# Chapter I - Introduction

For the culminating project in our Object-Oriented Programming course, we have developed a comprehensive application that leverages fundamental and intermediate principles of object-oriented programming. This project not only demonstrates a solid understanding of the basic tenets of OOP but also incorporates advanced concepts to enhance its overall sophistication. Java is the only programming language we have used in this project, a representative language in OOP. Besides that, we have used Swing to create GUI for this project.

This report will provide a detailed exposition of our project, delineating its various facets. The structural framework of our report is delineated as follows:

Chapter I: Introduction.

Chapter II: Rules and gameplay.

Chapter III: The details of Game technique.

Chapter IV: UML class diagram.

Chapter V: Evaluation.

# Chapter II - Rules and gameplay

* 1. Gameplay

Monopoly is a classic board game renowned for its strategic gameplay and economic themes. The game involves the following rules and gameplay elements:

1. \*\*Objective:\*\*

- The primary goal is to bankrupt opponents by acquiring their properties and accumulating wealth.

2. \*\*Game Setup:\*\*

- Players choose a color and receive a set amount of money.

- Properties are distributed evenly, and each player is given the opportunity to purchase or auction properties in their initial rounds.

3. \*\*Board Layout:\*\*

- The board consists of a square layout with 40 spaces, including 35 properties (31 Cities, 4 Beaches), 3 Chance spaces, a Go space, a Tax space, and Jail.

4. \*\*Turn Sequence:\*\*

- Players take turns in a clockwise direction.

- A turn involves rolling two six-sided dice and moving the corresponding number of spaces.

5. \*\*Property Ownership:\*\*

- When a player lands on an unowned property, they have the option to purchase it.

- If a player lands on a property owned by another player, they must pay rent, the amount of which is determined by factors such as property value and development.

6. \*\*Building Houses and Hotels:\*\*

- Players can enhance their properties by building houses and hotels.

- Building costs increase with each subsequent house and are determined by the property group.

7. \*\*Bankruptcy: \*\*

- A player is declared bankrupt if they cannot pay debts, including rent or fines.

- The remaining assets of a bankrupt player are usually distributed among the remaining players.

8. \*\*Jail:\*\*

- Players may land in Jail by landing on the "Go to Jail" space or get Go To Jail Chance Card

- Players can attempt to roll doubles to get out or after three rounds in Jail, player will pay fee to get out or using Get Out of Jail chance card.

9. \*\*Chance and Community Chest Cards: \*\*

- Players draw cards from these decks, which can provide various opportunities or challenges, such as advancing to specific spaces or paying fines.

10. \*\*Event\*\*

-When player step on event square, they are allowed to set an event at their own city. When a city has been set event (here is World Cup event), the visit cost of that city will be ten times.

- On the board only one city will be chosen to hold an event. For example, if player 1 has already held an event, and then player 2 steps on event square and player 2 wants to hold an event, the event city of player 1 will be removed and the player 2’s city can hold an event.

11. \*\*Plane\*\*

- When player step on plane (World Tour) square, player can travel to one of these squares:

+ Their cities

+ Unowned city

-When players travel to these squares, they can buy or upgrade these cities if they have enough cash.

Fee to use plane: 50k$

12. \*\*Winning the Game: \*\*

- The last remaining solvent player, who has not gone bankrupt, is declared the winner.

- Player owns 4 beaches.

Monopoly's enduring popularity lies in its blend of strategy, luck, and negotiation, making it a timeless and engaging board game for players of all ages.

Ingame GUI

A screenshot of a game

Description automatically generated

# Chapter III – The detail of Game technique

3.Main game tree

* 1. Introduction about function of used class:
* Game Manager:
* Class Monopoly:
  + To create squares on the board
  + To control game structure
  + To initialize player, game board
  + Connecting parts
  + Print result
* Class Squares:
  + Create basic attributes, method of each square.
  + Create simple GUI for each square.
* Class Property Square, City, Beach:
  + Create basic attributes and methods for each property.
* Class Dice:
  + Create a dice and calculate get dice value.
* Class Plane:
  + Control and calculate player location after using plane.
* Class Tax:
  + Calculate tax during the time elapsed.
* Class Timer:
  + Create a timer and calculate the time left.
* Class event:
  + Create events and handle the player’s city after event.
* Class Card:
  + Create community and chance card.
* Class Jail:
  + Hold player in Jail and allow player to get out of Jail
* Class StartMenuGUI:
  + Build a start menu and get players’ information.
* Class GUIV2:
  + Create game GUI.
* Player:
* Create basic player attributes.
* Handle interaction of player between player and square.
* Handle sell, buy, upgrade, exchange properties
  1. Flow of main game

First, the Monopoly class will call out a start menu for users to display game Welcome, authors.

Second, when user press on Start Game button, the game will require users to input number of players, player’s name and color for their representative label in the game.

Next, the main game GUI will display on screen and user can start to play by pressing roll dice button. For each round, the game frame now will listen to the mouse click of player on the roll dice button and randomly generated dice value. The dice value will be assigned for the step of player on the game board. All information or interaction will be stored and displayed on the text area on the left side of the screen. In each round, the player will be allowed to interact with square and the game will continue to run until one of winning conditions is reached.

Finally, the game will print out the winner and total money that they earned during the game.

# Chapter IV - UML Class Diagram

A computer screen shot of text

Description automatically generated

# Chapter V – Evaluation

* This game is designed and built for the simplest and flexible way that can improve freely
* Can improve a lot of features in future ( more chance card, more event, change the location or properties’ cost in the game)
* Can improve better GUI