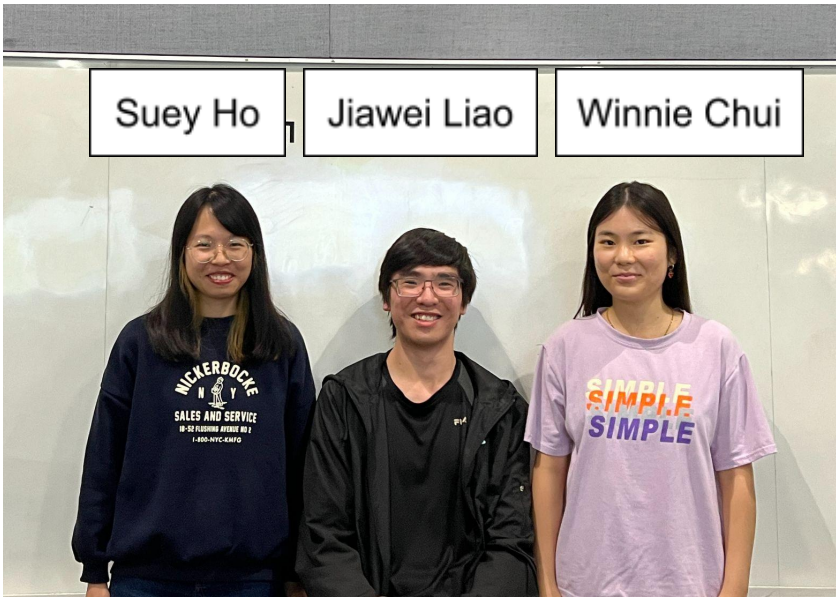
Team Information

Team Name & Team Photo

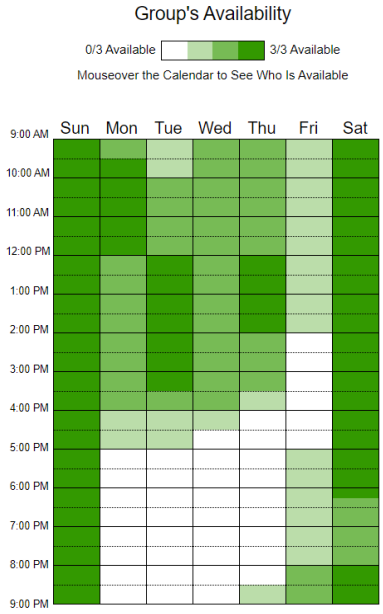
Pet Store



Team Membership

Name	Contact Details	Technical and Professional Strengths	Fun Fact
Suey Ho	0434 976 569 hlau0017@student.monash.edu	Programming languages that I know are C++, Python, Java, JavaScript/HTML/CSS. I have experience in building web pages, native windows games, and rogue-like games.	I cannot sleep whenever I drink tea.
Jiawei Liao	0468 367 352 jlia0058@student.monash.edu	I started programming/coding during university. The languages I have used so far are Python, JavaScript + HTML + CSS, Java, C++. The confidence I would have in creating 9MM in those languages would be in that order as well.	I have shot a gun before (small hunting rifle).
Winnie Chui	0482 013 799 cchu0061@student.monash.edu	I can code in Java, Python, JavaScript/TypeScript (+HTML & CSS). I have experience in using the Angular framework for building a web app. I enjoy working on frontend and UI/UX, but am also open to backend and any other kinds of work.	I have (almost) run a marathon before (33km).

Team Schedule



The table above shows the number of teammates available at a given time throughout a week. From this table, we were able to decide that we will meet at 12-2pm on Thursday on a weekly basis.

Over the weekend before the upcoming weekly team meeting, all team members will contribute to filling in the agenda for the upcoming meeting, by adding to it any action items to be done or topics to be discussed during the meeting.

After the agenda is filled in and before the team meeting on Thursday, each team member is expected to spend approximately 1-2 hours in their free time to prepare for the agenda of the meeting. For example, brainstorming ideas for creating the domain model and user stories.

During the meeting, all group tasks listed in the agenda will be completed by all team members together as a group. Any concerns or questions raised will also be discussed and addressed during the meeting.

At the end of the meeting, all tasks that require completing outside of the meeting and their deadlines will be identified, which will then be assigned to different team members evenly. All team members are expected to complete their assigned tasks by the agreed deadlines.

Although some tasks are assigned to be completed individually, the overall workload will be managed by the team collectively, where the team member to whom a task was allocated will provide updates when they finish the task or if help is needed. Each member will also check up on one another regularly, for example, 3 days before the agreed deadline, to make sure that all tasks are on track to being finished by the set deadline.

Technology Stack & Justification

Option 1 - Python

Pros

- Has a large range of libraries and frameworks, meaning that it is very flexible as there will most likely be a library that suits our needs
    - Libraries such as Tkinter and Pygame suit our needs very well
- Popular programming language, where every member of the team has experience using it.
- Supports object oriented programming
- Very easy to use, with very readable syntax and automatic memory allocation

Cons

- Libraries and frameworks would still need to be installed before being usable
- Slower to run than languages such as C++ or Java

Option 2 - C++

Pros

- C++ is a universal game programming language that is widely used for many platforms. The same C++ code can be compiled for different platforms such as Android, Mac OS, OS, and Windows.
- C++ has a large number of libraries available for game development, graphics rendering, and user input handling.
- C++ programs run faster than interpreted languages such as Python.

Cons

- C++ is a relatively complex language that can take time to learn if team members do not have experience with C++. This can slow the development.
- Need to be careful about memory management as manual memory allocation and deallocation is required.

Option 3 - TypeScript

Pros

- Very similar to JavaScript, which all team members have experience with
- It is a strongly typed language
    - It has compile-time error checking, preventing runtime errors
    - No runtime penalties for determining types
    - It gives more structure to the code, making it easier to debug
- Can be used with HTML and CSS, making it simple to build a web application
- Can be used in frameworks, such as Angular and React, which have access to many UI libraries
- Compatible with different platforms and browsers

Cons

- It is a strongly typed language, meaning code flexibility is reduced
    - It is more challenging to define a collection of heterogeneous objects
- It has to be transpiled to JavaScript before running the code which might require additional time

Final Decision

After discussing the pros and cons of each available option, it was decided that Option 3 - TypeScript will be used for building the 9MM game. The main reason behind choosing TypeScript was the ease of using it. Since all team members already have experience with using JavaScript with HTML and CSS, learning TypeScript will not be too difficult as it is very similar to JavaScript, with the main difference of it being typed. Moreover, TypeScript can be used with HTML and CSS to build a simple web application, without the need of using external libraries, which makes the development quicker and easier.

Option 1 - Python was a close second choice. While it is a simple language to use, the need to use external libraries, such as Tkinter and Pygame, made it so that another language had to be basically learnt. Therefore, this option was discarded as the benefits did not outweigh the drawbacks.

Option 2 - C++ was decided unsuitable for our team since not all members have experience in C++, which could introduce a steep learning curve if it were used as it is a complex language. This could lead to a slower development as the team needs to spend more time

learning the language before actually able to start building the 9MM game. Therefore, it may require more time and knowledge to develop the 9MM game compared to the other two options.