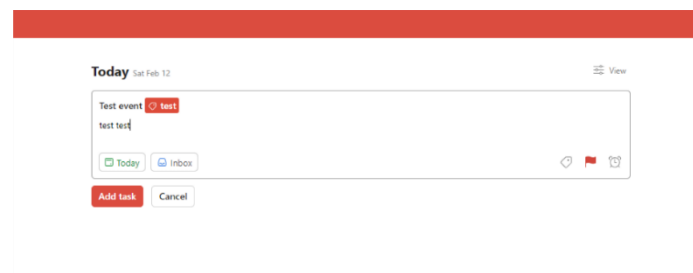
- 1) No calendar interface to show when tasks are due
- 2) Not very intuitive on how you can share/assign tasks to others
- 3) Can add multiple tasks with the same details and does not warn you – could lead to multiple of the same tasks

System Name: Todoist

General overview and highlights: The Todoist homepage are a very modern and clean space. This is great for displaying just useful information to the user without any distraction.



Navigation buttons on the left are clearly labelled to allow for easy navigation between areas where the user can create events for the current day, see upcoming events and add to certain days or filter through those and find specific events.



Creating events is easy and can be done with just a few clicks. Custom parameters can be added such as tags, which will allow for better grouping of events later. Priority can also be assigned when creating the event, again to allow for more efficient filtering.

Major Features:

- 1) Today's overview: lets the user easily and simply see all the events ahead of them today.
- 2) Upcoming: allows the user to clearly see their month with all events laid out for them.
- 3) Filters and labels: let the user see custom lists of events that are organised for their specific reasons.
- 4) Linked boards: linked lists of to-do lists are easily used to see a series of tasks that need to be completed. These tasks also offer sub-tasks for further depth to this process.
- 5) Productivity tracker to show how many tasks the user completes on average.

Operating Environment: H/W, S/W environment

- 1) Platform: Android, iOS, Windows, MacOS, Website.
- 2) Required HW: Storage (for .exe version).
- 3) Required SW: an operating system, a web browser.
- 4) Any other requirements: N/A

Pros:

- 1) Easy to use, inexperienced users require little training.
- 2) Free versions available
- 3) Offering lots of useful features, more than most competitors
- 4) Customizable themes
- 5) Cheap pro version upgrades

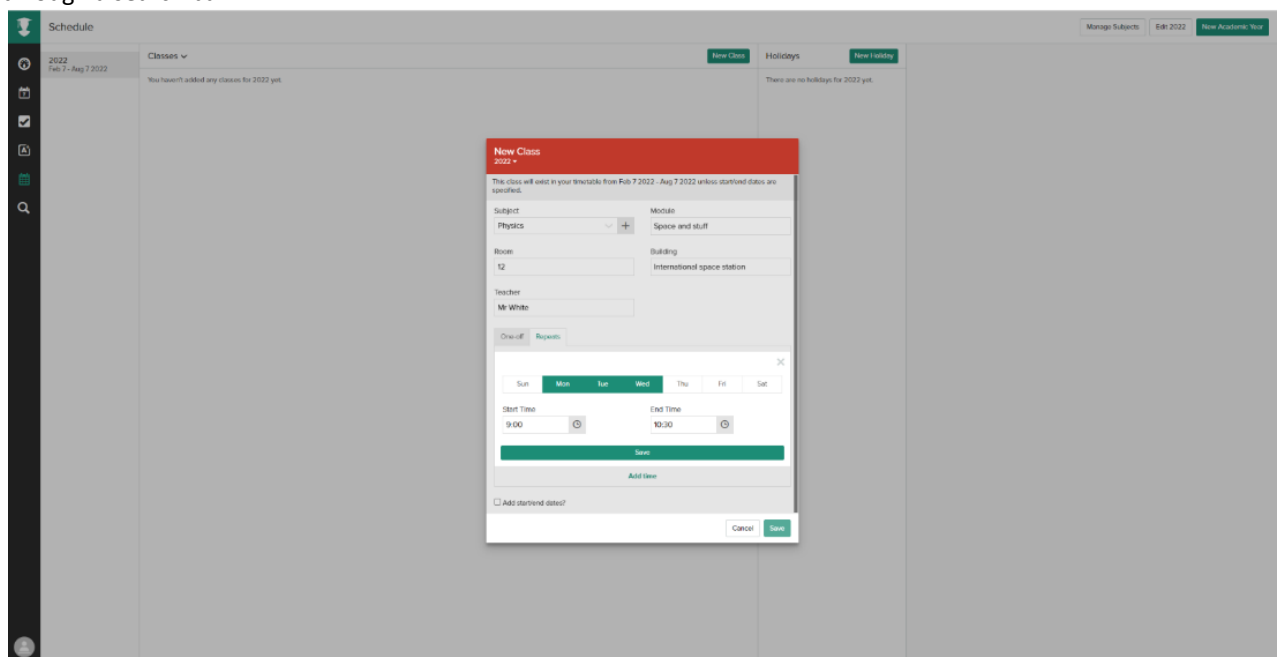
Cons:

- 1) Inaccessible for users with disabilities. I.e., no effortless way to increase font size and button size
- 2) No way to easily share your schedule with other users

System Name: mystudylife

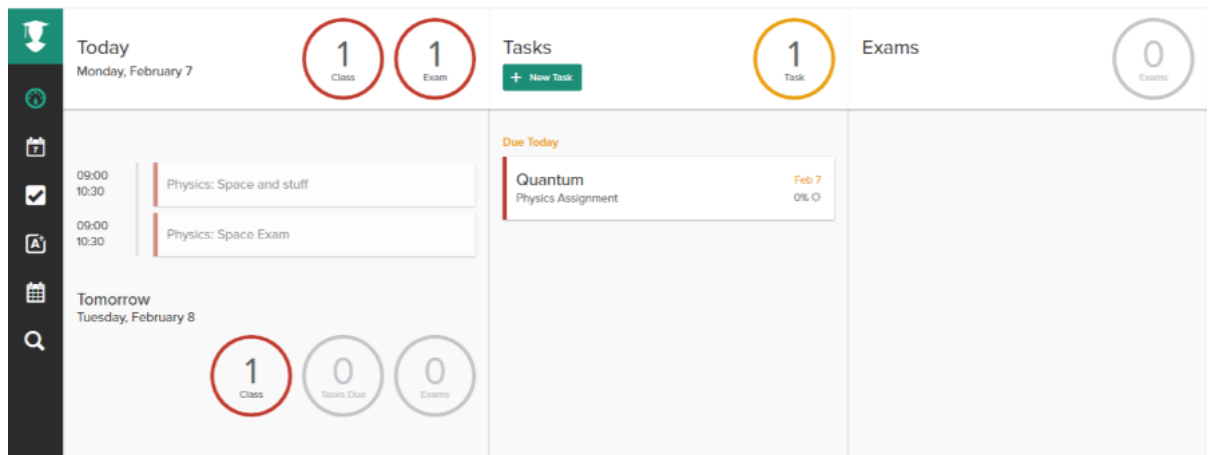
General Overview and Highlights:

This system allows users to plan and organise their academic year based on several various aspects. It gives you a set of ways to plan. Such as adding tasks and exams. You can then view these on a task list, a calendar, and through a search bar.



Major Features:

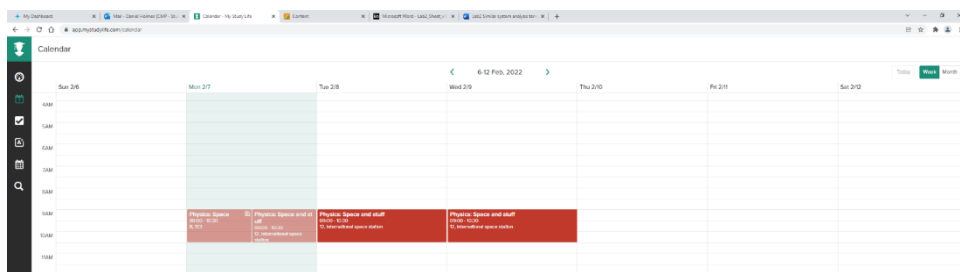
- 1) Dashboard: user is given a restricted area allowing users to organise and store their workload.



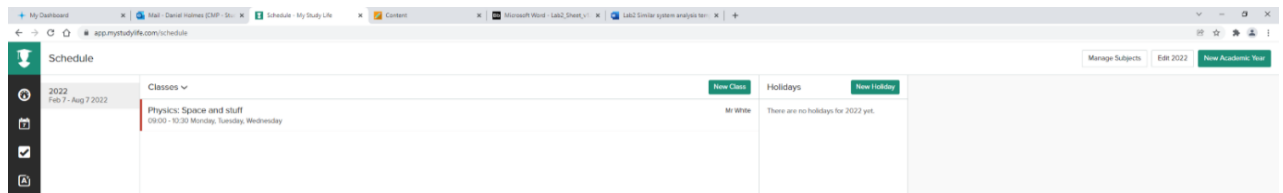
- 2) Add a task: user can define the type of work by its type either assignment, reminder, or revision. You can also assign a subject to a task. A title and description can also be added.

The 'New Task' form has a red header with the title 'New Task' and a dropdown for 'No year/term'. Below the header, there are two dropdown menus: 'Subject' (set to 'Physics') and 'Type' (set to 'Assignment'). There is a 'Due Date' field set to 'February 7 2022' with a calendar icon. Below that is a 'Title' field with the text 'Quantum'. A 'Detail' text area contains the text 'small stuff'. At the bottom right, there are 'Cancel' and 'Save' buttons.

- 3) Calendar: user can view their upcoming exams.



- 4) Schedule: users can create an academic year detailing their classes and any holidays.



- 5) Colour coded: users can choose the colour for a subject upon creating a new subject.
- 6) Sidebar: users are given quick access to various aspects.
- 7) Search bar: users can find specific work or classes by searching a key term.

Operating Environment: H/W, S/W environment

- 1) Platform: Windows, IOS, Android
- 2) Required HW: Computer
- 3) Required SW: A web browser
- 4) Any other requirements: N/A

Pros:

- 1) It is easy to use and requires little to no training
- 2) It is clear and colour coded
- 3) Can find/access tasks, classes etc easily

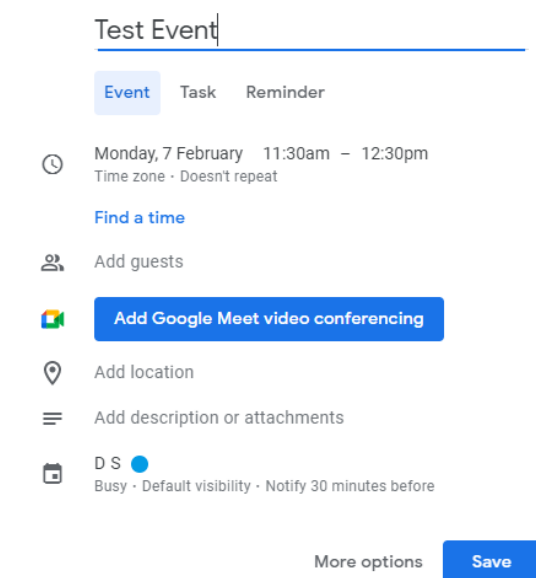
Cons:

- 1) Cannot share a schedule with other users.

System Name: Google Calendar

General Overview and Highlights:

The google calendar app allows for users to access their own private calendar system. This allows users to create events, reminders, and tasks visually. The system requires a google account and therefore has a lot of google integration support, such as maps, contacts, and other add-ons.



Month

Day

Week

Month

Year

Schedule

4 days

D

W

M

Y

A

X

✓ Show weekends

✓ Show declined events

Calendar

Today

<

>

February 2022

🔍

🕒

⚙️

Week

D

Create

SUN

MON

TUE

WED

THU

FRI

SAT

GMT+00

6

7

8

9

10

11

12

February 2022

<

>

S

M

T

W

T

F

S

30

31

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

1

2

3

4

5

6

7

8

9

10

11

12

🔍 Search for people

My calendars

☑ D S

☑ Birthdays

☑ Reminders

☑ Tasks

Other calendars

☑ Holidays in United Kingdom

1 AM

2 AM

3 AM

4 AM

5 AM

6 AM

7 AM

8 AM

9 AM

10 AM

11 AM

12 PM

1 PM

2 PM

3 PM

4 PM

5 PM

6 PM

7 PM

8 PM

📅 Test Task, 9:45am

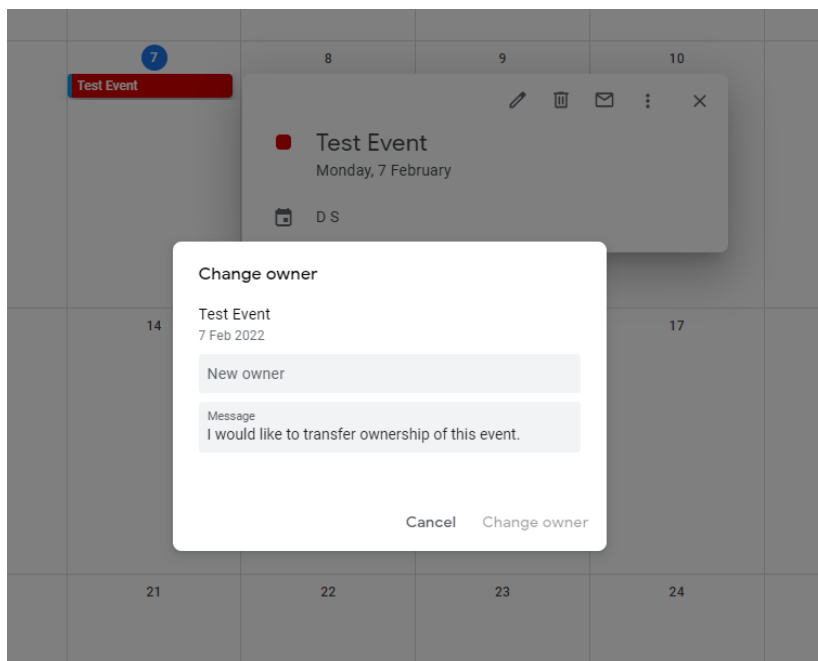
📅 Test Event
2:15 - 7:30pm

My calendars

- ☒ D S
- ☒ Birthdays
- ☒ New Calendar
- ☒ Reminders
- ☒ Tasks

Other calendars

- ☒ Holidays in United Kingdom



Major Features:

- 1) Adding Events/Tasks/Reminders: Allows the user to create visual reminders on the calendar interface. This can be in the form of an event, a task or a reminder. The user can setup a notification for these, invite others to have access to said events
- 2) Changing timeframe: The user can change the timeframe of their calendar between a day/four days/week/month/year and schedule view. A schedule view only shows events that have been created or holidays that are on the calendar by default.
- 3) Contact integration: The google integration allows for the user to directly access their phone contacts at a click, assuming they have a connected mobile number on their account.
- 4) Creating multiple calendars: The user can add custom calendars which have their own dedicated events. These can be toggled on and off depending on what the user wants to access.

Operating Environment: H/W, S/W environment

- 1) Platform: Android, iOS, Web
- 2) Required HW: Screen, keyboard, internet access
- 3) Required SW: Web app or app store app

4) Any other requirements: Google account

Pros:

- 1) Allows support for multiple calendars.
- 2) Easily share events with contacts or any Gmail account. Can also transfer control of events to any account. The user who takes control of the event has full control of it as it becomes part of their calendar.
- 3) Can install a wide range of Google Workspace Marketplace add-ons to fully customise the user experience.

Cons:

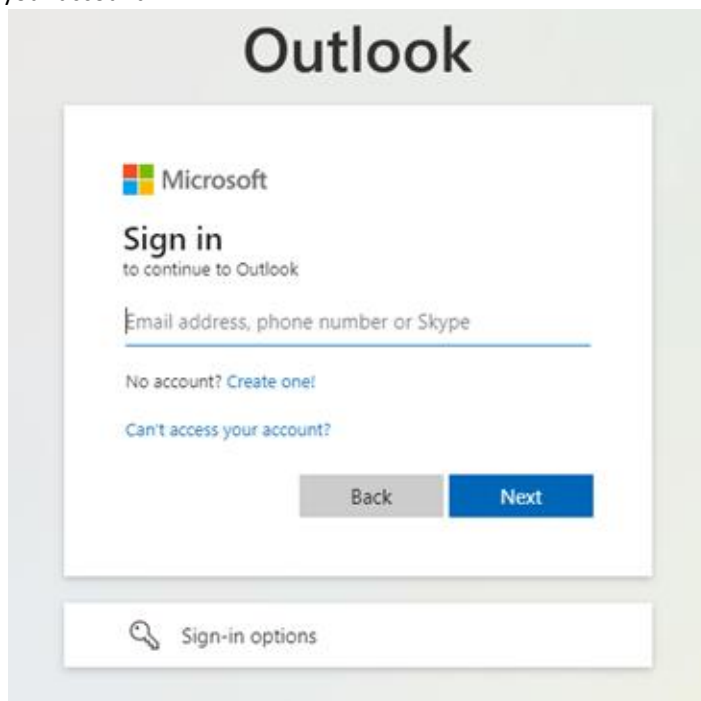
- 1) A Brand-new users may not find these features intuitively. I could not easily locate the 'changing timeframe' feature as I had accidentally swapped from 'week' to 'day'

System Name: Microsoft Outlook

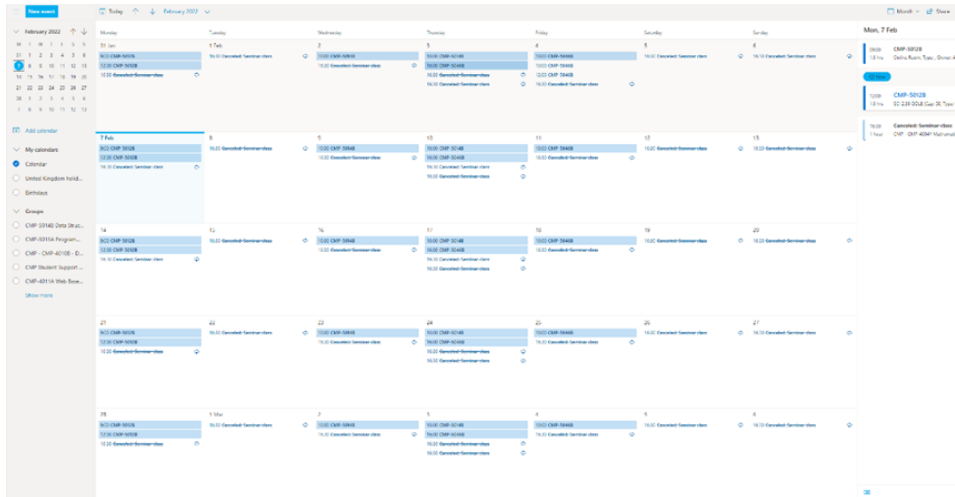
General Overview and Highlights:

The calendar on Outlook is a cloud-based calendar, showing all scheduled lectures, seminars, and assessments.

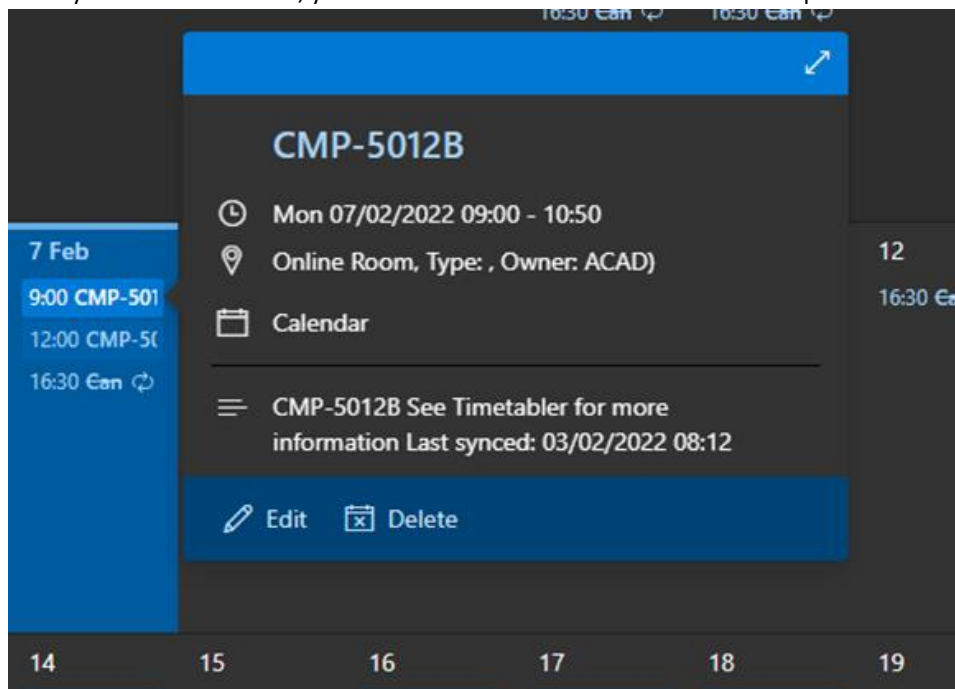
You access your calendar by signing in with an email. There are password recovery options if you cannot access your account.



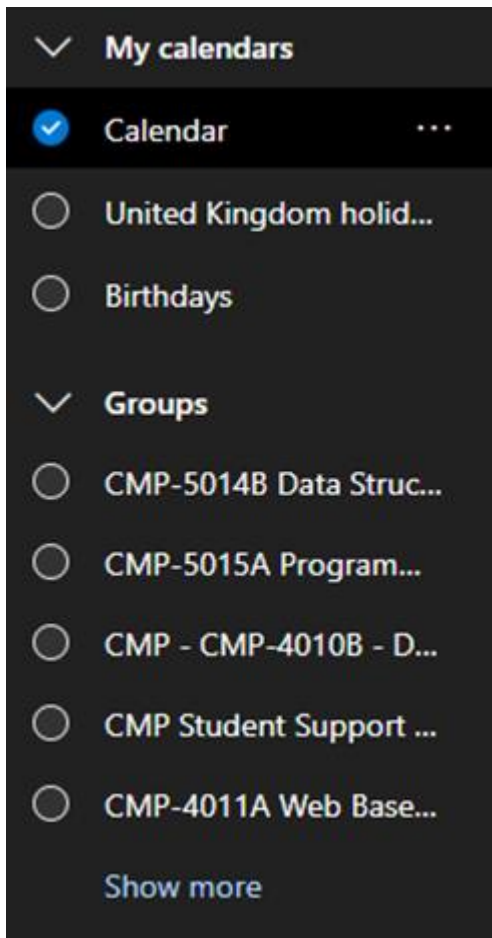
The main UI shows a month at a time. You can browse different months using the control bar on the right-hand side.



When you click on an event, you can view more details. There is also the option to edit and delete the event.



The search bar at the top allows for event searching. This is done by name. There are more event filtering options on the right of the UI, this allows you to filter by calendars and groups.



Major Features:

1. Event filtering: You can choose to only show events of a certain type or events belonging to a certain calendar.
2. Event searching: There is a search bar at the top of the page that allows for event searching by title
3. Event and calendar sharing: Calendars can be shared by email.
4. User login: Users log on using an email and password.
5. Password recovery: if a user cannot remember their password, they can recover it by sending a reset link to another email or a code to a phone.
6. Changeable time zones, date formats, time format and first day of the week: There is the option to change the time zone and therefore the times of events on the calendar. You can also change the date format, for example to the American format of MM/DD/YYYY

Operating Environment: H/W, S/W environment

1. Platform: Android, iOS, Windows, Mac OS
2. Required HW: Computer with basic peripherals or phone
3. Required SW: Chrome or the Outlook app
4. Any other requirements: An internet connection

Pros:

1. There is the ability to filter by event creator as well as by groups.
2. As well a view of the whole month, there is a side bar showing what events are happening throughout the current day.
3. You can share a calendar by email
4. You can print a calendar
5. The mobile version has the option of a widget, allowing you to view your calendar without opening the app.
6. You can receive notifications to remind you of upcoming events.
7. You can search for events using the search bar

Cons:

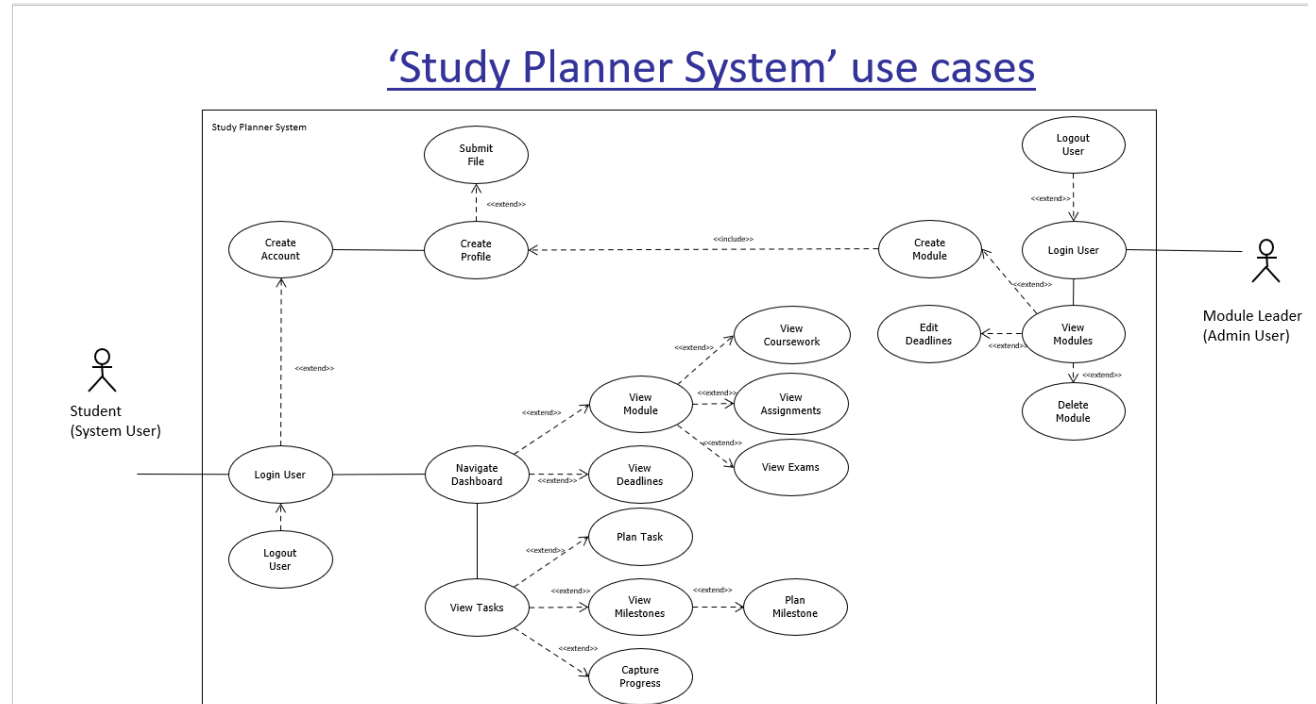
1. The calendar is reliant on an internet connection
2. Cancelled events still appear until manually deleted

1.3. Feature matrix of similar systems

Feature name	<i>Microsoft to do</i>	<i>Todoist</i>	<i>Mystudy life</i>	<i>Microsoft Outlook</i>	<i>Google Calendar</i>	Our System
User Login	Yes	Yes	Yes	Yes	Yes	Yes
Calendar view	No	Yes	Yes	Yes	Yes	Yes
Linking with other contacts	Yes	No	No	Yes	Yes	No
Email integration	Yes (with task prediction)	Yes	No	Yes	Yes	Yes
Works on all popular platforms	Yes	Yes	Yes	Yes	Yes	No
App available (Mobile)	Yes	Yes	Yes	Yes	Yes	No
Social network integration	No	No	No	Partial	Yes	No
Supports other apps or add-ons	Yes (connects with outlook/microsoft)	No	No	Connects to other Microsoft apps	Yes	No
Works without internet access	No	No	No	No	Yes, to an extent	No
Event filtering	Yes	Yes	No	Yes	Yes	Yes
Password Recovery	Yes	Yes	Yes	Yes	Yes	Yes
Changeable time zones	No (adapts to local time zone)	Works with local time zone	Yes	Yes	Works with local time zone	Works with local timezones

2. Requirements Analysis [40% marks]

2.1 Use Case Diagram (make sure you cover all features in the project spec.)



2.2 Use Case Table (minimum of five use cases)

Use case name:	1. Create Account
Scope:	Study Planner System
Primary actor:	System user
Secondary actors:	None
Summary:	The user creates an account with the student planner system.
Preconditions:	The user does not have an account with the student planner system
Main success scenario:	<ol style="list-style-type: none"> 1. The user enters a university email address 2. The user enters personal information such as name, date of birth and course. 3. The user enters their password of choice. 4. The user submits the form, a message then informs the user that the form has successfully submitted. 5. The account details are stored on a database
Alternatives:	<p>1a. The user's email is incorrect, they must check they have entered the correct email.</p> <p>2a. The user enters an invalid date for their date of birth (such as a date in the future). They must correct this before a form can be successfully submitted.</p>

	3a. The password doesn't meet the minimum-security requirements. The user must change the password so that all security requirements are satisfied before submitting the form.
Exceptions:	1a. The users email address is already associated with an account; they must contact an admin user.
Postconditions:	The user has created an account with the student planner system

Use case name:	2. Login User
Scope:	Study Planner System
Primary actor:	System user
Secondary actors:	None
Summary:	The user enters their email and password to gain access to their account.
Preconditions:	The user must've created an account.
Main success scenario:	<ol style="list-style-type: none"> 1. The user enters their email and password into their respective textboxes 2. The user clicks the login button and is successfully taken to the main UI of the planner
Alternatives:	<p>1a. The user enters an incorrect email. An error message informs the user. The user then enters a correct email.</p> <p>1b. The user enters a password that doesn't match the email entered. The user then enters the correct password.</p>
Exceptions:	<p>1a. The email is not associated with any account; therefore, the user will not be able to gain access to the planner.</p> <p>1b. The user doesn't enter the associated password and does not gain access to the planner.</p>
Postconditions:	The user gains access to the planner associated with the account details they have just entered.

Use case name:	3. Logout User
Scope:	Study Planner System
Primary actor:	Study planner system users

Secondary actors:	None
Summary:	The user presses a button to logout of their account and return to the login screen.
Preconditions:	The user must be logged in.
Main success scenario:	<ol style="list-style-type: none"> 1. The user presses the logout button and are no longer logged in to their account on the device they are using. 2. The login page is loaded.
Alternatives:	None
Exceptions:	None
Postconditions:	The user is logged out.

Use case name:	3. Create Module
Scope:	Study Planner System
Primary actor:	Admin user
Secondary actors:	Student user
Summary:	The admin user creates a module that users can add to their study profile
Preconditions:	Admin user must have an account
Main success scenario:	<ol style="list-style-type: none"> 1. The user navigates to the modules page. 2. The user selects the create module button. 3. The user enters the module name, subject and duration details. 4. The user adds all deadlines and tasks to the module details. 5. The user confirms the addition of the module. 6. The module now appears when a user is creating their study profile
Alternatives:	<p>3a. The module name already exists. The user must enter a different module name.</p> <p>3b. The module starts on a date in the past. The user must enter a current of future start date.</p>
Exceptions:	3a. The user didn't realize the module already exists and aborts the creating of a new module.
Postconditions:	The new module appears on the list of modules when creating a study profile.

Use case name:	4. Edit Module
----------------	----------------

Scope:	Study Planner System
Primary actor:	Admin user
Secondary actors:	None
Summary:	The creator of a module wants to amend details of a module they created.
Preconditions:	The user must be logged in The user must have created a module to edit
Main success scenario:	<ol style="list-style-type: none"> 1. The user navigates to the module page. 2. The user selects the module they would like to edit. 3. The user selects the 'edit module' button. 4. The user changes any details regarding name, duration, subject, associated tasks, description etc... 5. The user confirms the changes.
Alternatives:	4a. The user makes no changes and therefore does not need to save anything.
Exceptions:	None
Postconditions:	The module selected has been amended and the changes appear on the profiles of all users who have enrolled on the module.

Use case name:	5. Delete Module
Scope:	Study Planner System
Primary actor:	Admin user
Secondary actors:	Users with the same module in their calendar
Summary:	A user deletes a module. The module is marked as cancelled and tasks related to the module are also marked as cancelled on the task list of all users who have added the module.
Preconditions:	The admin user must have created a module.
Main success scenario:	<ol style="list-style-type: none"> 1. The user navigates to the list of modules. 2. The user selects a module to delete. 3. The user presses the 'delete module' button. 4. The user confirms they want to delete the module. 5. The event is removed and all users who enrolled on the module have their task list updated.
Alternatives:	5a. No user enrolled and no updates are made to the task lists.
Exceptions:	4a. The user is not the module creator, the module is not deleted.

	4b. The user does not confirm, and the module is not deleted
Postconditions:	The module is removed from the list of available modules when creating a profile. The users who have enrolled have any tasks relating to the module removed from their task list.

Use case name:	6. Navigate Dashboard
Scope:	Study Planner System
Primary actor:	Study planner system users
Secondary actors:	None
Summary:	The user navigates their way around the main UI of the study planner system.
Preconditions:	User is logged in
Main success scenario:	<ol style="list-style-type: none"> 1. The user is welcomed to the home page where there is an overview of all modules, tasks, exams, assignments, and coursework 2. The user selects the 'modules' button and is taken to the page where they can view modules in detail 3. The user selects the 'Tasks' button and can view tasks page where they can view tasks in more detail.
Alternatives:	<p>2a. The user can navigate to the 'tasks' section from 'modules'.</p> <p>3a. The user can navigate to the 'modules' section from the 'tasks' section.</p>
Exceptions:	None.
Postconditions:	The user has navigated the whole site and logged out.

Use case name:	7. Create Profile
Scope:	Study Planner System
Primary actor:	Student user
Secondary actors:	None
Summary:	The user creates a new study profile. The user selects a file that contains the data used to create the study profile. This includes activities, milestones, deadlines etc...

Preconditions:	The user is logged in
Main success scenario:	<ol style="list-style-type: none"> 1. The user presses the new study profile button 2. The user provides an identifier for the semester 3. The system prompts the user for a semester file which is provided by the hub 4. If successful, a profile is created
Alternatives:	None
Exceptions:	2a. User cannot select relevant semester identifier 4a. File upload failed (no file or corrupted file)
Postconditions:	The user has created a new study profile.

Use case name:	8. Plan Task
Scope:	Study Planner System
Primary actor:	Student user
Secondary actors:	None
Summary:	The user plans a task relating to a module and its exams, coursework and assignments.
Preconditions:	The user is logged in A data file for at least one module has been loaded.
Main success scenario:	<ol style="list-style-type: none"> 1. The user selects the create task button. 2. The user selects the module the task belongs to. 3. The user selects what the task is (studying, programming, writing etc...). 4. The user enters the criterion on which the task completion is assessed. 5. The user adds any necessary notes. 6. The user saves the task.
Alternatives:	5a. There are no notes to add and are therefore left blank.
Exceptions:	6a. The user decides not to save the task and discards it by pressing the 'discard task' button.
Postconditions:	The task is added to the users complete list of tasks.

Use case name:	9. Plan Milestone
Scope:	Study Planner System

Primary actor:	Student user
Secondary actors:	None
Summary:	The user creates a milestone (sub-task) relating to a certain task. This is so the user can break down a task into smaller components with their own individual deadlines.
Preconditions:	The user is logged in The user must have created a task
Main success scenario:	<ol style="list-style-type: none"> 1. The user selects a task and presses the 'create milestone' button. 2. The user enters the title, start date, end date and notes. 3. The user saves the milestone.
Alternatives:	None
Exceptions:	<ol style="list-style-type: none"> 1a. There are no tasks to create a milestone for. No milestones are added. 3a. The user decides not to create the milestone and presses the 'discard milestone' button.
Postconditions:	Milestone associated with an event is added to the users list of milestones.

Use case name:	10. Capture Progress
Scope:	Study Planner System
Primary actor:	Student user
Secondary actors:	None
Summary:	The user can edit the amount of progress they have made on a given task or milestone.
Preconditions:	The user is logged in The user has loaded a course profile The user has created a task
Main success scenario:	<ol style="list-style-type: none"> 1. The user selects the task they want to capture progress for. 2. The user enters the amount of time they have spent on the task in each session. 3. The user adds any additional notes to the activity. 4. The user confirms the progress by clicking the 'save' button. 5. The time is added to the total completion of the primary task and all related tasks. 6. The total completion is updated.
Alternatives:	3a. The user does not add any additional notes

	6a. If the total completion matches or exceeds the time goal, the task is marked as complete.
Exceptions:	4a. The user decides to discard these updates by pressing the 'discard progress' button.
Postconditions:	The user updates the progress of a given task.

Use case name:	11. View deadlines
Scope:	Study Planner System
Primary actor:	Student user
Secondary actors:	Admin user
Summary:	The user can look at exams and coursework deadlines for the modules they are enrolled in.
Preconditions:	The user must be logged in. The user must be enrolled in a module The module organizer must have added details of deadlines to the module.
Main success scenario:	1. The user navigates to deadlines section of the UI. 2. The user selects the module they want to view deadlines for.
Alternatives:	None
Exceptions:	2a. The user has had all their modules cancelled and has no deadlines to view.
Postconditions:	The user views the deadline details they were looking for

Use case name:	12. Edit deadlines
Scope:	Study Planner System
Primary actor:	Admin User
Secondary actors:	Student user
Summary:	The module leader edits a deadline of a task, exam or piece of coursework relating to a deadline.
Preconditions:	The user must have created a module The module must have an associated task, exam, or piece of coursework

Main success scenario:	<ol style="list-style-type: none"> 1. The user navigates to the modules page. 2. The user selects the module they want to edit. 3. The user selects the 'edit deadlines' button. 4. The user selects the specific task to edit. 5. The user amends the details of the task. 6. The user saves the changes. 7. All changes appear to students enrolled on the module.
Alternatives:	6a. The user decides not to change anything and presses the 'discard changes' button.
Exceptions:	None
Postconditions:	The module leader has successfully changed details of the deadline.

Use case name:	13. Submit file
Scope:	Study Planner System
Primary actor:	Student User
Secondary actors:	None
Summary:	The user submits a file into the system. The file determines what information is shown to the user.
Preconditions:	Prompted by the system to submit a file as part of profile creation.
Main success scenario:	<ol style="list-style-type: none"> 1. The user selects a file 2. System checks validity of file 3. The application creates a content area to display activities, milestones, deadlines etc...
Alternatives:	<p>2a. If the file is invalid, an error is raised. The user is asked to provide a new file.</p> <p>2b. The user provides a valid file and the main success scenario resumes.</p>
Exceptions:	2b. The user fails to provide a valid file. The profile is not created.
Postconditions:	The user will have access to their activities, milestones, deadlines and assessments.

Use case name:	14. View Modules
----------------	------------------

Scope:	Study Planner System
Primary actor:	Student User
Secondary actors:	None
Summary:	The user views all pieces of coursework they have been assigned.
Preconditions:	The user is logged in. The user is enrolled on a module The module has coursework assigned to it.
Main success scenario:	<ol style="list-style-type: none"> 1. The user clicks the modules section of the main UI 2. The user can now see all modules they have enrolled in.
Alternatives:	2a. The user can click on a specific module to view its details, assignments, coursework and exams.
Exceptions:	The user has not enrolled in any modules and therefore has nothing to view.
Postconditions:	The user has viewed all their modules.

Use case name:	15. View coursework
Scope:	Study Planner System
Primary actor:	Student User
Secondary actors:	None
Summary:	The user views all pieces of coursework they have been assigned.
Preconditions:	The user is logged in. The user is enrolled on a module The module has coursework assigned to it.
Main success scenario:	<ol style="list-style-type: none"> 1. The user navigates to the module they want to view. 2. The coursework will be listed along with other module details.
Alternatives:	2a. The user can click on a specific piece of coursework to view its details.
Exceptions:	2a. The user has no coursework to view as none has been assigned.
Postconditions:	The user has viewed all assigned coursework.

Use case name:	16. View assignments
Scope:	Study Planner System
Primary actor:	Student User
Secondary actors:	None
Summary:	The user can view all assignments belonging to a particular module.
Preconditions:	The user is logged in. The user is enrolled on a module The module has assignments assigned to it.
Main success scenario:	1. The user navigates to the module they want to view. 2. The assignments will be listed along with other module details.
Alternatives:	2a. The user can click on a specific assignment to view its details.
Exceptions:	2a. The module has no assignments for the user to view.
Postconditions:	The user has viewed all assignments.

Use case name:	17. View exams
Scope:	Study Planner System
Primary actor:	Student User
Secondary actors:	None
Summary:	The user can view all exams scheduled for a certain module.
Preconditions:	The user is logged in. The user is enrolled on a module The module has exams scheduled to it.
Main success scenario:	1. The user navigates to the module they want to view. 2. The exams scheduled will be listed along with other module details.
Alternatives:	2a. The user can click on an exam to view its specific details.
Exceptions:	2a. The module has no exams scheduled for the user to view.
Postconditions:	The user has viewed all exams scheduled.

2.3 Shall-Statements

2.3.1 First Sprint Features (Weeks 6 – 8) *(must include minimum of 3 NFRs)*

S1.1: A new user shall be able to create an account (FR).

Description: There will be an account creation section for users to create an account by registering their details.

S1.2: The system should output a message telling the user if the account has been created successfully. (NFR)

Description: Upon the successful creation of an account, the user will receive a confirmation.

S1.3: The system shall check if the email is already associated with an account in the system, prompting for a correction before allowing registration to proceed if there is a pre-existing account registered with the email address provided by the user. (FR)

Description: A message is displayed telling the user that the email is already linked to an account and there is a link provided to an account recovery page. Allowing the user to either return to the login page or the account recovery page.

S1.4: The system shall check that all the required details for creating an account have been entered before allowing registration to proceed. (FR)

Description: Any input box with missing form data will be highlighted in red.

S1.5 The user's account details shall be stored by the system (FR)

S2.1: An existing user shall be able to login to their account (FR).

Description: There will be a login page for users to enter their details. If the user enters the correct details, then they are logged in to their account and taken to their personal area (their dashboard).

S2.2: The system shall output an error message if unable to verify a user (log them in due to incorrect details) (FR).

Description: A message is displayed saying which parts of the login details they have tried to submit are incorrect.

S3.1: An existing user that is logged in shall be able to logout of their account (FR).

Description: There will be a sign out button for users, when pressed logs them out of their account.

S3.2: The system should send a confirmation that the user has logged out (NFR).

Description: Once the user has been signed out, the system should output a confirmation that they have signed out.

S4.1: A user shall be able to create a study profile (FR).

Description: The user clicks the "new study profile" button, they are then able to provide an identifier (name) for the semester/profile and select the semester year. The system will then prompt the user to submit this file, thereby creating a study profile.

S4.2: The system shall tell the user if the study profile file was unsuccessful in uploading to the system (FR).

Description: The system returns an error if the user tries to create a study profile and is unsuccessful in doing so.

S4.3: The system should allow the user to upload a different study profile file if the study profile file was unsuccessful in uploading to the system (FR).

Description: The user can either upload a different file or exit the create profile window (and no profile will be created).

S5.1: A user shall be able to plan a task (FR).

Description: The user selects the "create task" button. The user selects which module this task applies to. The user enters details about the task (what it is, how it is assessed and any notes). The user can then save the task, the system then adds this new task to the associated module.

S5.2 The system shall store tasks (FR)

Description: The system shall store any tasks created by the user associating them with the user's account.

S6.1: A user shall be able to remove a task (FR).

Description: The user finds the task they wish to remove, clicking on the task to see its details. The user can then click the remove task button and are prompted by the system to confirm the task's removal. Once confirmed, the system removes the task.

S7.1: A user shall be able to create a module (FR).

Description: The user clicks the create new module button. They then enter the details for the module (name, subject and duration). The user then adds any deadlines and tasks to the module details. The user confirms all details are correct and the module is created.

S7.2: The system should check whether the module already exists (NFR)

Description: If there is already a module stored under the same name as the module trying to be created, a warning message is output stating that there is already a module with this name.

S7.3: The system should allow the user to rename the module or exit the creation of the module (FR).

Description: If there is already a module stored under the same name as the module trying to be created, the user is given the option to rename the module being created or cancel the new modules creation.

S7.4: The system shall store modules (FR)

Description: The system shall store any modules created by the user associating them with the user's account.

S8.1: An admin shall be able to delete a module (FR)

Description: The user clicks the "delete module" button. The system then deletes the module, and all users enrolled on the module have their calendars and associated lists (task) updated.

S8.2: The system should issue a warning to confirm the deletion of a module (NFR)

S9.1 A user shall be able to view modules (FR)

Description: The system allows users to view any modules associated with their account.

2.3.2 Test Cases for First Sprint

Test ID	Feature ID and Brief Description	Input Operations (Sequence and data)	Expected Output (text and/or drawing of the outputs)
T1.1	Create Account: register account details	1. User navigates to account creation section. 2. User enters university email address and other personal data. 3. User enters their password of choice and re-enters this. 4. User submits the form; a message will appear telling the user the account has been created successfully.	A new account is created for the user.
T1.2	Create Account: email is linked to an existing account	1. User submits the registration form. 2. A message is displayed telling the user that the email is already linked to an account and there is a link provided to an account recovery page. 3. User returns to login page or account recovery page.	The user is informed that the email is already linked to an account.
T1.3	Create Account: incomplete registration form	1. User submits the registration form. 2. Form highlights the boxes in red which are incomplete/incorrect.	The user is informed which parts of the registration form they haven't filled in or have filled in incorrectly.
T2.1	Login User: logging in	1. The user navigates to the login page. 2. The user enters their account name/email. 3. The user enters their password. 4. 'Log in' is clicked and the user will be on their dashboard.	User can log into their account.
T2.1.1	Login User: admin account	1. The admin navigates to the login page. 2. The admin enters their special account details. 3. 'Log in' is clicked and the user will be on the admin dashboard.	Admin user will be logged into their account.
T2.2	Login User: incorrect detail	1. User attempts to log in.	User tried to log in to their account and is told which detail is incorrect.

		<p>2. A message is displayed saying which detail is incorrect (email or password)</p> <p>3. User retries logging in</p>	
T3.1	Logout User: logging out	<p>1. The user will click on their icon in the bottom left-hand side of the screen.</p> <p>2. The user will click 'sign out' and this will log them out of their account.</p> <p>3. A confirmation will be displayed telling the user they have been signed out.</p>	The user will be signed out of their account and will no longer be able to access their timetable.
T4.1.1	Create Profile	<p>1. User clicks the "new study profile" button.</p> <p>2. User provides an identifier (name) for the semester/profile.</p> <p>3. User selects the semester year relevant to their study semester.</p> <p>4. User submits a file once prompted by the program (this file is provided by the hub).</p>	The user creates a new profile for the semester.
T4.1.2	Create profile: incorrect file	<p>1. User is in the create profile page and tries to upload a file once prompted by the system.</p> <p>2. An error is returned as the file is incorrect.</p> <p>3. The user can either upload a different file or exit the create profile window (and no profile will be created).</p>	The user has provided an incorrect file for the creation of a profile and the system provides a warning once this has occurred.
T5.1	Plan Task	<p>1. The user selects the "create task" button.</p> <p>2. The user selects which module this task applies to.</p> <p>3. The user enters details about the task (what it is, how it is assessed and any notes).</p> <p>4. The user saves the task.</p> <p>5. The task is now added to the associated module.</p>	A task is created for the user and assigned to the specific module.
T6.1	Remove task	<p>1. User navigates to the task they wish to remove.</p> <p>2. User clicks on the task to see its details.</p> <p>3. User clicks the "remove task" button.</p> <p>4. User confirms they would like to remove the task.</p>	A user removes a task.
T7.1.1	Create Module	<p>1. The admin navigates to the modules page.</p>	A module is created, meaning when students create a study

		<p>2. The admin clicks “create new module” button.</p> <p>3. The admin enters the module details (name, subject and duration)</p> <p>4. The admin adds all deadlines and tasks to the module details.</p> <p>5. The admin confirms all details are correct and creates the module.</p>	profile this module will now appear.
T7.1.2	Create module: module name already exists	<p>1. The admin tried to create the module.</p> <p>2. A warning message is displayed advising the admin that a module of the same name already exists.</p> <p>3. The admin either renames the module or exits the creation.</p>	A warning is displayed to the admin if a module of the same name already exists, the admin can then either rename the module or exit creation.
T7.1.3	Create module: module starts in the past	<p>1. The admin tried to create the module.</p> <p>2. A warning message is displayed telling the admin that the module starts in the past.</p> <p>3. The admin can either exit the module creation page or update the module start date or confirm they want the module to start in the past.</p>	The admin is provided with a warning that the module they are creating starts in the past.
T8.1	Delete Module	<p>1. The admin navigates to the module page.</p> <p>2. The admin selects the module they would like to delete and clicks the “delete module” button.</p> <p>3. A warning shows that deleting this module will affect all other users that have this module in their timetables.</p> <p>5. The admin confirms they want to delete this module.</p> <p>6. The module is removed, and all users enrolled on the module have their calendars and associated lists (task) updated.</p>	An admin user deletes a module which is then removed for all other associated users.
T8.2	Delete module: not module creator	<p>1. The admin confirms they want to delete the module.</p> <p>2. A system message tells the admin that the module is not deleted as they were not the creator of the module.</p>	A admin tried to delete a module that they were not the creator of.
T9.1	View Modules	1. The user clicks on the modules section of the UI once logged in.	A user is able to view their modules.

		2. The user can view their modules.	
--	--	-------------------------------------	--

2.3.3 Second Sprint Features (Weeks 9 – 11) *(must include minimum of 2 NFRs)*

S10.1: The system shall have a dashboard (FR).

Description: The main homepage area

S10.2: A user shall be able to navigate their dashboard (FR).

Description: When a user is logged in, they should be taken to the homepage. They should then be able to access any an overview of modules, tasks, exams, assignments, and coursework associated with their account.

S10.3: A user shall be able to browse deadlines (FR)

S10.4: The system shall have a calendar (FR)

Description: This is where the deadlines of tasks will be shown.

S10.5: The system shall filter and display specific deadlines on the calendar (FR)

Description: Users select the module they want to see the deadlines for. The system then filters these displaying only the deadlines for the module selected on the calendar.

S11.1: The user shall be able to view tasks (FR).

Description: The system will allow the user to see any tasks associated with a selected module.

S12.1: The user shall be able to plan milestones (FR).

Description: The user selects a task which they have previously created and clicks the “create milestone” button. The user enters the title, start date and end date of the milestone. The user adds any notes to the milestone.

S12.2: The system shall save any milestones created by the user (FR)

S13.1: A user shall be able to view milestones (FR)

Description: Once a task is selected, the system should show the user all the milestones for that specific task.

S14.1: A user shall be able to capture progress (FR)

Description: Users can select a task and enter the amount of time they have spent on this task. This is then added to the total completion for the primary task and any related tasks.

S14.2: The system shall store the captured progress

Description: The system will update the total completion of a task so far.

S15.1: An admin shall be able to edit modules (FR)

S15.2: The system should prompt the admin user to make sure edits are correct (NFR)

S15.3: The system shall save any edits made to modules (FR)

S16.1: A user should be able to view deadlines (FR)

S17.1: An admin user shall be able to edit deadlines (FR)

S18.1: A user should be able to view coursework (FR)

Description: A user navigates to the module they want to see the coursework of. The coursework is then listed by the system

2.3.4 Test Cases for Second Sprint

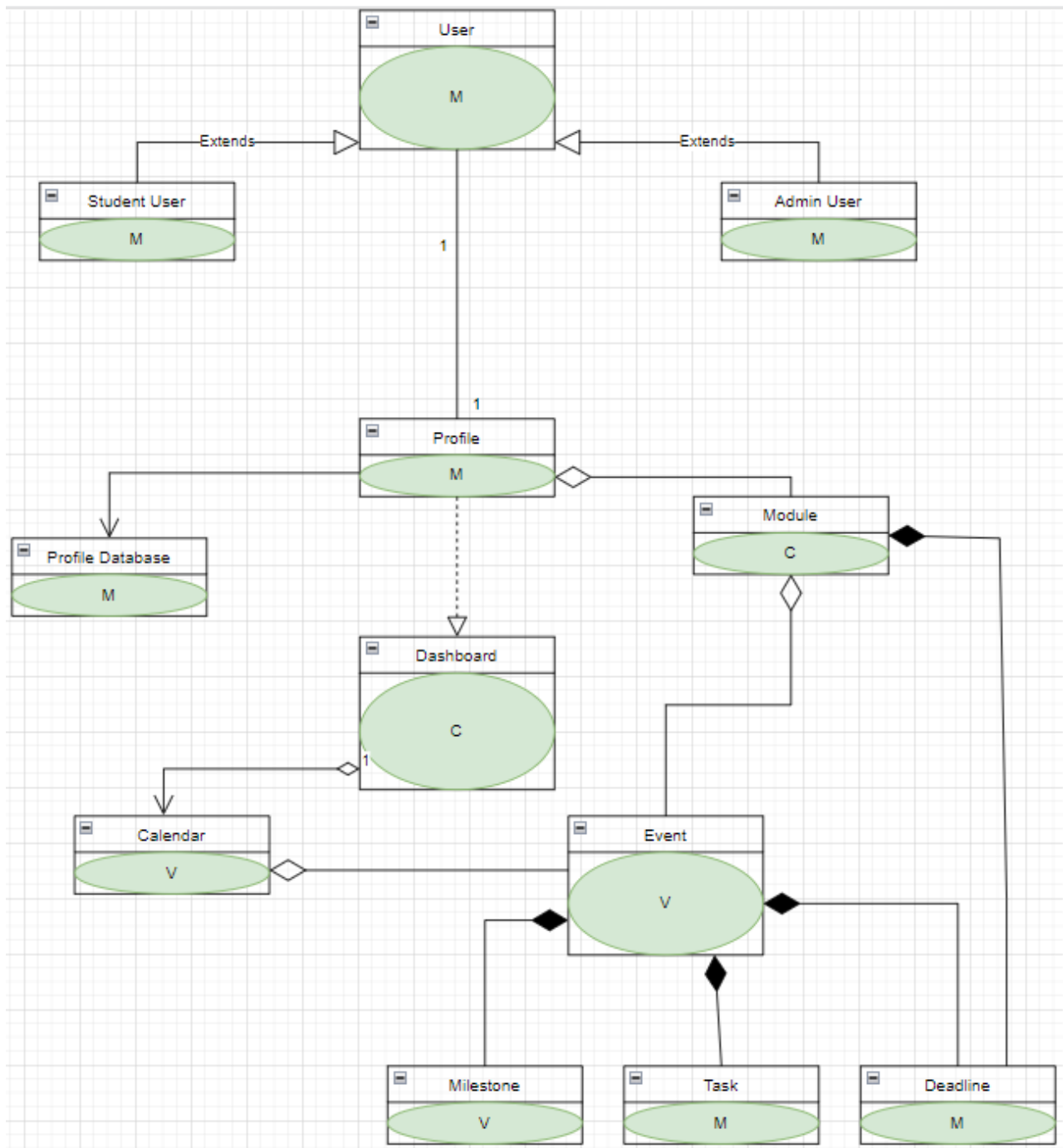
Test ID	Feature ID and Brief Description	Input Operations (Sequence and data)	Expected Output (text and/or drawing of the outputs)
T1.1	Navigate dashboard	1. The user logs into their account and is welcomed to the homepage. 2. On the homepage there is an overview of modules, tasks, exams, assignments, and coursework.	The user can view their dashboard.
T1.2	Navigate Dashboard: browse deadlines	1. The user navigates to the deadlines section of the user interface. 2. User selects the specific module they would like to see the deadlines for. 3. The calendar will be updated to display only these deadlines.	The user can see when their deadlines are for their modules.
T1.3	Navigate Dashboard: browse deadlines, no deadlines	1. The user clicks on the module they want to view the deadlines for. 2. A message from the system informs the student that they have no deadlines.	The user is told they have no deadlines.
T2.1	View Tasks	1. User must have a module pre-selected, or the system will use the default (top) module in the dropdown options. 2. User navigates to the tasks section by pressing the appropriate button.	The user has their tasks for a certain module displayed.
T3.1	Plan Milestone	1. The user selects a task which they have previously created and clicks the "create milestone" button. 2. The user enters the title, start date and end date of the milestone. 3. The user adds any notes to the milestone. 4. The user saves the milestone.	A milestone is created for the user which will now appear on the calendar.

T4.1	View Milestones	<ol style="list-style-type: none"> 1. User navigates to the task they want to see the milestones for. 2. User clicks on the task. 3. All milestones for chosen task are displayed. 	The user can see all of the milestones for a certain task.
T5.1	Capture Progress	<ol style="list-style-type: none"> 1. The user selects the task (from the task list) that they want to capture the progress for. 2. The user then enters the amount of time they have spent on the task in each session. 3. The user then adds additional notes on what they have done in each session. 4. The time is added to the total completion for the primary task and all related tasks. 5. The total completion is updated. 	The user is able to update the progress they have made to completing a task.
T6.1	Edit module	<ol style="list-style-type: none"> 1. The admin user navigates to the module page. 2. The admin user selects the module they would like to edit and clicks the "edit module" button. 3. The user changes any details they wish. 4. The user confirms the changes. 	The admin user updates the details of a module.
T7.1	View deadlines	<ol style="list-style-type: none"> 1. User navigates to the deadlines page. 2. User either views all deadlines or can view separate module deadlines. 	The user views their deadlines.
T8.1	Edit Deadlines	<ol style="list-style-type: none"> 1. The user navigates to the module page and clicks on the module they'd like to edit the deadline for. 2. The user clicks the "edit deadlines" button. 3. The user selects the specific task to edit and changes the details. 4. The user saves the changes. 5. This change is applied to all students enrolled on the module. 	An admin user changes the deadlines of a module, which is changed for every other student enrolled on the module.
T9.1	View coursework (exams/assignments)	<ol style="list-style-type: none"> 1. The user navigates to the module they wish to see the coursework for. 2. The coursework is listed here along with other course details. 	A user can view the coursework set for a specific module.

3. Object Oriented Design [40% marks]

3.1 Class Diagram (Abstract)

(all identified classes and their relationships. Classify them into three group: Model, View, Controller)



Rationale:

The user class is the super class to both student user and admin user classes. The difference between the two being that admin user has the privileges to create and edit modules. The main fields will store information about the user's account to identify them, such as a name and ID. They will also store information regarding the modules the user is enrolled in or has created. Finally, they will store task data that the user has been assigned or has created themselves.

The profile class and its database will allow for users to personalise their experience by supplying module and year specific content for users. Will also contain everything related to account creation.

The module class will contain identification information, such as name and moduleID. The module class will also include a field to store all assignments, exams and pieces of coursework assigned to the module. Finally, the module class will also contain a list of all students enrolled on the module.

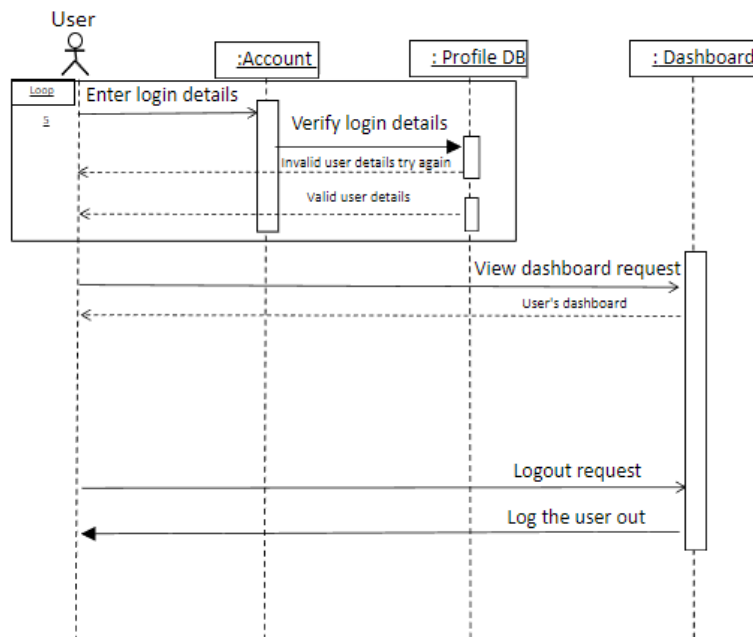
Our dashboard class will house the display the user receives when viewing the dashboard. This will contain calendars, events, and the subclasses of events.

The calendar class will display a user's upcoming events visually.

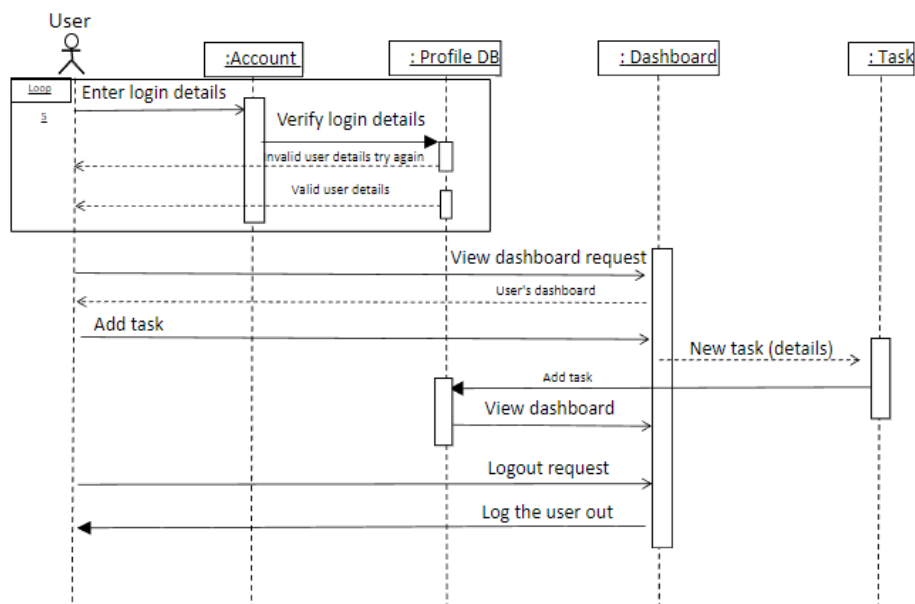
The event class will allow for event objects to be created. These objects will be defined by the following subclasses: milestone, task, deadline, assignments, exams, and coursework. The event class is a super class to milestone, task, and deadline.

3.2 Message Sequence Diagram (for three use cases of 2.2)

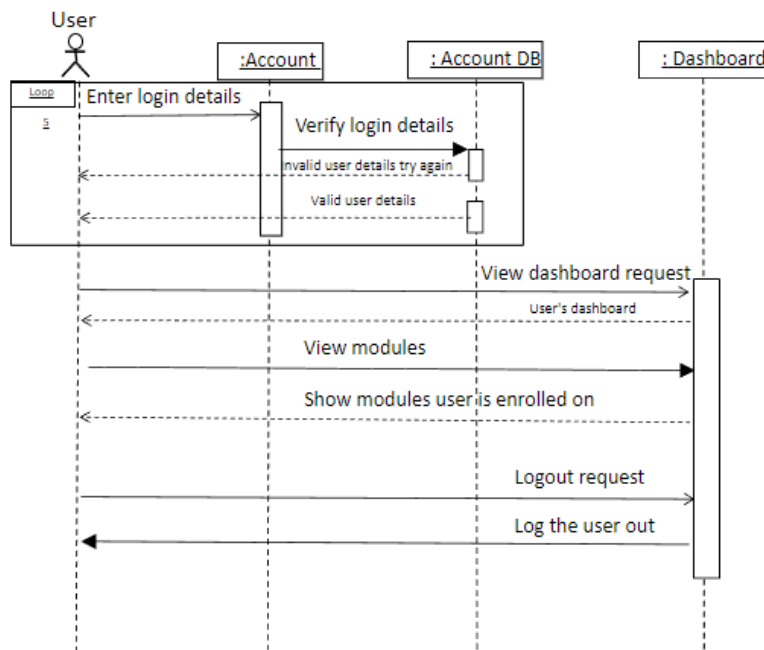
SD for 'Login User' use case



SD for 'Plan task' use case



SD for 'View modules' use case



3.3 Detailed Class Description

Three main classes of Student planner

We have excluded getters and setters to avoid bloat.

User
<u>userEmail</u> : String <u>userName</u> : String <u>password</u> : String <u>userId</u> : String <u>enrolledModules</u> : Module[] <u>userEvents</u> : Event[]
<u>resetPassword</u> () : String <u>login</u> (<u>userEmail</u> , <u>password</u>): Boolean <u>logout</u> () : Boolean <u>addTask</u> (<u>createEvent</u> , <u>eventName</u>): <u>viewTask</u> (Event): <u>submitFile</u> (file):

Module
<u>moduleId</u> : Integer <u>enrolledUsers</u> : User[] <u>tasks</u> : Event[] <u>coursework</u> : Event[] <u>exams</u> : Event[] <u>startDate</u> : Date <u>endDate</u> : Date <u>description</u> : String <u>additionalNotes</u> : String
<u>enrollUser</u> (userId): Boolean <u>removeUser</u> (userId): Boolean <u>removeExam</u> (): <u>removeCoursework</u> ():

Event
<u>eventName</u> : String <u>eventId</u> : integer <u>type</u> : String <u>location</u> : String <u>date</u> : Date <u>notification</u> : Boolean <u>notificationDate</u> : Date <u>additionalNotes</u> : String <u>progress</u> : Integer
<u>createEvent</u> (type): Boolean <u>deleteEvent</u> (eventId): Boolean <u>setNotify</u> (notification): Boolean

Event type will be overridden depending on the type of event required (milestone, task, deadline)

As a group, we were able to find a few foundational classes to build upon. We found four key classes: User, Module, Event, and Dashboard. Deciding that these four were needed as they were consistent in all our use cases. From our use cases, we identified more classes to be added. For our final three, we decided on User, Module and Event.

The User class is used for creating the subclasses StudentUser and AdminUser. The users are bidirectionally dependent on Profile. The Profile class will be used for storing accounts as objects, containing their login details as attributes. The Module will contain details of enrolled users and their personalised events, coursework, and exam dates. The Event class will have the following subclasses: milestones, tasks, and deadlines.

4. Implementation Plan [10% marks]

4.1 Task List and Team Member Allocation (*Weeks 6 – 11*)

Task list with duration and dependencies

Sprint 1

<u>Use case</u>	<u>Duration (Days)</u>	<u>Effort (Person Days)</u>	<u>Deadline</u>	<u>Start date</u>	<u>Dependencies</u>	<u>Allocation</u>
T0) Create account	2	5	14/3	12/3		DS, DH, MP
T1) Login	4	2	16/3	14/3	T0	MP
T2) Logout	4	2	18/3	16/3	T1	JM
T3) Create profile	5	8	20/3	15/3	T0	DS, DH
T4) View calendar	5	6	18/3	14/3		JM, FM
T5) Create module	4	3	21/3	18/3	T3	MP
T6) Delete module	3	2	23/3	21/3	T4	DS
T7) Plan task	2	5	23/3	21/3	T4	FM, JM, MP
T8) Plan milestone	3	3	24/3	21/3	T4, T3	DH
T9) Capture progress	4	8	27/3	23/3	T3, T7	DH, DS, FM

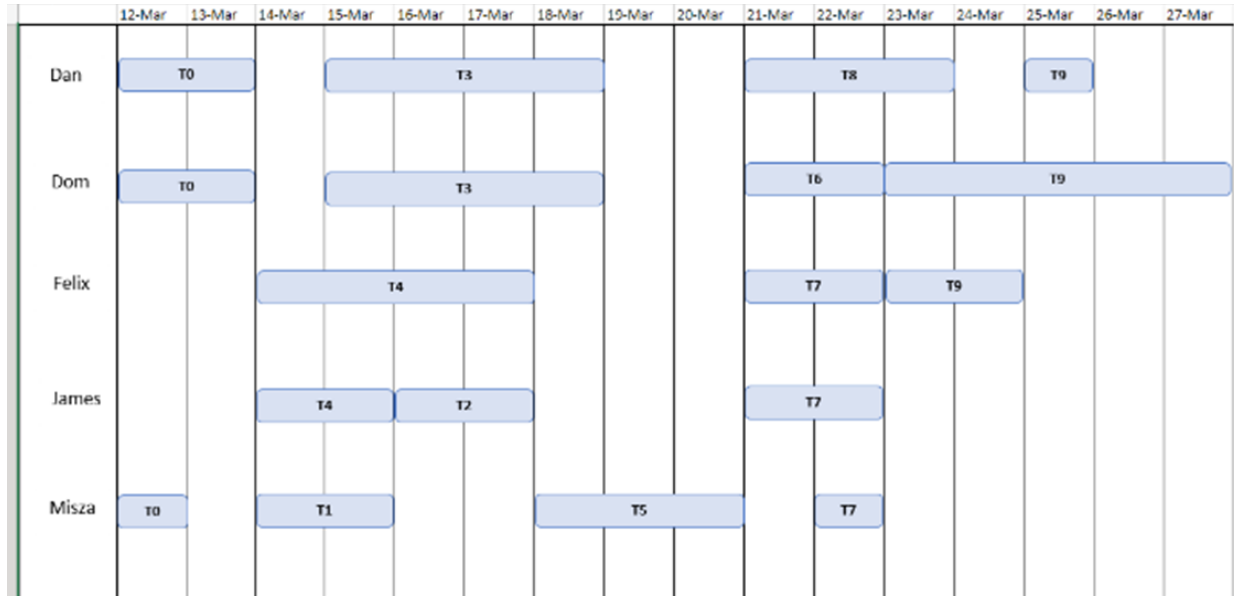
Sprint 2

<u>Use case</u>	<u>Duration (days)</u>	<u>Effort (Person days)</u>	<u>Deadline</u>	<u>Start date</u>	<u>Dependencies</u>	<u>Allocation</u>
T0) View modules	3	6	07/4	04/4		DH, DS
T1) Delete account	2	3	06/4	04/4		FM, JM
T2) Edit module	2	2	10/4	08/4	T0	MP
T3) Navigate dashboard	1	1	05/4	04/4		MP
T4) View deadlines	3	3	08/4	05/4	T0, T3	MP
T5) Submit file	4	4	09/4	05/4		JM
T6) Edit deadlines	2	2	08/4	06/4	T4	FM
T7) View coursework	2	2	11/4	09/4	T0, T3	JM
T8) View assignments	1	1	08/4	07/4	T0, T3	DH
T9) View exams	1	1	08/4	07/4	T0, T3	DS

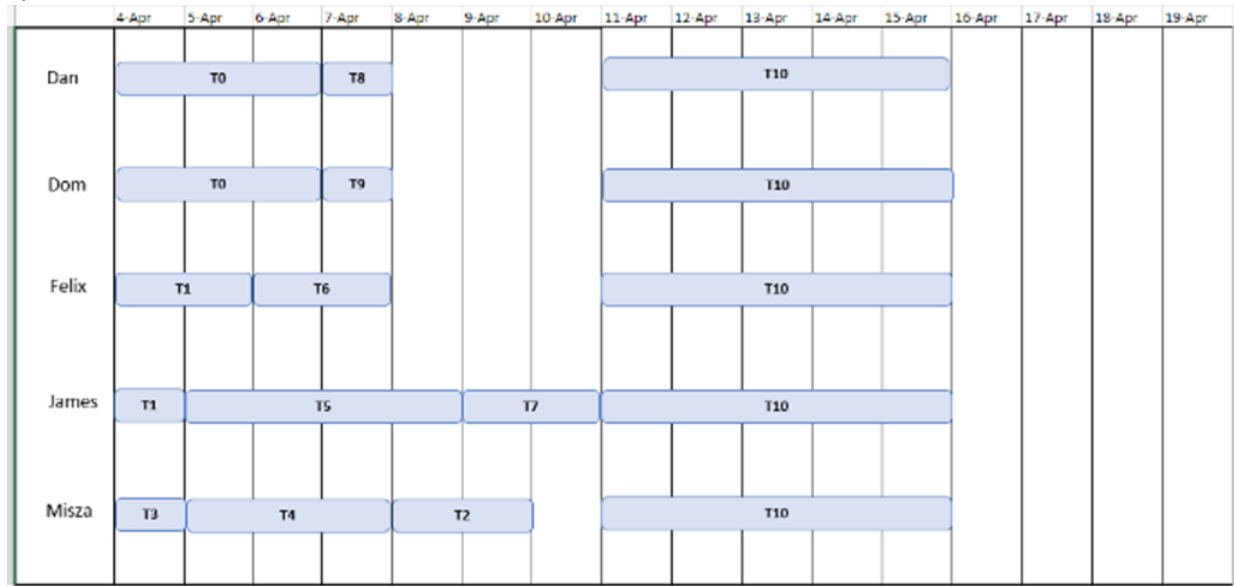
T10) Testing	5	25	16/4	11/4	T0,T1,T2,T3,T4, T5,T6,T7,T8,T9	DH,DS FM, JM MP
--------------	---	----	------	------	-----------------------------------	-----------------------

Staff Allocation:

Sprint 1



Sprint 2



4.2 Project plan for implementation, including milestones and deliverables

Milestones – sprint 1

#	Description	Dependencies	Start date	Duration (days)	End date	Assignee
1	Account creation, login, logout	T0, T1, T2	12/3	6	18/3	DS, DH, MP, JM
2	Working profile	T1, T3	15/3	5	20/3	DS, DH

3	Viewable calendar	T1, T2	14/3	4	18/3	JM, FM
4	Working module and task system	T1, T2, T5, T6, T7	18/3	5	23/3	FM, JM, MP, DS
5	Milestone and progress features	T1, T2, T4, T8, T9	21/3	6	27/3	All

Deliverables – sprint 1

Deliverable description	Prerequisites	Deadline
Working registration page		18/3
Being able to login and out of user's account	T1	18/3
Working profile	T1	20/3
Viewable calendar linked to profile with events and modules of that specific user	T1, T2	18/3
Milestones and progress features so users can track how they are doing with assignments	T1, T2, T5	27/3

Milestones – sprint 2

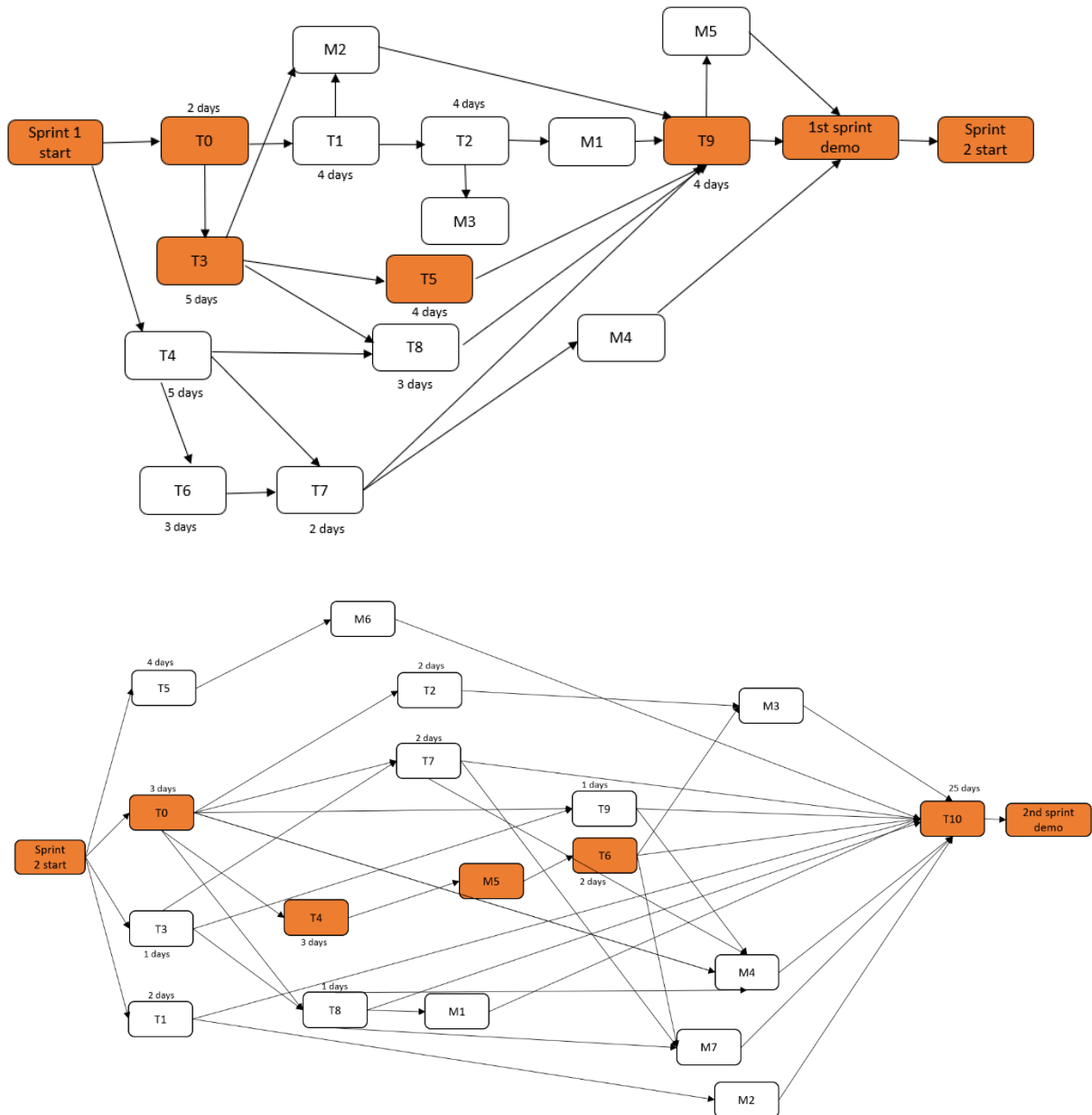
#	Description	Dependencies	Start date	Duration (days)	End date	Assignee
1	Viewable deadlines on assignments	T0, T8, T3	04/4	3	07/4	DH, DS
2	Option for users to delete their account	T1	04/4	2	06/4	FM, JM
3	Ability to edit modules	T2, T6	08/4	2	10/4	MP
4	Working UI that can be easily navigated		04/4	1	05/4	MP
5	Option for users to edit their deadlines	T4	05/4	3	08/4	MP
6	Working file submission point	T5	05/4	4	09/4	JM
7	Ability to view relevant information to students (Modules, coursework, assignments, and exams)	T6, T7, T8, T9	06/4	5	11/4	DH, DS, FM, JM

Deliverables – sprint 2

Deliverable description	Prerequisites	Deadline
Working assignments with deadlines	T3, T0	07/4
Ability to close accounts for users who no longer require the service		06/4
Fully fledged module system with ability to edit	T3, T2, T6	10/4
Working UI that is easy to navigate and learn for inexperienced users	T3	05/4
Working deadline systems	T0	08/4
File submission point with validation	T5	09/4
Ability to view modules	T3, T2, T6	08/4
Ability to view assignments	T3, T8	08/4
Ability to view exams	T3, T9	08/4

Ability to view coursework	T3, T7	11/4
Testing of system	T1 - T9	16/4

4.3 Activity network with critical path, and Gantt chart

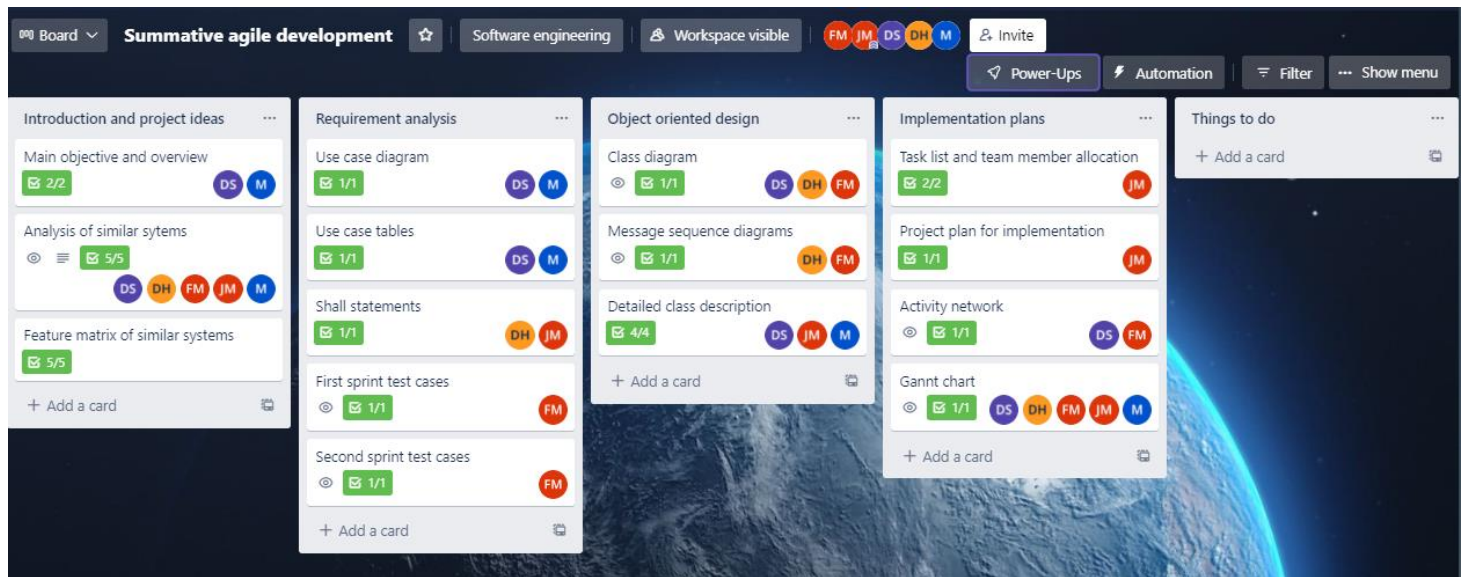


[illegible]

References *(Any external sources used in the document)*

- [1] [todo.microsoft.com](https://todo.microsoft.com/tasks/) | Microsoft To Do | [online] Available at: <https://todo.microsoft.com/tasks/>
[Accessed 7th February 2022]
- [2] Todoist | 2022 | Todoist: The to do list to organize work & life | [online] Available at: <https://todoist.com>
[Accessed 7th February 2022].
- [3] [mystudylife.com](https://www.mystudylife.com/) | mystudylife | Available at: <https://www.mystudylife.com/>
[Accessed 7th February 2022].
- [4] calendar.google.com | Google Calendar | [online] Available at: <https://calendar.google.com>
[Accessed 7th February 2022]
- [5] [outlook.office.com](https://outlook.office.com/calendar/view/month) | Microsoft Outlook | [online] Available at:
<https://outlook.office.com/calendar/view/month> [Accessed 7th February 2022]
- [6] [tutorialspoint.com/uml/uml_quick_guide.htm](https://www.tutorialspoint.com/uml/uml_quick_guide.htm) | UML Quick Guide | [online] Available at:
https://www.tutorialspoint.com/uml/uml_quick_guide.htm
[Accessed 21st February 2022]

APPENDIX



Capability number	Capability description
1.	Load module, coursework and deadline information from a defined file format
2.	Ability to define study tasks, details, milestones and deadlines
3.	Ability to record study activities that contribute towards completing study tasks and
4.	Visualise activities, dependencies, intermediate milestones and deadlines in a Gantt chart representation as well as a study progress dashboard that highlights upcoming deadlines, progress towards completing milestones and time spent for each module milestone