

Monopoly Board Game Requirement Analysis

CSE3063 Object Oriented Software Design
Marmara University

Team

Göksel TOKUR 150116049
Zahide Gür TAŞTAN 150119827
Merve AYER 150119828
Ertuğrul SAĞDIÇ 150116061



MARMARA
ÜNİVERSİTESİ

1.Introduction

Vision: The objective of the project is the implementation of the monopoly board game Java based object-oriented software design approach. The project is designed and developed by a team of 4 members. Team members are Göksel Tokur, Zahide Taştan, Merve Ayer. Ertuğrul Sağdıç.

Scope: Monopoly simulation

Special Requirements

- Java
- Visual Studio Code
- IntelliJ
- Git
- GitLab
- SSH Secure File Client

Stakeholders:

Murat Can GANİZ (Customer)

Serap KORKMAZ (Customer)

Göksel TOKUR

Zahide Gür TAŞTAN

Merve AYER

Ertuğrul SAĞDIÇ

2.Requirement Specifications

2.1 Must Have Requirements

Implement a text based console of the Monopoly Game Simulation for this requirement a consistent implementation of the Monopoly that can be played by two to eight players.

- The observer input and game output will be via terminal.
- Provide some observer commands that control the simulation.

Simulated computer players

- Simulated players must obey by the rules of the game.
- Computer players are going to challenge with themselves.

2.2 Could Have Requirements

Improved artificial intelligence of the simulated player.

2.3 Non-Functional Requirements

Reliability

Improvable software

Efficiency in terms of response time and memory

Maintainability

2.4 Game Specifications (or Rules)

This is an insight into the way the group aims to achieve the implementation of the text based console version of the game. This is not a comprehensive list; it will be updated once the team has prepared the detailed design, class diagram and the sequence diagrams for the game.

2.4.1 Start-Up:

The Monopoly has been played in a series of rounds. During a each cycle each player takes one turn.

- The Bank gives to each player 10000. The remaining Money will be kept in the Bank until required.
- Each player rolls two dice. The player who rolled the biggest integer from faces of the dice, will start at first. The rest will be started regarding to their rolls.
- In each turn, players need to roll two dice, and players move their own pieces.
 - Every turn, the name of the player and current location of the player's piece will be displayed.
- The piece belongs to player, will stay on the board and it will continue from that square.

2.4.2 Landing on specific squares

There are several different squares, such as start square, tax square. If a player lands on start square or passes through that square, player takes wage which has been set at the beginning of the game by the observer. If a player lands on tax square, player has to pay the amount which has been set at the beginning of game.

3. Glossary

- **Board:** The Monopoly game-board consists of 40 squares
- **Square:** Containing twenty-eight properties, four railroads, three Chance spaces, a Tax space, and the four corner squares: GO, (In) Jail/Just Visiting, Free Parking, and Go to Jail etc
- **Start Square:** The Square that each player locates at the starting of the game, players collect moneys when they passed over it.
- **Tax Square:** The Square that if a player landed on, they have to pay tax to the bank.
- **Dice:** Player rolls the dice and his/her pawn is moved according to the sum of dices' faces.
- **Player:** A bot that is behave like human. This game can be played by 2 to 8 players.
- **Piece:** Piece is a token of a player. Each player has a unique piece.
- **Money:** Monopoly money is a type of play money used in the board game.

4. Use Case Model

Actors of the game are simulated players. In other words, the software is run as a computer simulation watched by an observer. Game simulation will be controlled by an observer. Observer decide how many players will play and then start a game. Simulated players are going to start playing the game.

1. The user determines the number of players who will play the game via terminal input.
Two to eight players can play the game.
2. Observer starts a game.
3. The system shows turn and cycle counter.
4. The system shows name, location information, money of the player which takes a turn.

5. Player toss the dice, after tossing dices the system will output face values and sum of faces.
6. The player will go to the new location.
7. The system shows the name and the type of the location, name of the owner if exists, type and number of properties in the square, the amount of rent or fee if any and current balance of the user.
8. At the end of each cycle the system shows the list of all simulated users' information. And properties ordered by their balance.