Team 5 Sprint Document 1

1) Summary data	
Team number	5
Sprint technical lead(s)	Eric Shi
Sprint start date	03/02/2024
Sprint estimated end date	15/02/24

2) Individual key contributions	
Team member	**Key contribution(s)**
Eric Shi	Rule and action diagram, GitHub setup
Lin Hein	Began to design UI
Owen Chen	Development of basic game mechanics
Duncan Law	Development of basic game mechanics
Stuart Baker	Resource finding, board design, rules

3) User stories / task cards

For task 1, this is purely just to help with development as having examples of similar games is useful for an initial idea of what we could create. Combining this with the user requirements from Mr Raffles provides our team with an abundance of resources and examples.

Task 2 will provide us with a document showing all the rules in a format that is much easier to follow than the extensive text showing the rules provided in the brief. This will make programming the rules simpler.

The same is true for task 3, breaking down the actions allows for easier programming as there is less need to analysis the full brief during development. Instead of using the full brief during development, we will have a document outlining the different actions in each scenario. The full brief will still be used and referenced when needed but would be too time-consuming to repeatedly read when programming.

Task 4 and 5 can be combined and viewed more as the start and end product of each other. Each class will include their respective components found in the diagrams produced in task 3 and 4. For instance, the player class will include code for the player starting the game with $1500, and the property class will have a charge_rent function.

Task 6 is like task 2 and 3 but tailored to the UI development. A visual design before writing the UI code is useful as it will provide us with an image of what we need to create. Task 1 will help to design the board as we'll be able to use the resources, along with the board tile description provided by Mr Raffles in the PropertyTycoonBoardData.xlsx document.

Task 7 and 8 can also be combined, being the starting and finishing stages of each other. This will be where we implement the board design and layout in a Python file, so we have it ready to be combined with the rest of our classes to bring the game to life.

Task 9, will provide us with a main class that can execute the game, calling the different classes and making it playable for a user.

Task 1: Find resources to help with the development, such as examples of the game – both physical and online

Task 2: Interpret and breakdown rules into an easy-to-follow chart/diagram

Task 3: Interpret and breakdown game actions and events into an easy-to-follow chart/diagram

Task 4: Begin to program rules and actions, using Python and Pygame

Task 5: Produce player, property, game and board classes

Task 6: Design a board that incorporates all of Mr Raffles requirements, and can be used as a template when creating the UI

Task 7: Begin to program the UI

Task 8: Produce UI class

Task 9: Develop a main class/file to run the software

4) Requirements analysis

For task 1,2,3 and 6, the requirement analysis would just be analysing the brief and breaking it down into its respective components – highlighting important information and making notes on all 'must dos' and 'optional dos'. There aren't really any certain guidelines we must adhere to for this as it is an interpretation.

Task 4 and 5 has certain requirements outlined in the documents created in task 2 and 3. Player class:
- Player must start with $1500, on the GO position (position 0) and with no properties owned

Property class:
- Must have a name, price, rent when owned and no owner at the start

Game class:
- Initialises board makeup, holds player board position information and has states 'roll', 'buy' and 'end'
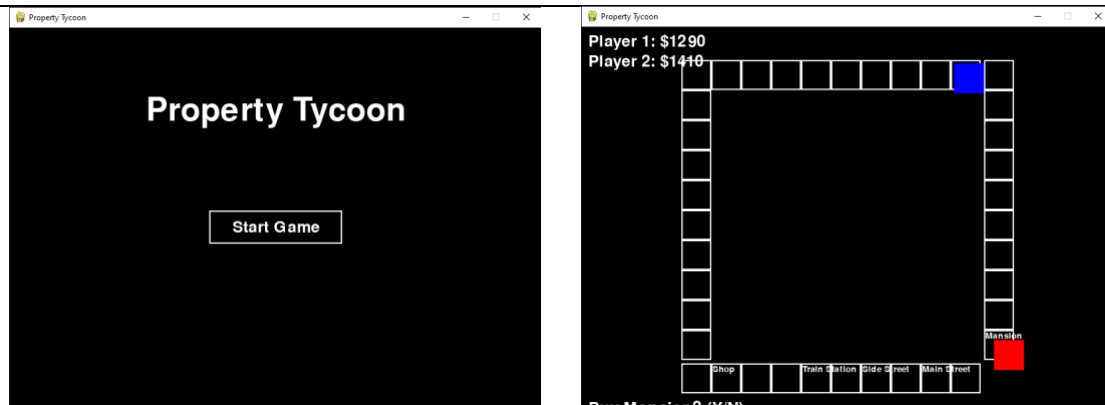- Handles jail logic, roll logic and purchasing logic

Board class:
- Holds the first bits of board data, such as tile properties etc.
- Creates board positions when initialised

Task 7 and 8 also has specific requirements needed to be implemented. UI class:
- Must be colourful and reflect the spirit and character of the original board game
- Handle clicks

5) Design

Above shows the first running prototypes of the games start page, and board, with the red and blue squares showing player location. Below:

Top left is one of the resources we gathered, being a photo of a physical monopoly board game. Bottom left is the rules flow diagram we produced, with the action flow diagram being top right.
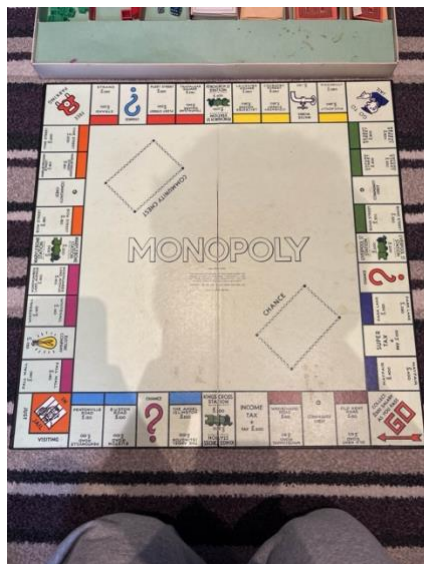
Bottom right is the board design we created to provide a visual idea of what we are aiming to produce.
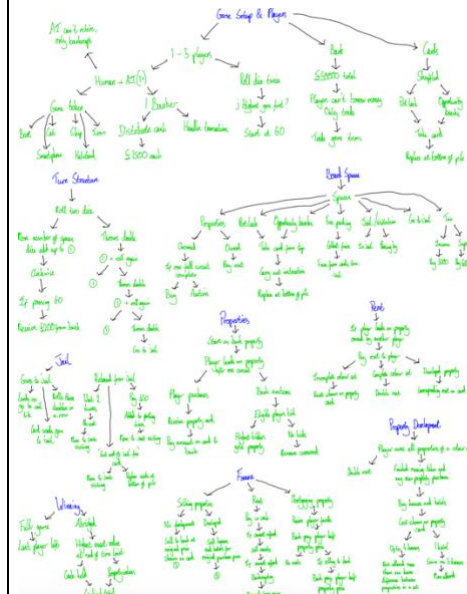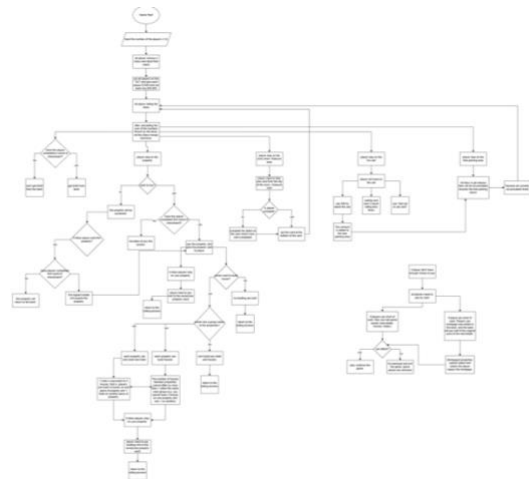
6) Test plan and evidence of testing

No formal testing was undertaken at this point in our development. We tested that the code worked to meet our sprint targets, but only really through no errors being thrown when the program was executed, and visually at what the program produced.

We will be undertaking more testing as the project becomes more complex and further developed.

7) Summary of sprint

This sprint cycle overall went very well. We achieved our sprint objectives to varying degrees of success, worked well together as a newly formed team, and communicated effectively throughout the cycle. Each member was actively involved in the process, helping to ensure the objectives were achieved.

Tasks 1 through 9 were all completed to a good standard, providing us with a strong foundation and a good start for our project. The non-coding tasks were found to be very helpful for when the code was being written as it saved a lot of time for the coders as they didn't need to constantly be reading through and checking the full brief.

We found that most classes incorporated aspects from other classes, leading to an interconnection between them all. The classes also worked well, particularly the UI class which went above our objectives and created a start page for the game, which improves the overall user experience.

One area we underperformed in was in the implementation of the board. The board lacked the colour that Mr Raffles was asking for in the brief, which affects the fun aspect as boring colours are used. This is an area that we will have to focus more on in the future, but it is an easy fix and will be addressed in the next sprint cycle.

The working prototype is a very good stepping stone into the future development of this project, providing us with the basic game mechanics being implemented, and the start of the rules being programmed too. Certain aspects of the game are not yet fully functional, such as the character movement after the dice roll, the actual dice roll, property purchasing and auctioning, and the final board design. However, the players ability to move/jump around the board and their position to be seen is a very rewarding start to the project.

Overall, a promising start and a successful sprint. No major issues occurred, and communication was effective as a team with any questions being answered and multiple meetings happening. We haven't shown Mr Raffles our current working prototype, but we will aim to show him the next prototype where feedback would be more useful because the development would be further, but not too far, along.