REQUIREMENT ANALYSIS DOCUMENT

Requirement Specification

Vision

We want to observe a monopoly game which is played by the simulated players. The game will be run until one player is left in the game. While players move on the board, they can buy some property squares and may take actions according to the features of squares. While the game is running, the status of each players actions will be printed to the system user.

Problem Statement

We want to observe many different scenarios, but a real monopoly game takes too much time. So, we will simulate it and be able to examine different and large number of scenarios.

Scope

The simulated monopoly game will provide the following:

- The ability for a game observer to give names to the simulated players.
- The ability for the observer to see the status and actions of the simulated players.
- The ability for the observer to see how the simulated players interact with the game.
- The ability for the observer to see the winner at the end of the game.

System constraints:

- Will run automatically with the simulated players
- Will run without a user interface
- Will print the outputs to the command line

Stakeholders:

- Customer
- Analyst/Programmer

Glossary of Terms:

User interface: is the space where interactions between humans and machines occur.
Simulated player: The player plays the game by himself/herself and makes decisions randomly.
Board: A board used for playing the monopoly game.
Token: is an object to represent the player in the board.

Square: The board consists of 40 squares which have different properties.
Go square: It is the starting square and the player gets money whenever he/she passes through it.
Jail square: It is the jail; the player loses money whenever he/she passes through it.
Chance Card and Community Chest Card: They have different features, are used mostly to move players, make players gain or pay money during the game.

Use Cases:

Observer runs the simulation.
Actors: observer and monopoly simulation

1. Simulation asks the observer to give number and names of players.
2. Observer gives the inputs to the simulation.
3. Simulation creates a game.
4. Game creates board, players and dice. Players create tokens.
5. Board creates squares.
6. After these object creations, simulation starts the game.
7. Player rolls the dice and according to face values moves to the next square.
8. If the player passes the go square, he/she collects money.
9. If the player comes to jail square, he/she stays here for 3 rounds. But he/she can get out of jail if decides to pay money or he/she has a jail free card.
10. If the player comes to the property square, he/she can buy that square or if another player has already bought the lot square, then current player pays rent to the owner.
11. If the player comes to the chance or community chest square, he/she draws a card. According the card, player does the operation.
12. If the player comes to the tax square, he/she loses money.
13. End of each turn, all players are checked if a player goes bankruptcy. If a player is gone bankruptcy, the game sends the player off.
14. Turn passes to the next player.
15. Return to step 7.
16. Until one player is left in the game.