

API Design Document

for

Monopoly

Prepared by:

MAN Furui (19081789d)
LIU Sicheng (19079181d)
WANG Meng (19078543d)
XING Shiji (19079008d)

The Hong Kong Polytechnic University
Software Engineering
Course Project
Prof. Yu PEI

Hong Kong, 22 October 2021

Contents

1	Introduction	2
1.1	Purpose	2
1.2	References	2
1.3	Monopoly Class Diagram	2
2	Main	4
3	Controller	5
4	Model	6
4.1	Board	7
4.2	Squares	7
4.3	Player	7
5	View	8
6	Appendices	9
6.1	API Documentation	9

Chapter 1

Introduction

1.1 Purpose

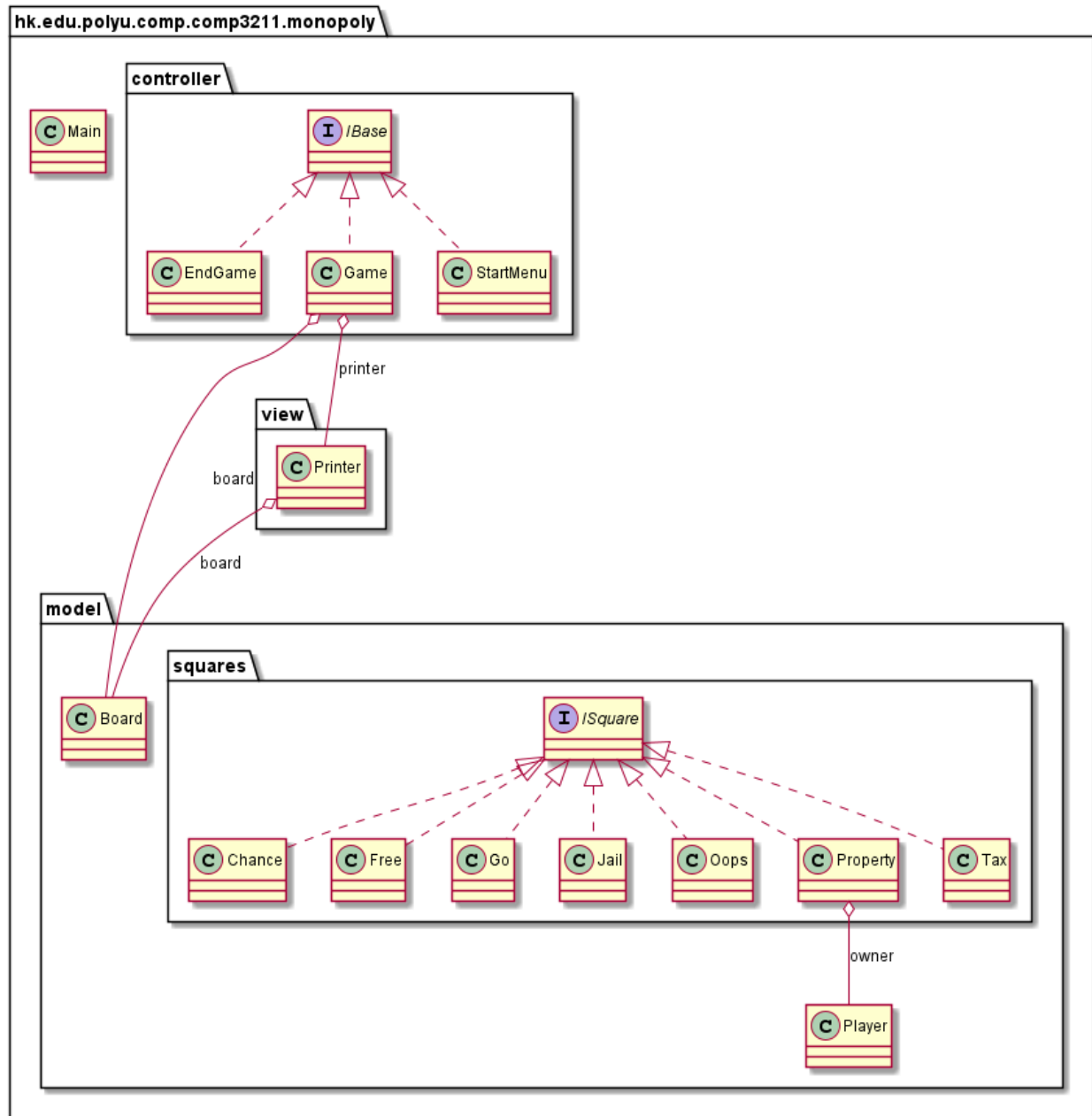
This document mainly illustrates important design decisions of the architectures of the product. The product is implemented following the MVC model. However, since the product is based on command line, basic prompts are implemented in controller model, while the printing of the game board is handled in a specific class in view model.

1.2 References

- COMP3211 Software Engineering Course Project Description

1.3 Monopoly Class Diagram

MONOPOLY's Class Diagram



COMP3211 Software Engineering Course Project: Monopoly, 2021 Fall
by Group 22: MAN Furui, LIU Sicheng, WANG Meng, XING Shiji

Figure 1.1: Monopoly Class Diagram

Chapter 2

Main

The entry point of the application.

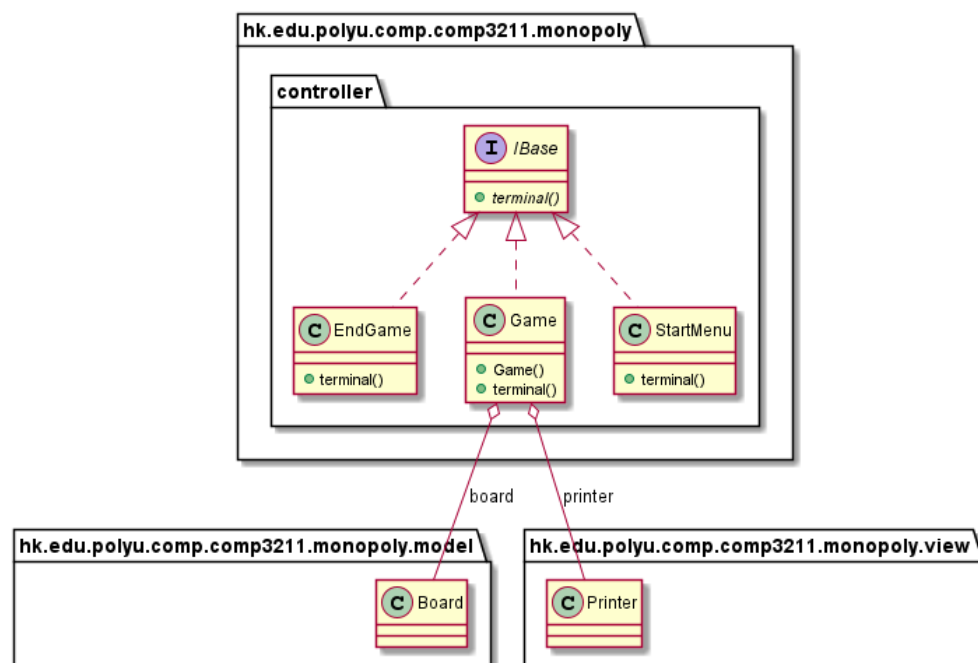
A globally-shared scanner is defined in this class for other classes to use. The main control flow of `Main()` is calling the `terminal()` methods of different controller interface in a infinite while loop. Different controller handles different scenarios in the application, such as start menu, in game and end game. At the beginning of the application, `Main()` will create a `StartMenu` controller interface and pass the control to it.

Detailed description of controller class will be noted in the controller section.

Chapter 3

Controller

CONTROLLER's Class Diagram



COMP3211 Software Engineering Course Project: Monopoly, 2021 Fall
by Group 22: MAN Furui, LIU Sicheng, WANG Meng, XING Shiji

Figure 3.1: Monopoly Controller Class Diagram

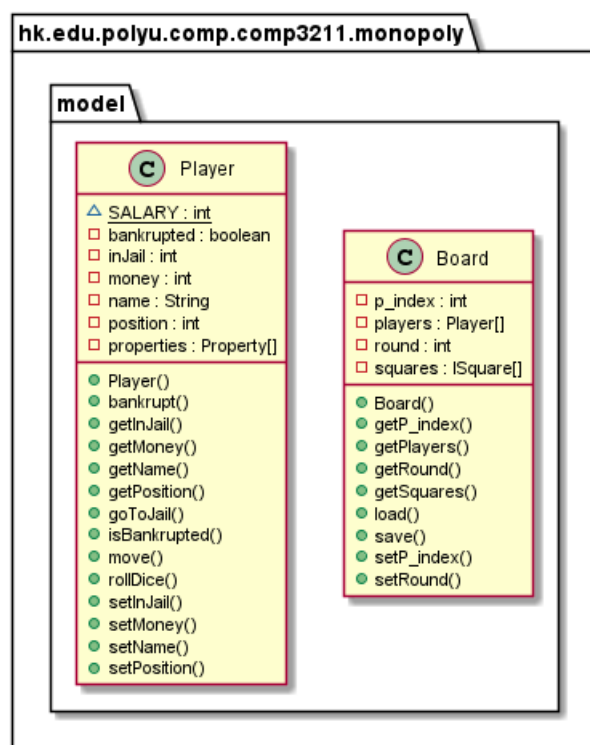
All classes in this model should implement the interface `IBase`, which has a method `void terminal()`. This method is called by `Main()` in an infinite loop, and is designed to process one or more user inputs at a time. Several classes are defined implementing this interface, and they are used in different interfaces. Specifically, `StartMenu.terminal()` is called when the game is in start menu, and `Game.terminal()` is called when it is in the middle of a game. Same applies to `EndGame.terminal()`, when a game ends. The `Main` class will have a reference to the current interface, and this can be changed by controller classes in order to perform interface switching.

Chapter 4

Model

This section defines the model of the application, in which there are several significant components, namely the board, the square, and the player, implemented respectively by classes under the same reference. Details of each component class are covered in its corresponding section.

MODEL's Class Diagram



COMP3211 Software Engineering Course Project: Monopoly, 2021 Fall
by Group 22: MAN Furui, LIU Sicheng, WANG Meng, XING Shiji

Figure 4.1: Monopoly Model Class Diagram

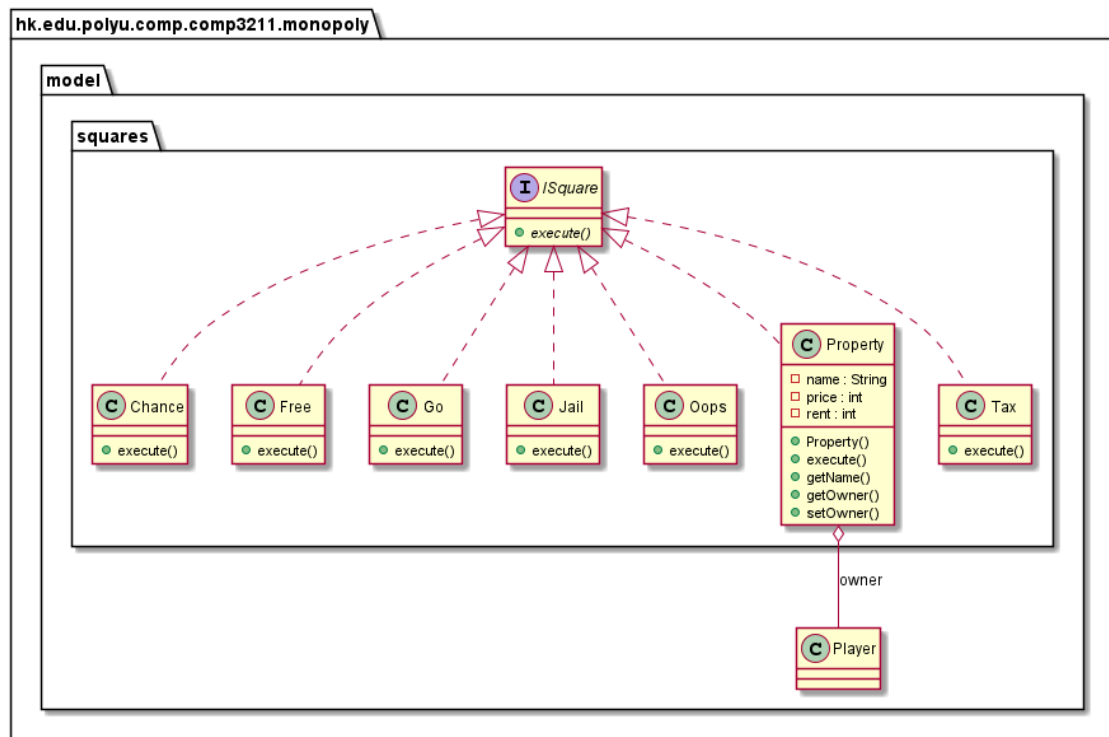
4.1 Board

This class stores the information of the game. It keeps an array of players and an array of squares. The layout of the board (along with its functionalities) is initialized in the constructor and is stored in the array of squares. Furthermore, it also records the round of the game and current actioning player. The board also supports saving and loading to/from local file.

4.2 Squares

The basic unit of map. All subclasses should implement the interface `ISquare`, which has one method `void execute(Player player)`. The method is called to perform specific operations on the current user stepping on the square. The following classes implements this interface: `Chance`, `Free`, `Go`, `Jail`, `Oops`, `Property` and `Tax`. The name of classes correspond to the type of squares in Appendix B of the group project document. Note that class `Oops` refers to the square "Go to Jail".

SQUARES's Class Diagram



COMP3211 Software Engineering Course Project: Monopoly, 2021 Fall
by Group 22: MAN Furui, LIU Sicheng, WANG Meng, XING Shiji

Figure 4.2: Monopoly Squares Class Diagram

4.3 Player

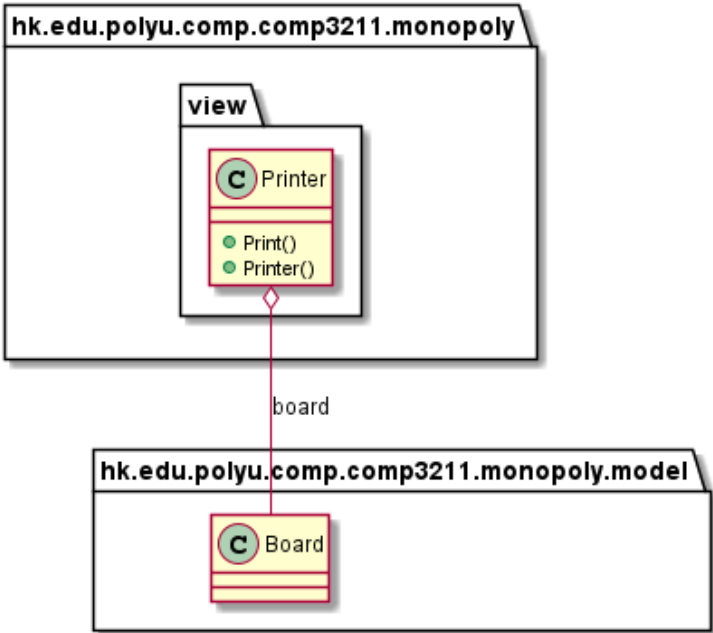
The class storing information about players. It should store players' name, money, current position, jail state, etc. In addition, it should supports auxiliary methods that adds money, moves forward in the map, teleport to Jail and rolls the dice. Moreover, it should have a `Property[]` array to store the properties owned by the player.

Chapter 5

View

The model only contains one class `printer`, which is used to print out the game board during rounds.

VIEW's Class Diagram



COMP3211 Software Engineering Course Project: Monopoly, 2021 Fall
by Group 22: MAN Furui, LIU Sicheng, WANG Meng, XING Shiji

Figure 5.1: Monopoly View Class Diagram

Chapter 6

Appendices

6.1 API Documentation