Bilkent University

Department of Computer Engineering

CS 319 Course Project

Group: 2C

Analysis Report

- Furkan Kazım AKKURT
- Abdullah Can ALPAY
- Murat ANGIN
- Ümit Yiğit BAŞARAN
- Muhammed Emre YILDIZ

Instructor: Eray TÜZÜN

Teaching Assistant(s): Barış Ardıç, Elgun Jabrayilzade, Emre Sülün

Analysis Report

Project short-name: Monopoly Project

Introduction

A monopoly is a strategy-based board game presented by Hasbro. The purpose of the game is to become the player with the most money by buying properties, collecting rents, and selling the properties while trying other players to declare bankruptcy. The game is played by 2-8 people. Each player starts with $1500, and they roll dice at their turns. Based on the value of the dice, players move on the board and can allocate spaces by buying them. There are also different kinds of squares such as jail, chance, and community chest. When a player visits to jail, they should choose one of the two options; either getting out from the jail using a jail card which can be retrieved from the chance cards, or skipping their turns for 2 consecutive turns. When a player passes through the starting point, they receive $200. When a player owns all the properties in the same color, they are allowed to buy houses from the bank, in which the amount of rent other players have to pay increases. Hotels are allowed to be bought when a player has 4 houses on each property in the same color group. The game-winner is determined after all the players declare bankruptcy except one player, in which the winner is counted as the player that did not declare bankruptcy. There are different versions of the Monopoly game, but in this report, the standard version is going to be explained.

Current System

In the current system of the original game which is published by Hasbro, there are a board, 2 dice, tokens, 32 houses, 12 hostels, 16 chance cards, 16 community chest cards, and monopoly money worth 20.580 units.

The Board

The monopoly board is where the gameplay takes place. (Figure 1)

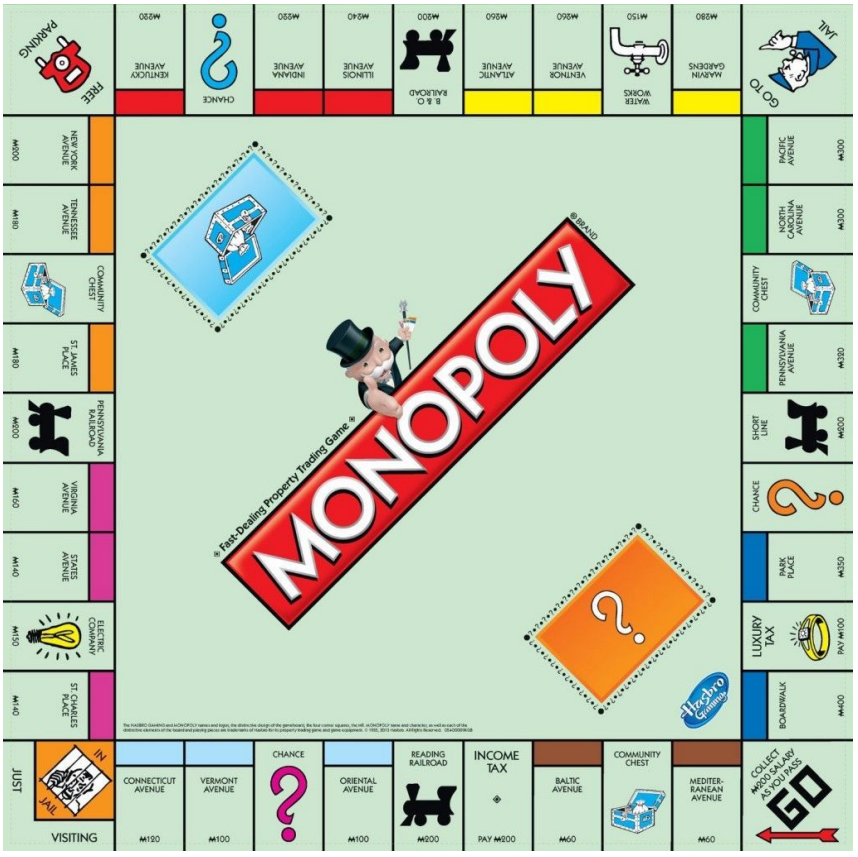

Figure 1: The Official Monopoly Board

Board has some subcomponents such as starting point, properties, chance and chest spaces, free parking space, jail space, go to jail space, and tax spaces.

Starting Point (Go)

Go is the first tile of the board and every player has to start from this tile (Figure 2) at the beginning of the game. After the beginning whenever a player passes to this tile bank should give this player 200 monopoly units.

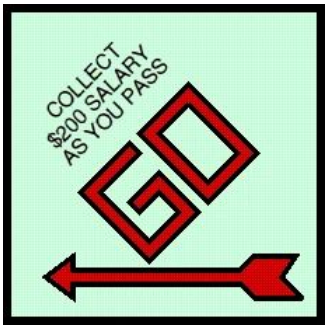

Figure 2: Starting Point (Go) Square

Properties

Because the main objective of the game is to bankrupt all opponents, players should have properties to get other opponent's money. "Properties may be bought in one of 3 ways: landing on the property space and buying it, being the highest bidder in an auction for property, or buying it from an opponent in a trade. Properties may also be received from bankrupted players, provided the Bank didn't bankrupt them".

Whenever a player buys or takes ownership of some property, s/he needs to get title deeds (Figure 3), which has all the information about the bought property.

Figure 3: Title deed example

There are **groups that are labeled with eight different colors, four railroads, and utilities such as Electric Company and Water Works** on the board.

Brown Property Set

These properties are placed at the beginning of the game. They are the cheapest properties of the game. In the standard version of the Monopoly Mediterranean Avenue (60 Unit) and Baltic Avenue (60 Unit) are the members of this set (Figure 4).

Figure 4: Brown Property Set Title Deeds

Light Blue Property Set

Although this property set is more valuable than brown ones, still they are one the cheapest property sets of the game. They are placed right after the brown property set. In the standard version of the Monopoly Oriental Avenue (100 Unit), Vermont Avenue (100 Unit) and Connecticut Avenue (120 Unit) are the members of this set (Figure 5).

Figure 5: Light Blue Property Set Title Deeds

Pink Property Set

They are more valuable than light blue ones but less valuable than orange ones. They are placed at the beginning of the second side of the board. In the standard version of Monopoly St. Charles Place (140 Unit), States Avenue (140 Unit) and Virginia Avenue (160 Unit) are the members of this set (Figure 6).

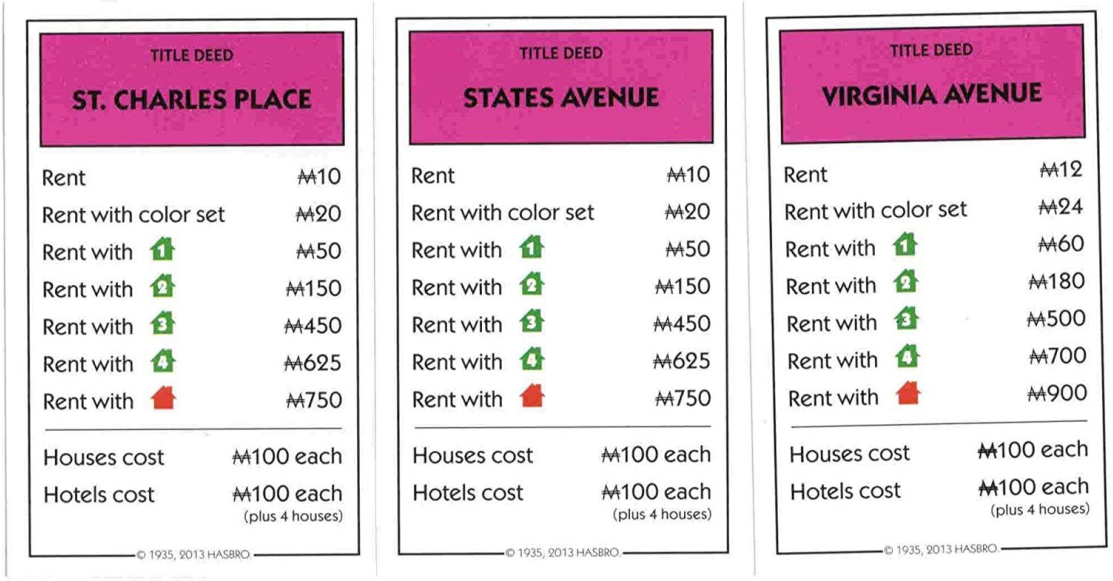

Figure 6: Pink Property Set Title Deeds

Orange Property Set

They are the most valuable properties of the second side of the board. They are placed after the pink property set. In the standard version of Monopoly St. James Place (180 Unit), Tennessee Avenue (180 Unit) and New York Avenue (200 Unit) are the members of this set (Figure 7).

Figure 7: Orange Property Set Title Deeds

Red Property Set

Red property set is more valuable than the orange ones but properties inside the red group are the least valuable ones of the third side of the board. They are placed at the beginning of the third side. In the standard version of Monopoly Kentucky Avenue (220 Unit), Indiana Avenue (220 Unit) and Illinois Avenue (240 Unit) are the members of this set (Figure 8).

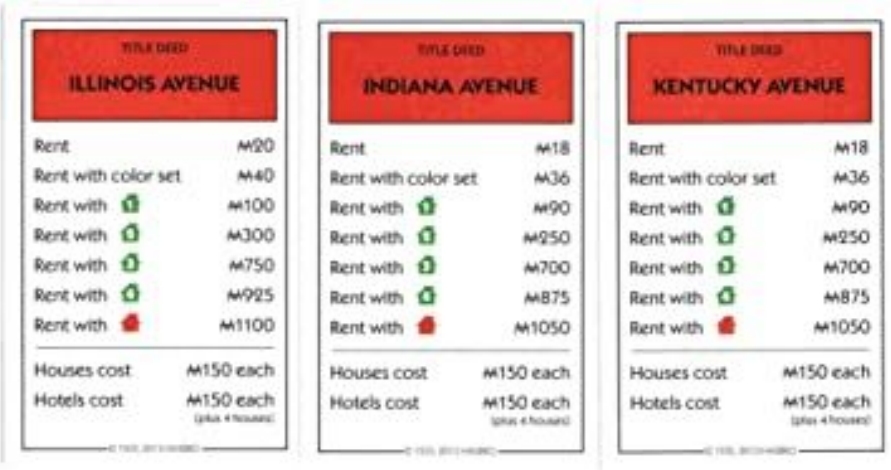

Figure 8: Red Property Set Title Deeds

Yellow Property Set

They are the most valuable properties of the third side of the board. They are placed at the end of the third place. In the standard version of Monopoly Atlantic Avenue (260 Unit), Ventnor Avenue (260 Unit) and Marvin Gardens (280 Unit) are the members of this set (Figure 9).

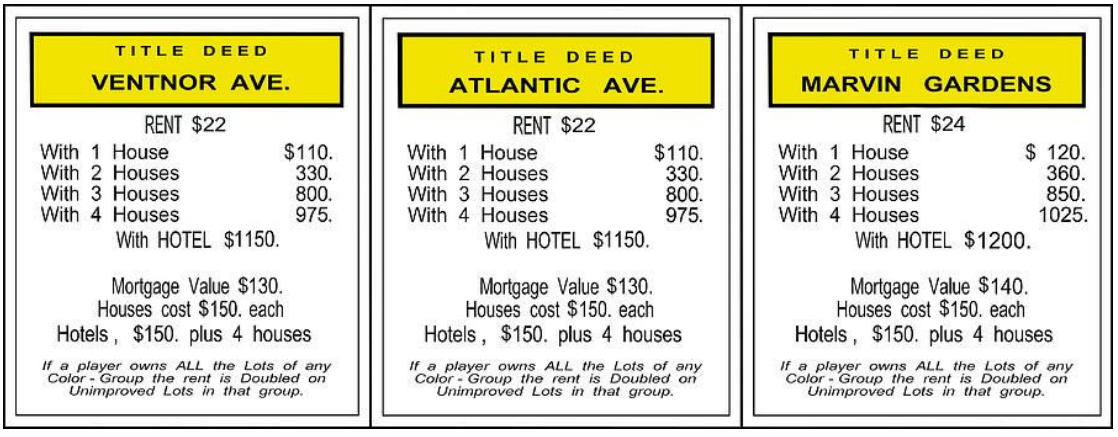

Figure 9: Yellow Property Set Title Deeds

Green Property Set

These properties are in the top two of the game. They are placed at the beginning of the fourth and the last side of the board. In the standard version of Monopoly Pacific Avenue (300 Unit), North Carolina Avenue (300 Unit) and Pennsylvania Avenue (320 Unit) are the members of this set (Figure 10).

Figure 10: Green Property Set Title Deeds

Blue Property Set

This set is the most valuable set of the game. They are placed in the last piece of the last side. In the standard version of Monopoly Park Place (350 Unit) and Boardwalk (400 Unit) are the members of this set (Figure 11).

Figure 11: Blue Property Set Title Deeds

The Four Railroads

The railroads are one of the properties. They are more profitable than other properties. Reading Railroad, Pennsylvania Railroad, B. & O. Railroad, and Short Line are the railroads of the standard Monopoly game (Figure 12).

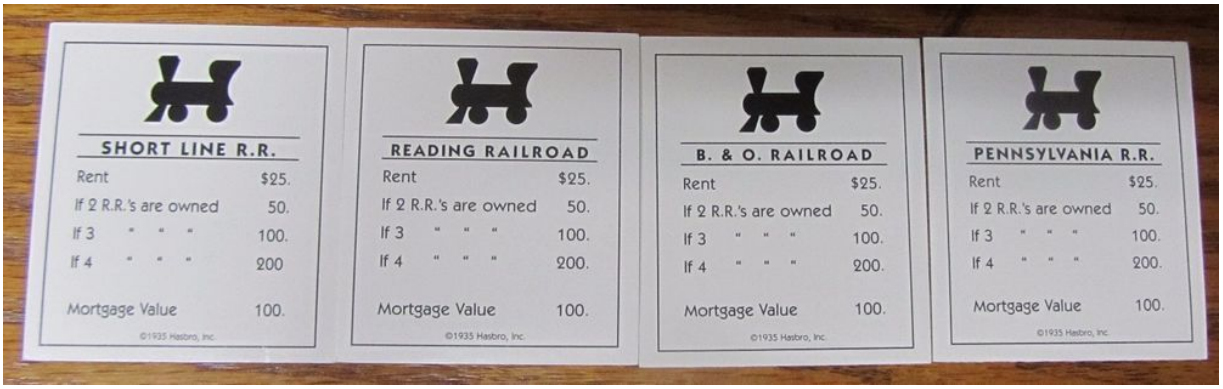

Figure 12: Railroads in the Standard Version of Monopoly

Electric Company

It is one of the cheapest properties. Also, buildings can't be built in this place. It is placed between St. Charles Place and State Avenue properties in the standard version of Monopoly.

Figure 13: Electric Company

WaterWorks

It has the same value as the Electric Company property. It is located between Ventnor Avenue and Marvin Gardens properties in the standard version of Monopoly.

Figure 14: Water Works

Free Parking

Figure 15: Free Parking

When a player comes to Free Parking, nothing changes. He/She only parks on the ground and waits for the next round.

Jail

Figure 16: Jail

One of the most important parts of the game is going to jail. In some game genres, this part may be called prison. Jail is usually a bad thing. Because when you go to jail, you cannot play 3 rounds. In some cases, waiting in jail may give you an advantage. Example: If your opponent has many properties in front of you and you are likely to fall into one of them, you can expect the opponent to pay you 3 rounds of rent instead of leaving the jail.

To go to jail as follows:

1. If the number of dice you roll coincides with the jail tile, you will remain in jail.
2. If you withdraw the go to jail card from the chance cards, you will go to jail.
3. If you roll double dice 3 times in a row in the same round, you will go to jail.

Possibilities to exit jail:

1. When you wait for three rounds, you have the right to play the next round and you exit the jail.
2. If you have a "Get Out of Code" card that you won before, you can use it to exit.
3. You can get out of jail by paying a certain amount of money. You can pay to get out of jail, especially if you are early in the game. Because it is very important to buy property at the beginning of the game. Waiting for you while other players buy 3 rounds of properties may cause you to lag behind the game.

Visitor

If you have not been sent to the Jail and come to this section while the game is in progress, you will be counted as a visitor and will not be penalized. When it is your turn, you continue playing.

One of the most important parts of the game is going to jail. In some game genres, this part may be called prison. Jail is usually a bad thing. Because when you go to jail, you cannot play 3 rounds. In some cases, waiting in jail may give you an advantage. Example: If your opponent has many properties in front of you and you are likely to fall into one of them, you can expect the opponent to pay you 3 rounds of rent instead of leaving the jail.

To go to jail as follows:

4. If the number of dice you roll coincides with the jail tile, you will remain in jail.
5. If you withdraw the go to jail card from the chance cards, you will go to jail.
6. If you roll double dice 3 times in a row in the same round, you will go to jail.

Possibilities to exit jail:

4. When you wait for three rounds, you have the right to play the next round and you exit the jail.
5. If you have a "Get Out of Code" card that you won before, you can use it to exit.
6. You can get out of jail by paying a certain amount of money. You can pay to get out of jail, especially if you are early in the game. Because it is very important to buy property at the beginning of the game. Waiting for you while other players buy 3 rounds of properties may cause you to lag behind the game.

Visitor

If you have not been sent to the Jail and come to this section while the game is in progress, you will be counted as a visitor and will not be penalized. When it is your turn, you continue playing.

Go to Jail

Figure 17: Go to Jail

Whenever a player is placed in this space then s/he needs to go to the Jail space of the game. This space is placed in the third corner of the board in the standard version of the Monopoly.

Tax

This is the space that if you are placed in this space then you have to pay a predefined tax. Income Tax and Luxury Tax are the tax spaces that are in the standard version of the Monopoly.

Figure 18: Income and Luxury Taxes

Chance and Chest Spaces

Whenever a player comes over these spaces, then, s/he needs to take a card from either the community chest card set or chance card set.

Figure 19: Chance and Chest Spaces

How to play monopoly

In turn, all players roll two dice together. The player with the highest roll starts the game and the game continues clockwise.

When it's your turn, roll both dice and move your pawn clockwise as much as the dice you rolled. Two or more pawns can stand on the same digit. Depending on the house where your pawn is, you must do one of the following:

1. Buying plots and other property (if not owned by another player).
2. Asking the banker to auction a property (if you do not want to buy it).
3. Paying rent (if another player has bought that property).
4. Paying taxes.
5. Taking a Chance or Public Fund card.
6. Entering the Jail.

If you own the title deeds of all plots in a color group, you can build houses and hotels on these plots.

If you run out of money and have to pay a loan, you can mortgage or sell your properties to the bank. If you owe more debt to the bank or another player than you can pay by selling your goods, you will go bankrupt and withdraw from the game.

Players cannot borrow money from each other or lend money to each other. However, if a player wishes, he can agree to take a property from another player instead of money for his loan.

If you roll double dice (that is, the same number comes on both dice you roll), move your pawn forward as usual and move according to the point on which you stand. After doing what you have to do, roll the dice again and continue the game. If you roll double dice three times in a row, on the third time you will enter the Jail and cannot advance your pawn.

The last player remaining in the game without going bankrupt wins the game.

Buying Property

There are three types of property: Properties, Railroads, Utilities.

If you stand on unclaimed property, you can buy it. If you decide to buy the property, you must pay the price written on this household to the bank. The bank will give you the Title Deed card as proof of your ownership.

Owning property allows you to receive rent from all players who have paused on your property. It is an advantage to have all the plots in a color group, i.e. a "Monopoly". In

this case, you can get more rental income by establishing a house or hotel on these plots.

Pay Rent

If you stand on an owned property, you have to pay the property owner. If the property is mortgaged, the rent is not paid.

The owner must claim the rent before the next player rolls the dice. The rental fee is written on the Title Deed Card of that property and changes according to the number of buildings on the property.

If you own the title deeds of all plots in a color group, the rental prices of all vacant plots in that color group (i.e. plots without houses or hotels) doubles. Even if your properties in a color group are mortgaged, you can get twice the rent for unencumbered ones.

Utilities

Utilities can be bought and auctioned like any other property. If you stop on an owned Utility, you will pay rent based on the damage you've inflicted to get to that house.

If the utility owner has only one, the rent is 400 times the dice you rolled.

If the utility owner has both, the rent will be 1,000 times the dice you rolled.

Railroads

Railroads are bought and auctioned like any other property. If the railroads you are stopping are owned, you must pay the amount indicated on the Deed Card.

The amount to be paid is specified on the Deed Card and depends on how many Railroads cards that player has in total.

Building a House

When you have all the plots in a color group, you can buy a house (or houses) and install them on any of these plots. The price of a house is stated on the Title Deed card. While it's your turn, you can buy houses (or hotels) between rows of other players; however, you need to build a house in a balanced way.

You cannot build your second house on any plot without installing a house on each plot of the same color group. As long as you can afford to pay, you can buy as many buildings as you want from the bank!

If any plot of a color group is mortgaged, you cannot build a house in that group. You need to place your houses and hotels on the property you want to build.

Setting Up a Hotel

In order to buy a hotel, you must have four houses on each plot of a color group. In order to buy a hotel, you have to give the four houses on your property to the bank and also pay the amount specified in the Deed Certificate. You can only set up one hotel on a Property.

Buying a Building

If the bank has no houses to sell, you must wait for other players to return or sell houses to be able to buy a house.

If there are a limited number of houses or hotels left and two or more players want to buy more buildings than the number of buildings held by the bank, the banker will auction the houses or hotels, starting from the lowest amount stated on the Deed Cards in question. The auction is held individually for each building, and the highest bidder buys the building.

Money Trouble

If you don't have enough money left, you can do the following to raise funds:

1. Sell the building
2. Mortgage a property
3. Selling properties, railroads or utilities (even if it is mortgaged) to another player for an agreed price.

Property Sale

You can sell vacant lots, railroads, and utilities to another player at a mutually agreed price. However, if any of the plots belonging to a particular color group have a building, you cannot sell any plot of that color group to another player. First of all, you have to sell the buildings on these plots to the bank.

Houses and hotels are sold for half of the amount purchased to the bank (shown on the Title Deed Card). You can sell your buildings while it's your turn or between the other players' turn.

Selling house

You must sell the houses the same way they purchase them. So if you are going to sell more than one house to raise funds, all of these houses should not be on the same plot.

Selling hotel

While the hotel is being sold; the bank pays the player half of the hotel price and half of the four house prices the player gave to the bank to buy a hotel. If necessary, funds can be created by turning hotels into homes. In this case, you sell a hotel to the bank and buy four houses, along with half the hotel price.

Sale of a mortgaged property

A mortgaged property can be sold to another player at a mutually agreed price. The new owner can immediately remove the mortgage by paying the mortgage amount and 10% interest. If he decides not to remove the mortgage immediately, he must pay the 10% interest immediately. In this case, when he decides to cancel the mortgage, he pays 10% interest once again in addition to the mortgage amount.

Bankruptcy

If you have more debt than you can pay by selling your goods, you will go bankrupt and withdraw from the game.

If you owe a bank:

Give all your Deed Cards to the bank. The bank auctions them one by one and sells them to the highest bidder (s). If you have it, you must also put your "Get out of Jail for free" card at the bottom of the relevant deck.

If you owe another player

If any, sell all your houses and hotels for half of the amount you bought to the bank (shown on the Title Deed Card). Transfer all your money, Title Deed cards, and, if any, your "exit Jail for free" card to this player.

Chance and Community Chest

The player standing in one of these squares draws the card from the top of the respective deck and applies what is indicated on the card. After following the instructions written on the card, you should place the card at the bottom of the deck, turning the printed side down.

You can just keep the "Quit Jail for free" card until you need it or sell it to another player.

If the card you draw tells you to move to another digit, move in the direction of the arrow. If you pass the start, get the start fee. If the card you withdraw sends you to jail or tells you to go back to a house without going through the Starting point, you will not be able to get the start money.

Proposed System

Overview

In the current version of the Monopoly game, there are two types of cards, community chest, and chance cards, that can add fun to the flow of the game. Also, the flow of the game starts to repeat itself after some point, players roll dice, buy properties, build houses and hotels, and etc. Therefore, we plan to not bore the players by exposing them to the boring flow of the game. In order to do that, we added 7 special characters to the game, in which each of them has their own strongness and weaknesses. In addition to the special characters, there is also a quest feature in the game, in which the players are to complete the quests and earn rewards depending on the difficulty of the game. The proposed system is not just limited to these features only. There are also special characters and buildings added to the game so that the flow of the game is going to be changed. Within the system that is proposed, we plan to make the game more enjoyable, fun, and not boring.

Functional Requirements

7 Special Characters

Fugitive

The fugitive can escape from jail. However, after escaping two times, he will be sent to the highly protected prison where the fugitive cannot do anything for the 5 turns. After 3 turns, he starts waiting for his parole approval in which he plays jail games to release. Moreover, there is also a limitation in his escape. When the fugitive is sent into jail, he waits for 1 turn, and then, he chooses 2 locations on the map where he wanted to go. Unfortunately, he does not have a compass thus, he can end up going somewhere else. Therefore, the game also chooses a random place from the map. When the 3 locations are selected, the fugitive rolls the dice and according to the mod of 3, the fugitive goes to the according to location.

Traveler

The traveler has two special features. Firstly, during the game, he will have 5 chances to go anywhere he wants after waiting for 1, 2, or 3 rounds. He rolls the dice and if the sum of the dice is 5 or 6, he waits for 1 round (9/36 possibilities); if the sum is 2, 3, 4, 7, 10, 11, or 12, he waits for 2 rounds (18/36 possibilities); and if the sum is

8 or 9 he waits for 3 rounds (9/36 possibilities). When he chooses where to go he cannot change and if the location is bought by some other player, he pays the tax. Secondly, if the traveler dominates the travel agencies' cards (train, ship, plane, etc.), he can go anywhere if he stopped at one of the agencies without losing a turn. His special features do not affect each other.

Coach-Sport Guy

Everybody loves the sport. Therefore, everybody loves the coach. During the game, he can buy land registers or real estate much cheaper.

Thief/Fraud

Fraud's special feature is tax evasion. He can reduce the amount of his payment, and each time he reduces the payment, the payment getting smaller and smaller. While the increases tax evasion, the risk of being caught also increases. When he gets caught, according to the caught risk stage the fine increases, and he is sent to jail for 3 turns. To catch the fraud, game and the fraud rolls the dice, whoever gets the higher sum wins. If the game gets the higher point, fraud gets the penalty, otherwise, fraud successfully evades tax. Higher risks mean more dice rolls for the game.

Car Guy

He can move forward or backward on the map. However, he cannot move backward two times in a row unless he possesses at least one travel agency. He cannot collect the start-pass bonus without completing the full round of the map (forward or backward). When the character stops at the car parking, he can roll the dice again.

Rich

A rich lifestyle means having more money and more expenses. During the game Rich character puts his money to the bank and with each completing round (one full circle of the map), he gets more interest than any other player. Hence, he earns money much easier than the others he spends much more than any other player. Each land register, rent, real estate costs him more than other players. Additionally, if he loses money (buys something or pays rent) three turns in a row, he loses additional money to the bank. Start-pass bonuses do not affect the last condition.

Homeless

The homeless character is the most difficult character to play in the game. He starts with almost no money and travels around. During the early stage of the game (first three rounds), he pays no rent and tries to save money. If he comes to free land

registered homeless character and the game rolls the dice. If the homeless' sum is higher than the game he earns half of the land register price. After three rounds he earns the 1.5 start-pass bonus and starts collecting a start-pass bonus. In the early stage homeless cannot collect the start-pass bonus. After the early stage, he plays like a normal player but only pays fewer rents until he completes one land register set. When he builds real estate, he earns a highly generous money bonus from the bank, and with two hotels homeless automatically wins the game. On the other hand, if the homeless could not become wealthy or bankrupted, he is still able to play the game. He still earns a start-pass bonus but stops paying rent until he can buy a land register and collect 4 rents.

Special Features

In our Monopoly game, we have additional features.

The players can trade lands whenever they want. Once they agree on the trade, their process will be confirmed during one player's turn. In other words, once they trade a land, they obtain the land after waiting for one of the player's turn.

Additional features listed in the following subtopics:

Special Cards

There are several additional cards that will affect the course of the game. These cards are hidden in the pile of lucky cards. Cards can be used immediately, some of them can be used as bluffs and some can be used later, it is up to the players.

Natural Disaster Cards

These cards divide into three categories. In the first category, cards immediately cause an earthquake in the mentioned locations. In the second category, cards cause an earthquake in the mentioned location whenever the player decides. In the third category, the player decides when and where to cause an earthquake.

When a natural disaster card is played, the location loses half of the value, and thus, the player loses half of the profit from that location.

Some natural disaster cards also destroy the buildings into the lands.

Tips: These cards can be used for damaging other players' fortunes or to make them deceive. For instance, a player holds a card that can damage one of his locations. He can trade this location with someone else and then he can use the card.

Blame Cards

During the game, any player can accuse the other player to hurt the profile of the player. When a player accuses the other player three scenarios can happen. Firstly, the player that is being accused can accept the charges and lose some of his fortune (the game phase decides how much the player loses). Secondly, the player can refuse the charges. In this case, the blame cards become effective. If the player that is being accused, refuses the charges and the player that accuses shows the blame cards, the accused player loses much more fortune (the game phase decides how much the player loses). If the accusing player does not have a blame card, in other words, if the player bluffs and the other player calls the bluffs, the accusing player loses fortune repeatedly until he can profit from his lands. The last scenario is that if the accused player calls the bluff and immediately plays the blame card, the accusing player loses much more money (the game phase will decide) until the player collects rent from one of his lands.

There are two types of blame cards: the bluffs and the accusation cards.

Profit Cards

These cards are divided into two categories according to effectiveness. Some of them can be used any lands that the player wants while others can only be used for specific locations. Profit cards can be used one time whenever the player decides to use them.

When the profit cards are used, according to the cards statement the land can profit up to three times the land's previous profit. Profit cards are 1.5 more effective to the locations that are damaged by the natural disaster cards.

Special Buildings

These buildings are located in specific lands (the game randomly chooses the lands before the game starts). When a player stops at these specific lands they can use these buildings to increase their fortune. The special cards, such as natural disaster cards, also affect these buildings. If the buildings are destroyed, the players lose the money that they invest. Additionally, with more money land's profit also increases.

Banks

During the game, players can invest their money into the bank after paying the fee. Thus, during the game, their income will be increased. With more money, the land's profit will be increased. Even if the player does not have the land register they can still invest money into the banks.

Custom Station

During the game, random items will appear in the custom stations. The player that stops at the land and pays the custom fee can obtain the items. Each item will have different effects and different profits to the players.

Items are divided into two categories. The first one is the sculptures. When a player pays the sculpture fee in the custom station, he can build it any land that the player wants. When the sculpture is built, the land's rent fee is increased (the increase determined by the game). If the land is damaged by the disaster cards, the sculpture can be damaged as well. When the disaster card is played to damage the land, the owner of the land rolls the dice, and if the sum is odd sculpture survives, if the sum is even sculpture destroyed. Similarly, if the profit card is used to the land, the sculpture's benefit increases. The second one is the artifacts. They can be put into specific lands to increase the land's rent fee. They are not affected by any cards.

Special Quests

Seven Wonders

The game has one special quest that if the player buys all the seven wonders he automatically wins the game. These lands are not cheap and they have high rents and during the game, their costs will change.

Players can team up to complete this quest. If a team completes the Seven Wonders Quest, they earn money, but the game continues.

Downtown Roll

Before the game starts the game chooses 10 lands and random roll combinations for those locations (2 combinations per land). When a player rolls the specific combination and stops at that land, the player will win a jackpot.

Each jackpot can be traded with money from the bank. If a player possesses the 3 jackpots, the player will earn a huge fee. Jackpots can be traded between players and can be used to get out of jail.

Landlord King

If a player buys 6 locations during one round, the player becomes the Landlord in which he can build houses or hotels without completing the land sets.

Bankruptcy

If a player cannot earn any profit from other players for 4 full map rounds and he pays at least 6 rents, the player becomes withdrawn. When a withdrawn player pays two more rents or is affected by the natural disaster cards, the bank collects 25% of the player's fortune. This will continue until the player collects rent.

This quest is not effective for the homeless character.

During the game phase, the game can shift the rules of becoming a withdrawn player.

Comeback

When a withdrawn player plays a profit card or is affected by a profit card, they earn money from the bank.

This quest is not effective for the homeless character.

Jail Games

There are two types of games that players can play during jail time. During the jail, game players can put their lands, money, or buildings to earn more prices.

Auction

During the game, players can sell their lands via auction.

Non-functional Requirements

Usability

The game will be implemented in such a manner that everyone who wants to play the game will not be having any difficulties understanding the game and thus will not lose a considerable amount of time just for learning how to play. Also, the application is going to be implemented by considering the simplicity of navigating through the game menu. We will try to ensure that the players will enjoy the game while using the application safely and efficiently.

Reliability

Since software applications are prone to produce errors more than expected, we will make the game as bug-free as we can. We, developers, know that even the best software application can have different bugs in it, but we also know that the aim is to reduce the bugs as much as we can. If a user tries to do something that is not

allowed, for instance, trying to move their character even if it is not their turn, an appropriate error message will be shown. Different unit tests will be implemented to ensure that the program is working correctly and in the desired way.

Performance

Since the game will be implemented using the Electron framework of JavaScript, it will actually run embedded into a web browser, by combining the Chromium rendering engine and Node.js runtime. Considering the fact that the game will require too much rendering in the background, we plan to not decrease the performance of the game and provide a compatible game for as many players as we can.

System Models

Use-Case Models

Start Game Use-Case Model

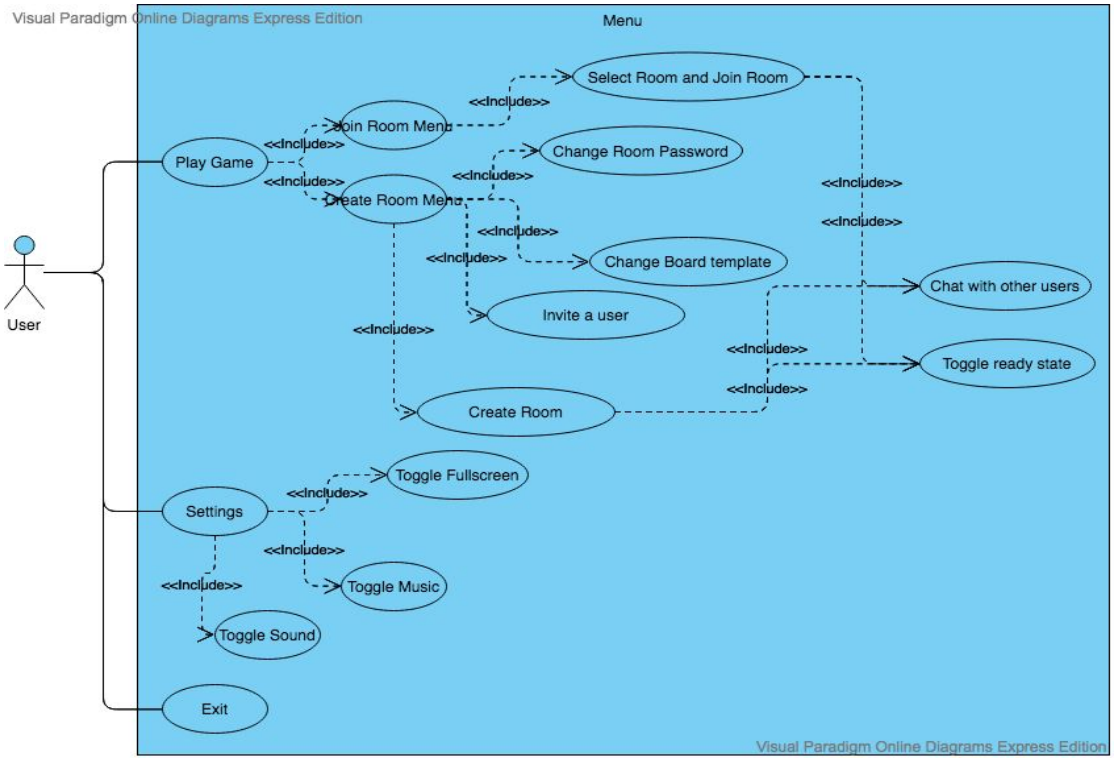

Figure 20: Start Game Use-Case Model

Use Case Name	Settings
Actor	Player
Possibilities	1- Adjusting the screen 2- Toggle Music 3- Adjusting the sound
Flow of events	Player changes the different game settings including screen size, music, or sound volume.
Enter Condition	The player must enter the Settings option
Exit Condition	Player returns the previous menu

Use Case Name	Play Game
Actor	Player
Possibilities	Player either creates a game room or joins an existing room.
Flow of events	1- Player starts the ‘Play Game’ then chooses either ‘Join a Game’ or ‘Create a Game’
Enter Condition	Player must click the Play Game button
Exit Condition	Player returns the main menu

Use Case Name	Join a Room
Actor	Player
Possibilities	1- Select a room 2- Back to the previous menu
Flow of events	Player clicks the ‘Join a Room’ button then chooses the room. After that players can chat with other players and by pressing the ‘Ready’ button.
Enter Condition	Player must select the room and enter the password correctly.
Exit Condition	Player returns the previous menu

Use Case Name	Create a Room
Actor	Player
Possibilities	1- Create Room 2- Invite Players
Flow of events	Player can create a game room where the player can adjust the board template or adjust the password of the room. Players can invite players to the created room. After that players can chat with each other and toggle the ‘Ready’ button to start the game.
Enter Condition	Player must enter the Room password, decide the board template and the player can invite the other players.
Exit Condition	Player returns the previous menu.

Play Game Use-Case Model

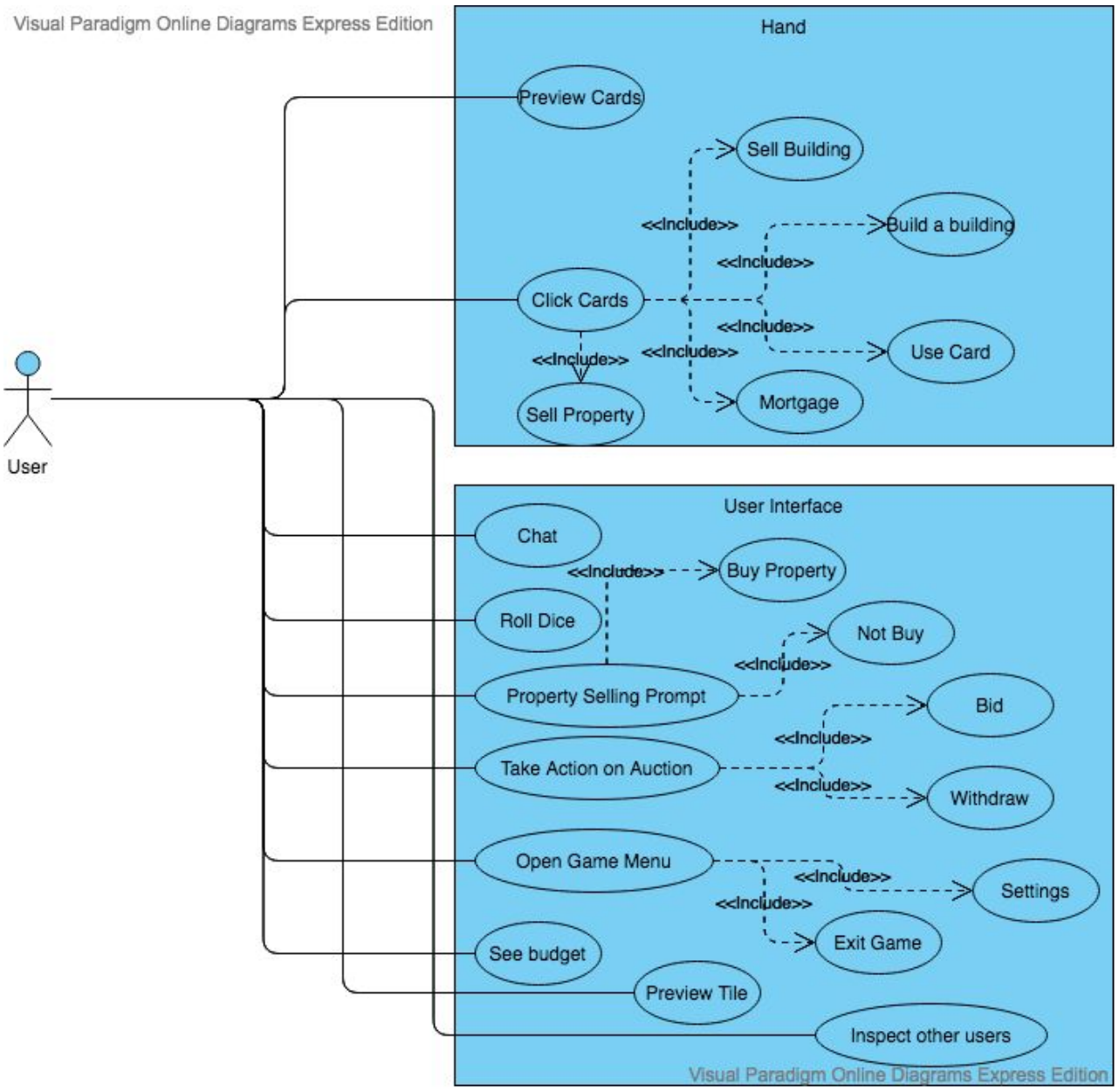

Figure 21: Play Game Use-Case Model

Use Case Name	Preview Cards
Actor	Player
Possibilities	Player can see the registered lands, free lands, player's special cards, and other players' registered lands.
Flow of events	During the game, the player clicks the 'Preview Cards' button.
Enter Condition	During the game, the player clicks the 'Preview Cards' button.
Exit Condition	Player returns to the board.

Use Case Name	Click the Cards
Actor	Player
Possibilities	The player can click registered lands, free lands, and own special cards.
Flow of events	1- The player clicks the lands. The clicked land shows the buildings, rent fee, and effective special effects on the lands. If the land is the player's property, the player can build buildings. The player can also mortgage the land, sell the building or land. .If not, the player can only see the effective cards on the land, items, and special buildings, and rental fee. 2- The player clicks the player's own special cards. The player decides to use it or not.
Enter Condition	Clicking the cards or lands.
Exit Condition	Player clicks the exit button or clicks the outside of the card. Player returns the board.

Use Case Name	Chat
Actor	Player
Possibilities	Player chats with everyone Players chat privately. Player can see the messages
Flow of events	1-Player clicks the chat button. Chooses to talk publicly. Write the message to the message blog. Clicks send. Every player sees the message. 2-Player clicks the chat button. Chooses a player to chat with. Writes the message to the message blog. Clicks send. Only the selected player sees the message. 3-Player clicks the chat button. Sees the unread messages. Selects the desired one. Enters the conversation with the owner of the message.
Enter Condition	Player clicks the chat button.
Exit Condition	Player clicks the exit button or outside the chatbox. Player returns the board.

Use Case Name	Property selling
Actor	Player
Possibilities	Player can buy the stopped land or not.
Flow of events	Player plays according to the dice or cards and if the stopped land is unregistered, the player can buy the land or not.
Enter Condition	Player stops at the unregistered land.
Exit Condition	Player clicks either the not buy button, exit button, or outside the property box. Player returns the board.

Use Case Name	Auction
Actor	Player
Possibilities	Player can bid in the auction or put an item or land in the auction.
Flow of events	1-Player clicks the auction button. Sees the current auction items, lands, and current bids. Clicks one of the items or lands and places a bid. 2-Player clicks the auction button. Selects my auctions and sees the player's bidded auctions and the player's own auction items and lands. The player clicks one of the own auctions and sees the current bid. The player can sell anytime the player wants.
Enter Condition	Player clicks the auction button and sees the auctions.
Exit Condition	Player clicks the exit button or the outside of the box.

Use Case Name	Open Menu
Actor	Player
Possibilities	Player can change the game settings or exits the game
Flow of events	1- Player clicks the menu option. Player clicks the settings. Player changes the game settings. 2- Player clicks the menu option. Player clicks the exit game. An additional box pops up and asks 'Are you sure?'. Player chooses yes or no. If the player chooses no, player returns the previous menu, else if the player chooses yes, the game closes.
Enter Condition	Player clicks the menu option or 'ESC' key.

Exit Condition	Player clicks the exit button or clicks outside of the box.

Use Case Name	Roll the dice
Actor	Player
Possibilities	When the player’s turn comes, the player can select the roll the dice button. Player cannot select before the player's turn. If the player waits long enough, player loses the turn.
Flow of events	1- Player cannot click the roll dice button because it’s not the player's turn. 2- Player clicks the button and gets the dice. The player chooses how to play according to the dice. 3- Player don’t click the roll dice button and after some time, the player loses the turn.
Enter Condition	When the player’s turn comes, player can click the button.
Exit Condition	1-Player waits long enough during the player’s turn 2-Player clicks the end turn button.

Object Class Diagram

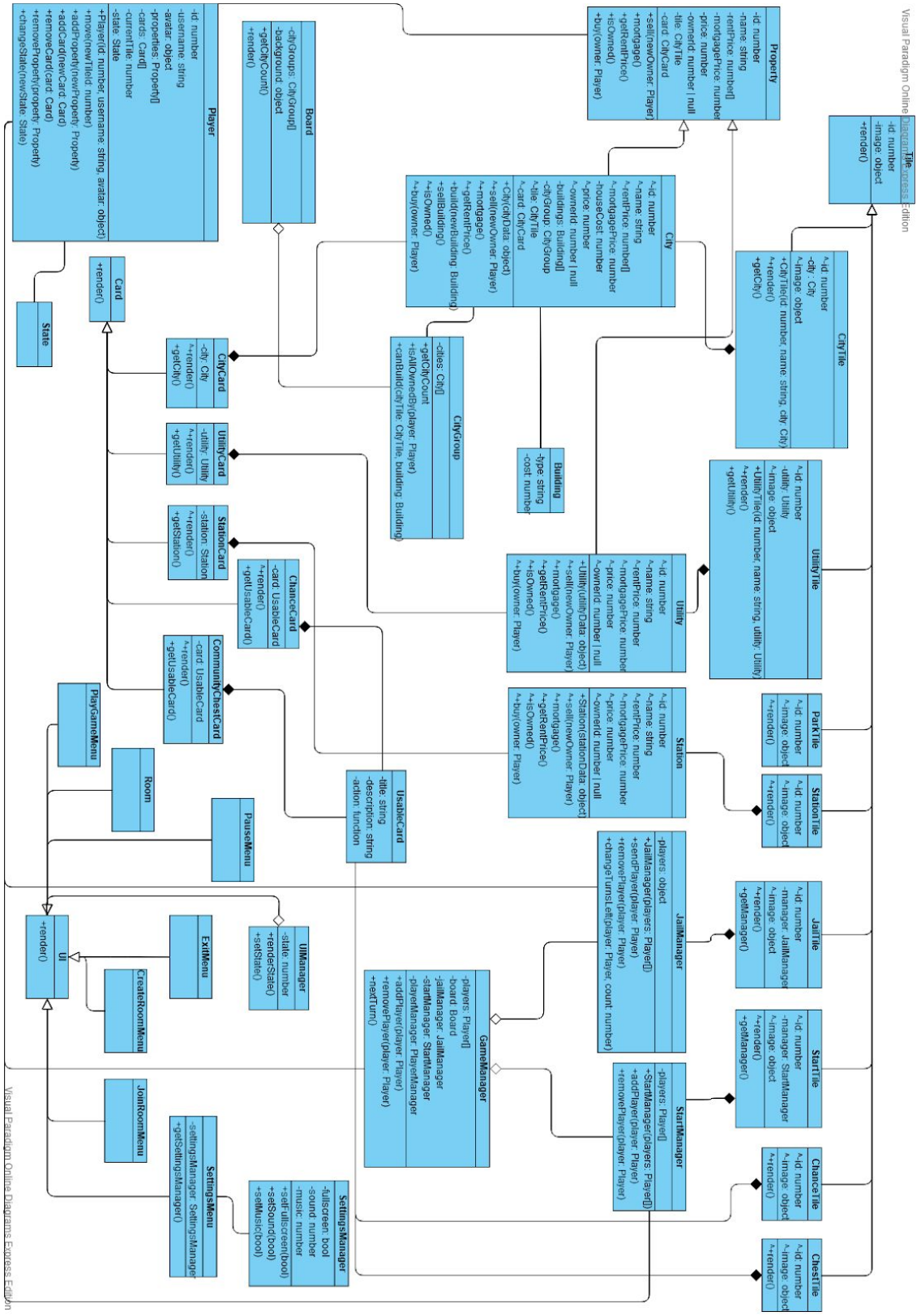

Figure 22: Object Class diagram

Tile (Interface)

Tile is an interface which contains “id” and image properties, and a method called render. “render” method handles the rendering of all classes that take “Tile” as an interface on board. In this render method, the image property will be used.

CityTile (Class)

CityTile class will be used in all cities that exist in the game. It will handle the drawing procedures of city tiles onto the game board. The image property will be used in order to separate this tile from tiles. This class has got a reference to the city which it shows via a property called the city.

UtilityTile (Class)

UtilityTile class will be used in all utilities that exist in the game such as “Water Works” and “Electric Company”. It will handle the drawing procedures of city tiles onto the game board. The image property will be used in order to separate this tile from tiles. This class has got a reference to the utility which it shows via a property called utility.

ParkTile (Class):

ParkTile class will be used in order to show the park tile in the original monopoly game. It will handle the drawing procedures of this tile on to the game board. The image property will be used in order to separate this tile from tiles.

ChanceTile (Class):

ChanceTile class will be used in order to show change tiles in the monopoly board. It will handle the drawing procedures of this tile on to the game board. The image property will be used in order to separate this tile from tiles. It has got a connection to UsableCard class.

StationTile (Class):

StationTile class will be used in all four stations that exist in the game such as “O. Railroad” and “Reading Railroad”. It will handle the drawing procedures of station tiles on to the game board. The image property will be used in order to separate this tile from tiles. This class has got a reference to the station which it shows via a property called utility.

JailTile (Class):

JailTile class will be used in order to show the jail tile. It will handle the drawing procedures of this tile on to the game board. The image property will be used in order to separate this tile from tiles. It has a connection to JailManager class in order to process players coming to this tile.

StartTile (Class):

StartTile class will be used in order to show the start tile. It will handle the drawing procedures of this tile on to the game board. The image property will be used in order to separate this tile from tiles. It has a connection to StartManager class in order to process players coming to this tile.

Property (Interface):

Property is an interface that must be inherited by all the properties such that Cities and Utilities in-game. It has got different properties such as id, name, rent and mortgage price, ownerId, tile, and card which is a reference to the card object of this property. It has got methods that change these properties such as sell and buy.

City (Class):

City class inherited from the Property interface. It has got all the methods inherited from Property, in addition, it has got buildings and cityGroup properties, as a consequence it has got methods like build and sellBuilding. This object will be used in CityTile and CityCard for references.

Utility (Class):

Utility class inherited from the Property interface. It has got all the methods inherited from Property. This object will be used in UtilityTile and UtilityCard for references.

Station (Class):

Station class inherited from the Property interface. It has got all the methods inherited from Property. This object will be used in UtilityTile and UtilityCard for references.

JailManager (Class):

JailManager's class handles the logic about jail. It will be used in GameManager. It has got different methods like sendPlayer which sends a player to jail and changeTurnsLeft which changes the turn left in jail for a player. It uses an array of players from GameManager.

StartManager (Class):

StartManager class handles the logic about the starting tile. It will be used in GameManager. It will handle different events that happen when a user passes from the starting tile. It uses an array of players from GameManager.

GameManager (Class):

GameManager class handles the logic about the game loop which starts after the main menu and room selection phases. It will handle game events and control other managers such as StartManager and JailManager. It contains an array of players.

Board (Class):

Board class contains cityGroups in cityGroup array. It has got a background. getCityCount method traverses all cityGroups and returns the total number of cities in the game. In the end, there is a render method that draws a board into the screen.

Building (Class):

The building is a class that has a string property called type and an integer property called cost. It represents buildings in general.

CityGroup (Class):

CityGroup is a class that contains cities in the same group. It can check whether the group is owned by a player or not and determine if a special building can be built on a cityTile.

Card (Interface):

The card is an interface which has got only a render method in order to draw cards in the user's hand.

CityCard (Class):

CityCard is a class that was inherited from Card interface. It has a render method and a property which contains the city reference. You can get this reference by calling getCity method.

UtilityCard (Class):

UtilityCard is a class that was inherited from Card interface. It has a render method and a property which contains the utility reference. You can get this reference by calling getUtility method.

StationCard (Class):

StationCard is a class that was inherited from Card interface. It has a render method and a property which contains the utility reference. You can get this reference by calling getStation method.

ChanceCard (Class):

ChanceCard is a class that was inherited from Card interface. It has a render method and a property which contains the chance reference. You can get this reference by calling getUsableCard method.

CommunityChestCard (Class):

CommunityChestCard is a class that was inherited from Card interface. It has a render method and a property which contains the chest reference. You can get this reference by calling getUsableCard method.

UsableCard(Class):

UsableCard class will be responsible for cards that can be consumed such as chest cards, chance cards, secret cards and quest cards. It will be used by tile classes and card classes.

UI(Interface):

All the menu and UI elements will be inherited from this class.

UIManager(Class):

All the menu and UI elements will be drawn by this manager class. In order to store the current state of UI, it has got a property called the state and there are methods like getState and setState in order to change this property. It draws classes inherited from UI.

SettingsManager(Interface):

This class will handle all the changes in settings. It will connect to the SettingsUI in order to provide an interface for the manager.

SettingsMenu(Interface):

This class will handle all the drawings in the settings menu. It will connect to the SettingsManager which handles all the settings changes. It will send all the inputs and choices to the SettingsManager.

Player (Class):

The Player class contains all the information about a player. It has got different properties such as id, username, avatar, properties, cards, currentTile, and state. Properties and cards hold the stuff the player has got. Avatar and username will be used to differentiate the player from other ones. It has got different methods that change these properties such as addCard, removeProperty, and move.

Dynamic Models

Sequence Diagrams

Settings Sequence Diagram

In this diagram, a sequence diagram for the settings that will be added to the project is given. Firstly, in order to reach the settings menu program send states to UIManager class, and then this controller creates the required system. Inside this settings menu, users can select operations that s/he wants to do such as play game menu, create or join game menus. The sequence diagram of this case can be seen in the figure below (fig. 23).

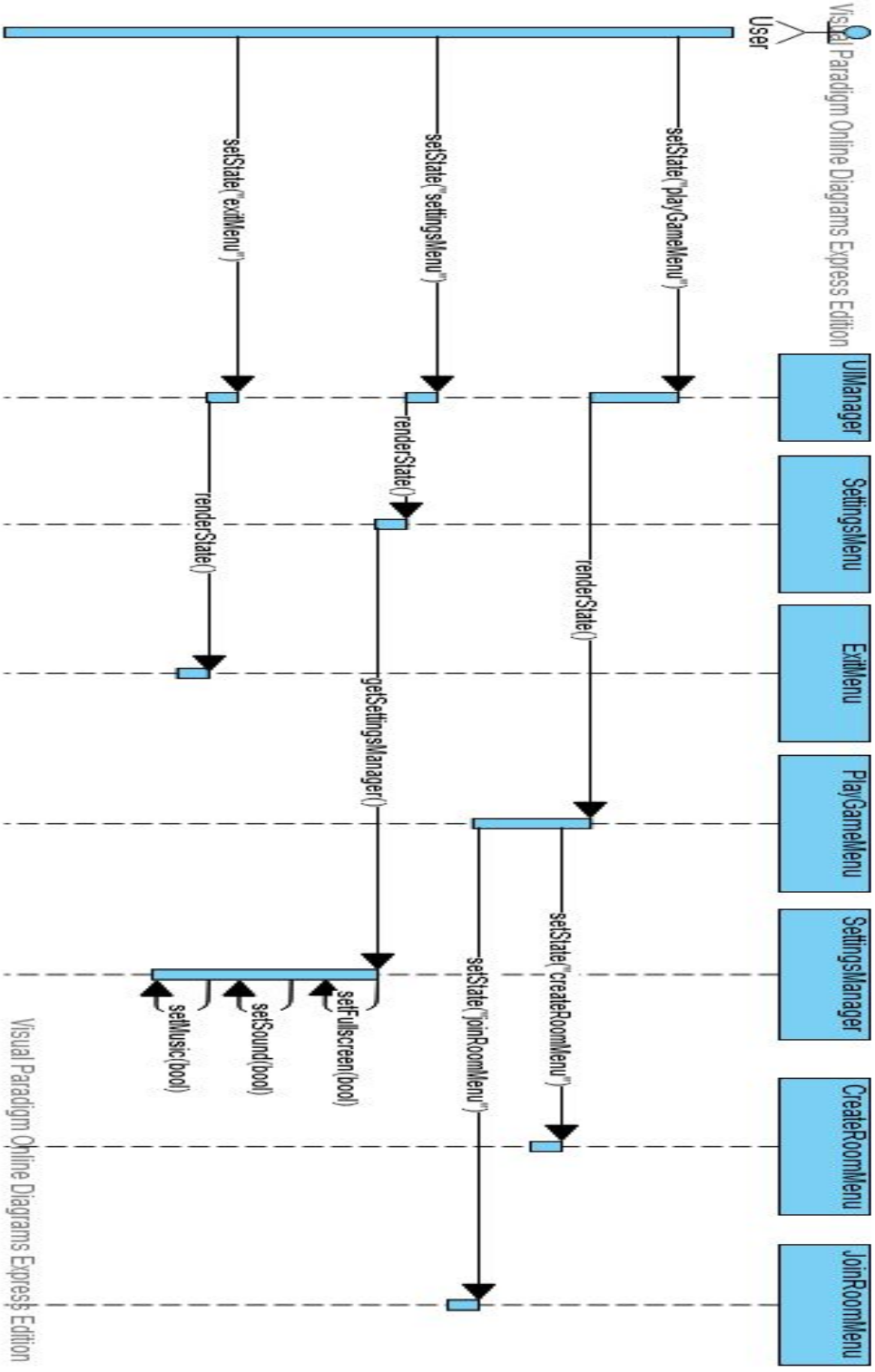

Figure 23: Sequence Diagram of Chat

Chat Sequence Diagram

In this diagram, a sequence diagram for a chat option that will be added to the project is given. Firstly, the user needs to have the room id that is used for joining this room. After taking this room id, the user needs to join this room. While in this room, users can send messages to the chatting class and receive messages from them. This process can continue as long as s/he does not leave the messaging room. The sequence diagram of this case can be seen in the figure below (fig. 24).

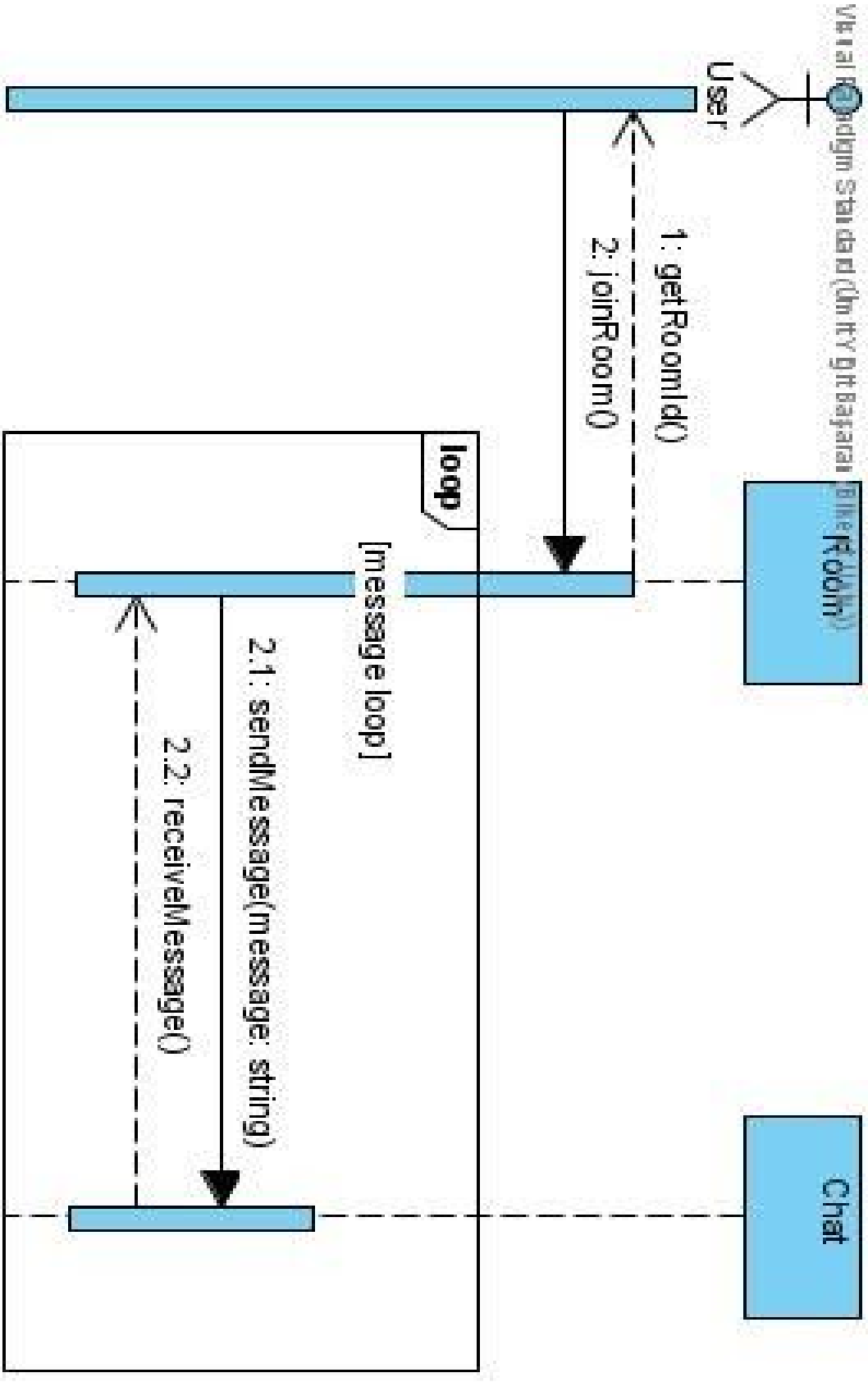

Figure 24: Sequence Diagram of Chat

Buying Property Sequence Diagram

In this diagram, a sequence diagram which represents buying a property from the tile that the current player is on is given. Firstly, the user needs to get the availability information from the property with the isOwned function that we will define in our project. Depending on the availability of the property, the user can buy either a utility or city. After this buying operation the tile id, meaning, id that we can use to provide specific information about the property is sent to the Player class. From the player class with setUserProperties() function, specifications of the user are set. The sequence diagram of this case can be seen in the figure below (fig. 25).

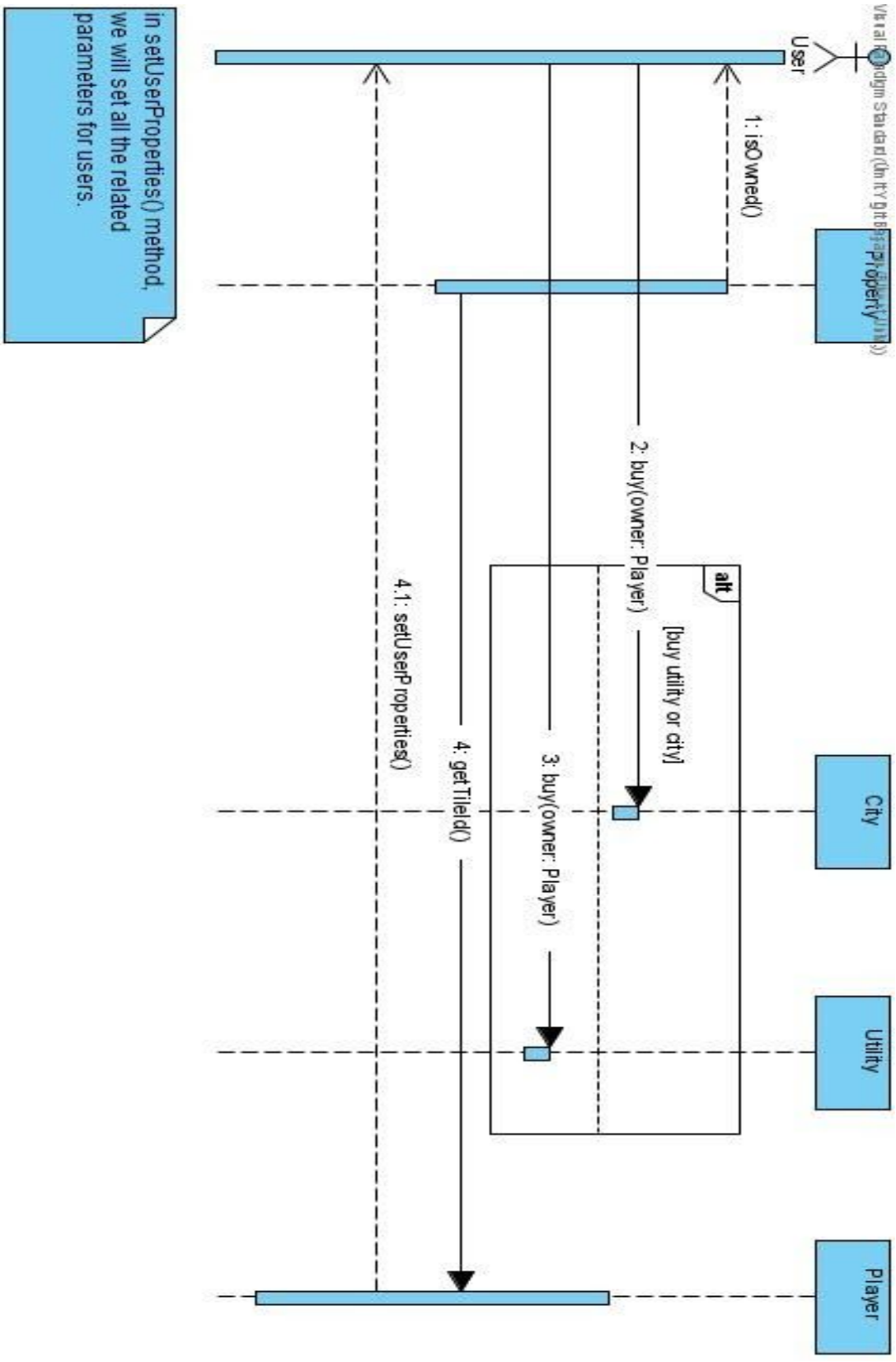

Figure 25: Sequence Diagram of Buying Property

Auction Sequence Diagram

In this diagram, a sequence diagram for auction operation is given. It represents the case that one player will activate it if s/he doesn't want to buy the property and wants to auction. Firstly, the user checks the status of the auction by using the isCurrentPlayerBoughtProperty() function. Then the system asks every player to either bid for this property or withdraws from an auction. After all, players are passed the system checks the bid status and returns the winner of this auction. Finally, the property is sold to the player who won the auction. The sequence diagram of this case can be seen in the figure below (fig. 26).

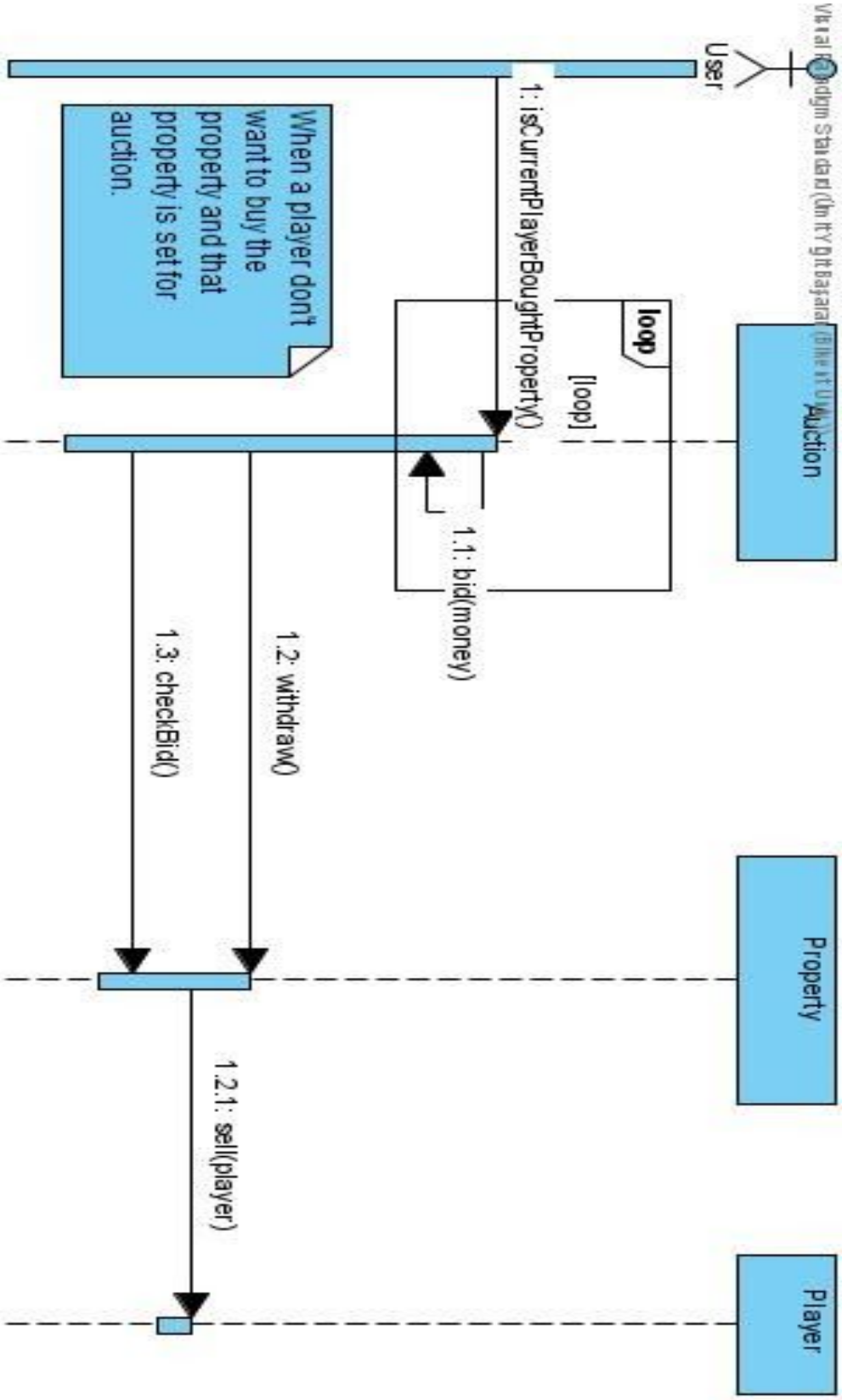

Figure 26: Sequence Diagram Of Auction

In-game Menu Sequence Diagram

In this diagram, a sequence diagram for the in-game menu of the project is given. Firstly, in order to open the in-game menu, the user needs to press the ESC button. In the beginning, the system keeps track of the ESC button listener. If the ESC button is pressed while in the game then the system goes to the UIManager and checks the state of the user. The system generates the menu when it is sure the player is still in the game. After that, if the user presses the “settings” button then it activates settingButtonListener. After adjusting settings from the settings menu, with the help of the setSettingsConfiguration() method, the adjustments are set. If the user wants to press the exit button, then exitButtonListener() is activated and the game will be closed for the player who uses this button. The sequence diagram of this case can be seen in the figure below (fig. 27).

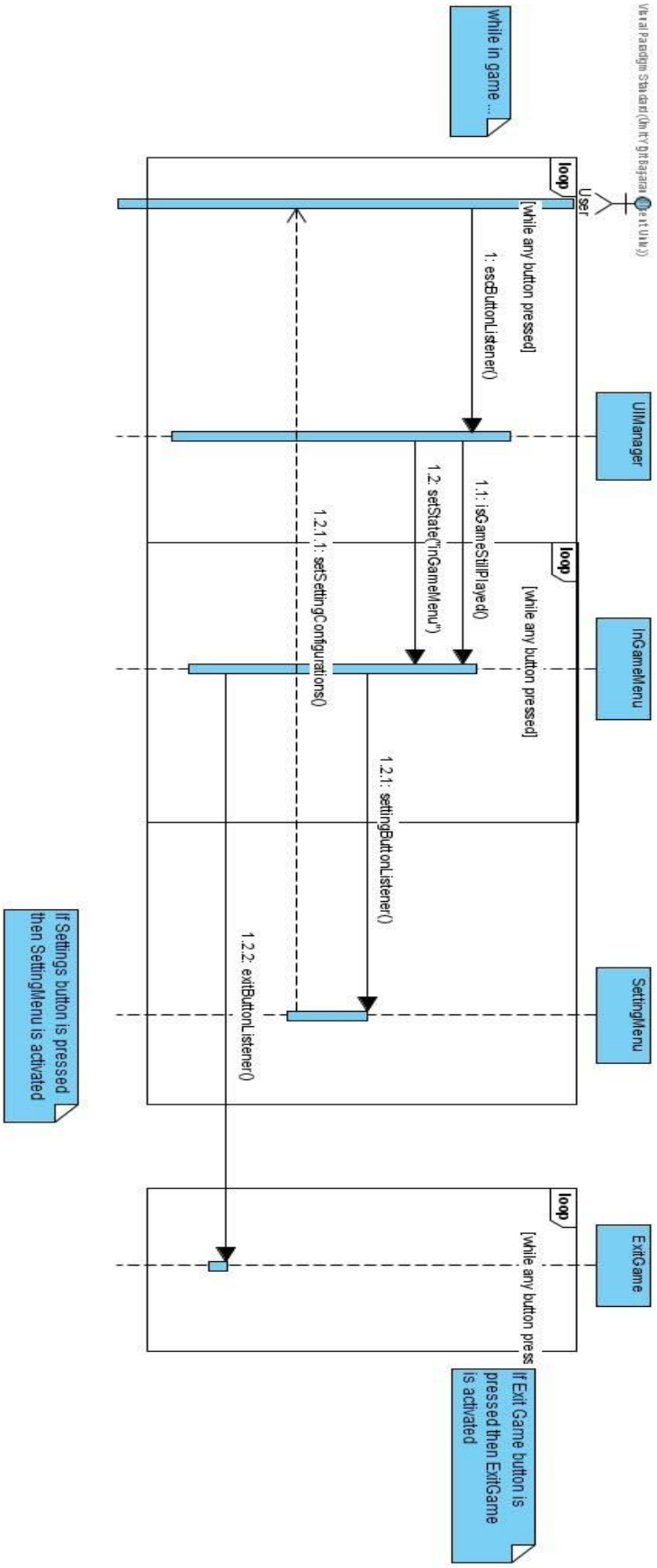

Figure 27: Sequence Diagram Of In-game Menu

Building Sequence Diagram

Below figure 28 represents the sequence diagram of the building process. If a player owns all cities of the same color, they can build houses or hostels in a city of their choice by paying the price. In our object class model, the house and hostels are properties of property class in order to see these properties on the board in the user interface. Firstly, in this case, the user must have this city so we check it with the isOwned() method in the User class. After that If s/he has, the building required conditions are checked in cityGroup class. These requirements are given in How to Play Part of the report. Moreover, if it is possible, users can build houses or hostels with particular game rules (see How to play Part). Therefore, if a user wants to build a house, there is a loop that s/he can build at most 4 houses, or an alternative way of it, s/he can build a hostel so these situations are given in the alternative part of the diagram. Finally, using the id of User, the budget is updated with setBudget() method.

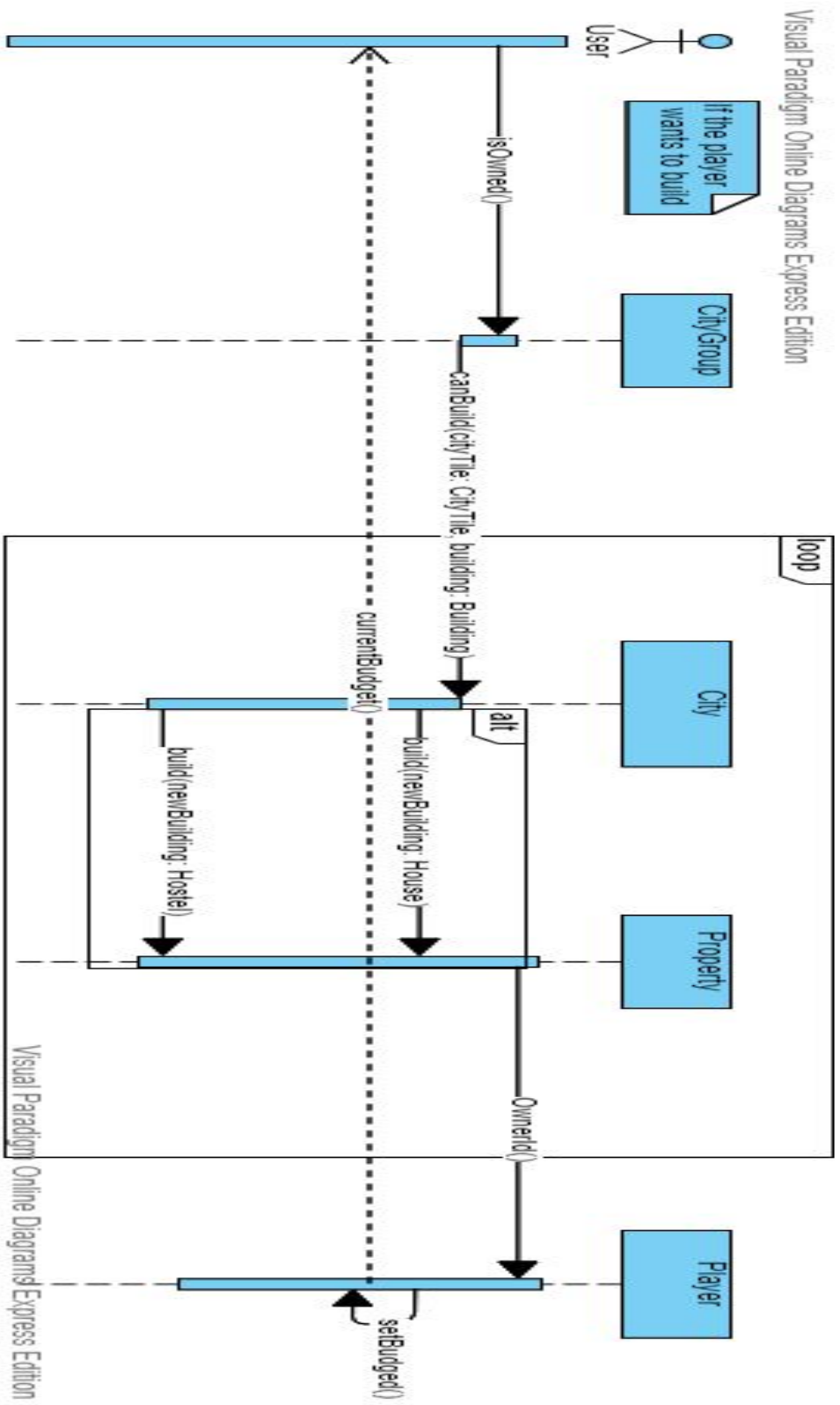

Figure 28: Sequence Diagram Of Building

Use Card Sequence Diagram

The below figure represents the sequence diagram for the use of card action. Use card action can be taken in two cases, where the user tries to use the chance or the community chest card that s/he holds or gets in their turn. In our object-class model, the ChanceCard and CommunityChestCard classes extend the Card class, in order to follow the inheritance principle of object-oriented programming. We also have the Tile class used in the diagram, which basically is the essential class of the board to hold all the properties, card squares, jail, and etc. The user, or the player, starts the action by rolling dice and moving to the tile according to the value on the dice. Then the Tile class checks whether the current tile is a ChanceCard tile or a CommunityChestCard tile, and takes the proceeding action accordingly. After determining the type of the tile, necessary action is taken by either one of the two classes given below. Finally, the task() method returns what the task written on the card is, which completes the sequence diagram. The mentioned figure can be seen below in Figure 29.

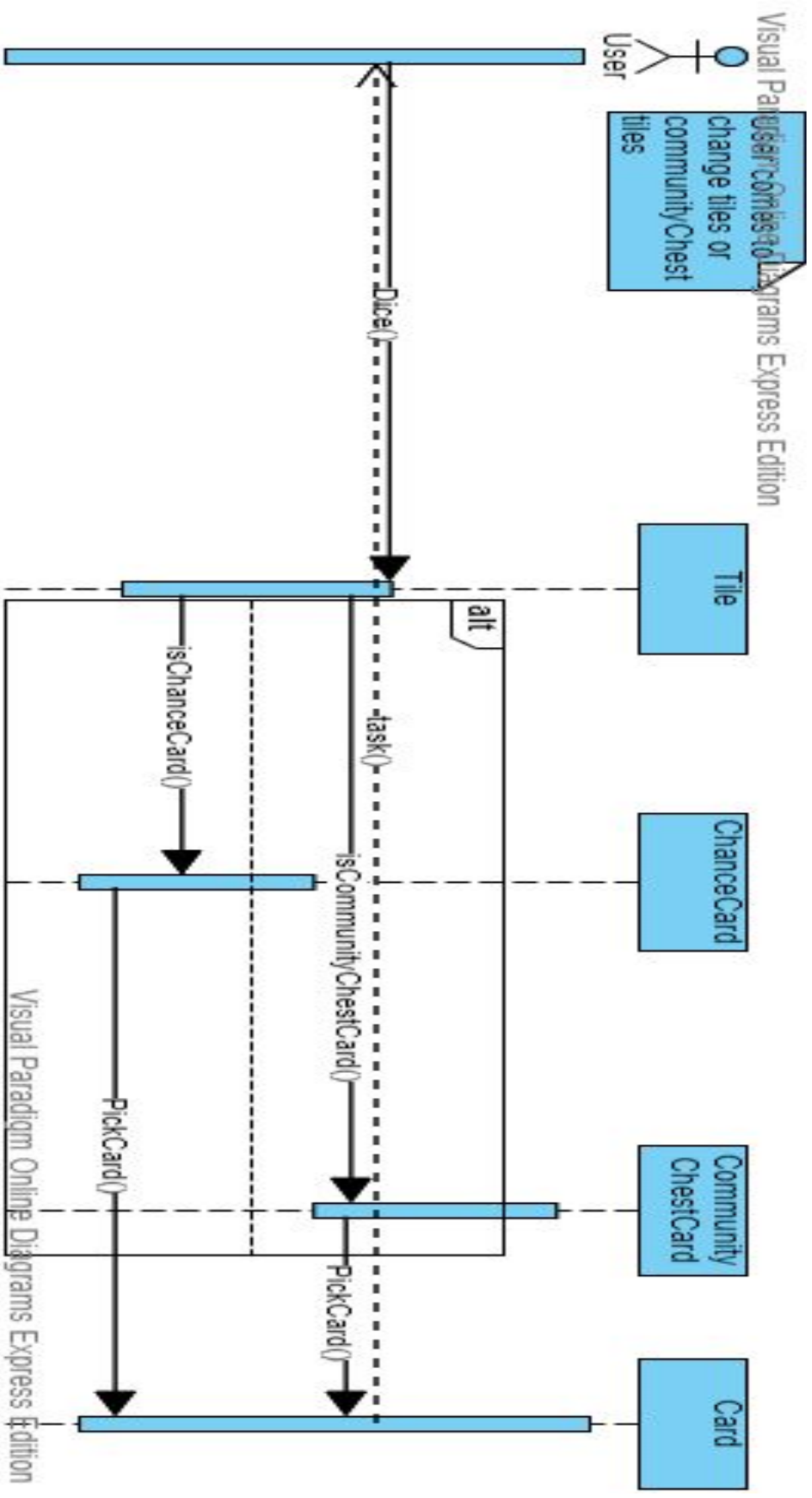

Figure 29: Sequence Diagram Of Use Card

State Machine Diagram

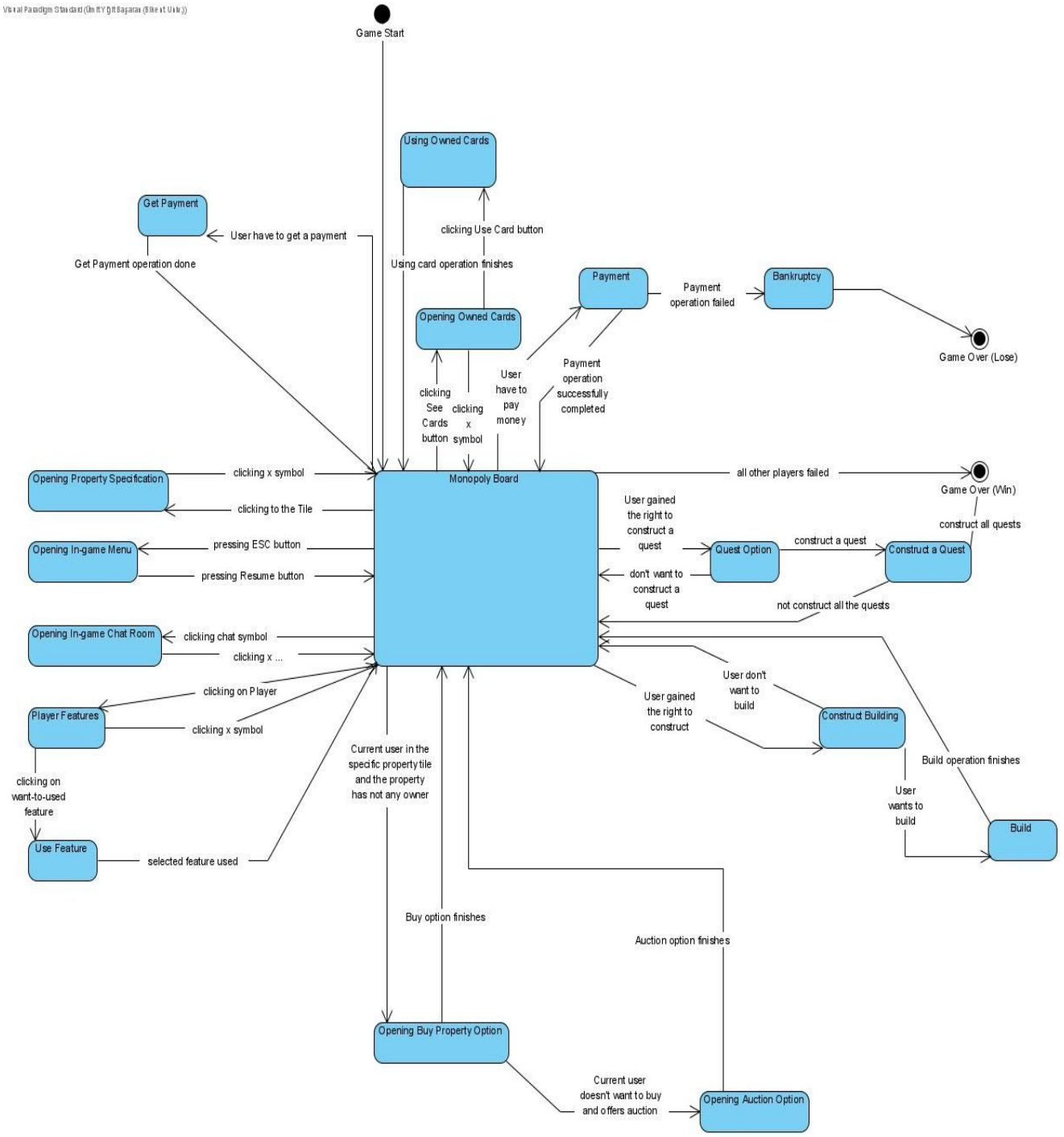

Figure 30: State Machine Diagram

In this State Machine Diagram, modeling the states of the Monopoly game while playing the game is given. Monopoly Board state is the state in which players play the game and where users can reach other features and states of the game. In opening property specifications, state users can open property specifications. In opening the in-game menu state by pressing the ESC button users can open the in-game menu of the game. In opening the in-game chat room state by clicking the chat symbol on the board user can open the chat room of the game. To switch the state to player feature state users can click their own players and see their specific features. From this state, users can switch the state to use feature state and use their players' specific powers. To switch state to opening buy property state users have to fulfill requirements defined in the diagram, after switching this state user can either buy or switch the state to auction state. Again in the game, if users can fulfill the requirements of constructing a building then the current state is changed to construct building state. In this state, users can either want to build and stitch the state to build properties or don't want to build and change the current state to the board state. Like constructing a building state, construct a quest state has the same idea, however, if the user can build all quests which are required to win the game, users can win the game before waiting for others' bankruptcy. In a payment state, because the main objective of the game is causing others to go bankrupt, the game is designed with payments. If one user can not pay the payment then s/he loses the game. In opening owned cards state users can see their owned cards by clicking over them and in this state they can switch state to use card state to use them.

Activity Diagram

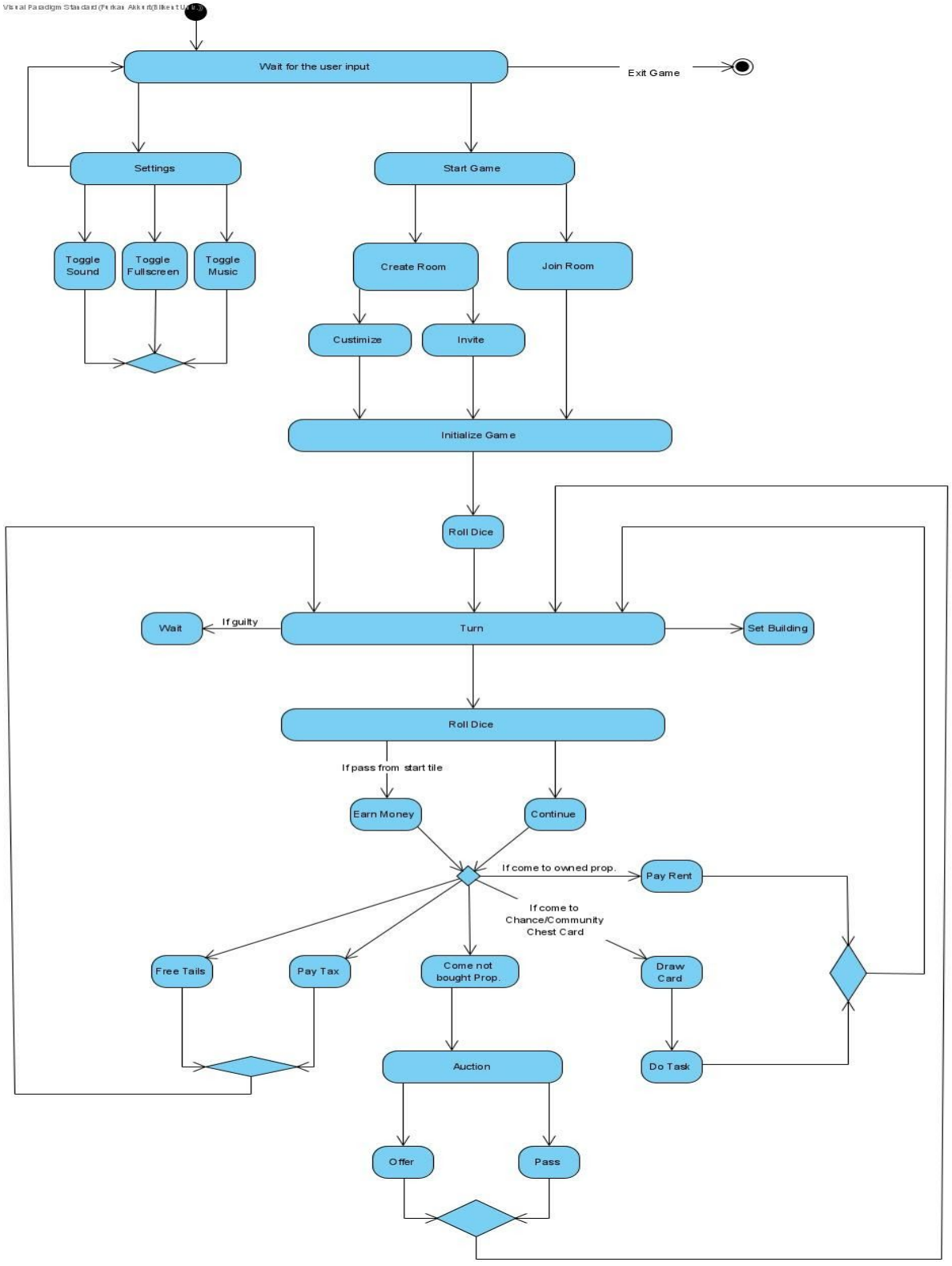

Figure 31: Activity Diagram

In the Activity Diagram, the flow of the game is shown by representing the actions as rectangles, and the relationship between the actions by arrows. The game is opened by the user as the first action, in which the main menu is shown to the user. At that point, there are two actions that the user can select to perform, starting a game, or going into the settings menu. If a user decides to go with the settings, s/he can adjust the setting according to their preferences and can go back. Else, s/he can start a game by creating a new room or joining an existing room. While creating the game, the user can also invite other players and customize the game. After completing these steps, the Join Room and Create Room actions merge together in the Initialize Game action. After that point, the diagram represents how the in-game actions flow. After rolling dice, the player moves, and the first if condition to be checked is whether the player passes through the starting point or not. If a player passes through the start tile, s/he earns money, else s/he does not. Then there are several if conditions, in which the next action for the current player is determined. For example, if the player ends their move at a tile that is owned by another player, s/he has to pay rent to that user. Or, if the player is at one of the card tiles at the end of the move, the Draw Card and Do Task actions are performed respectively. After a player completes their turn, the actions are merged and returned back to the Turn action, where the steps are repeated until the end of the game.

UI Mockups

Figure 32: Main Menu

This is the expected template for the main menu of the game. There is the logo of the game at the top right corner, as well as three buttons for starting a game, adjusting the settings, and exit.

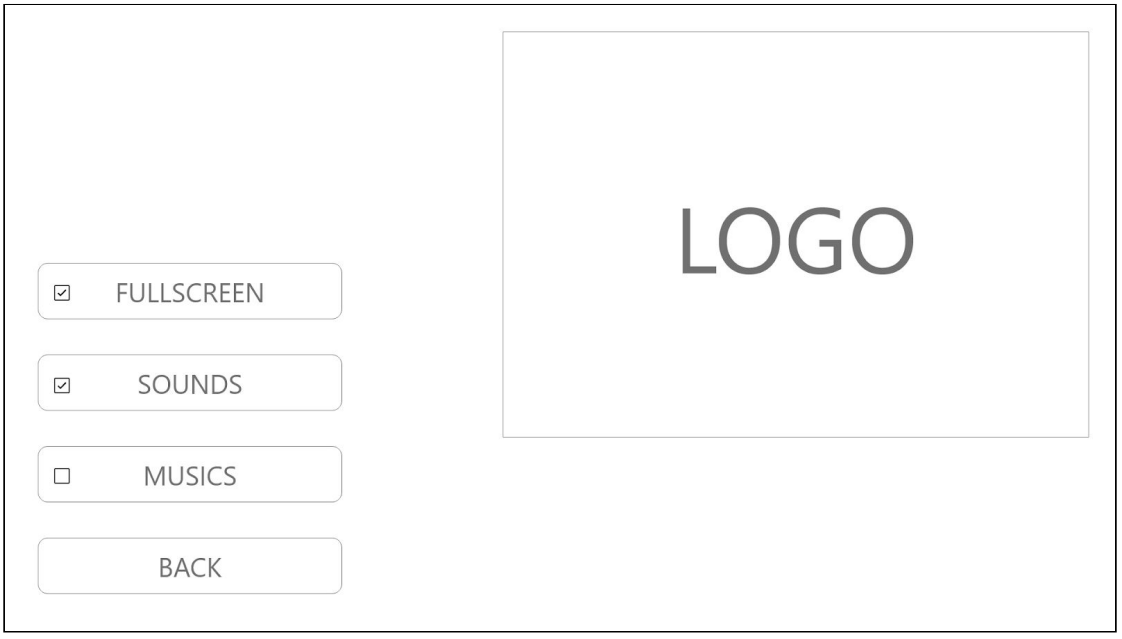

Figure 33: Options

The template above is to be used for the settings menu. Users can adjust sound, music, screen resolution, and can go back to the main menu by using one of the four buttons.

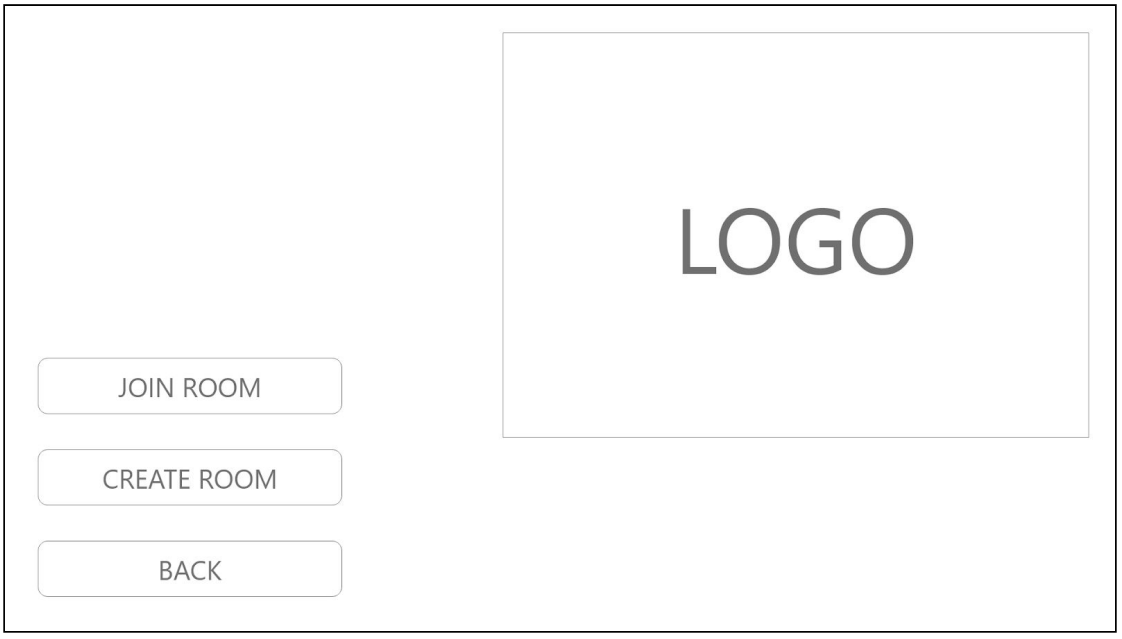

Figure 34: Start the game menu

The screen above is shown to the user when s/he clicks on the start game button on the main menu. Here the player can create a new room or can join an existing room by selecting the corresponding option. Also, the back button can be used to go back to the main menu.

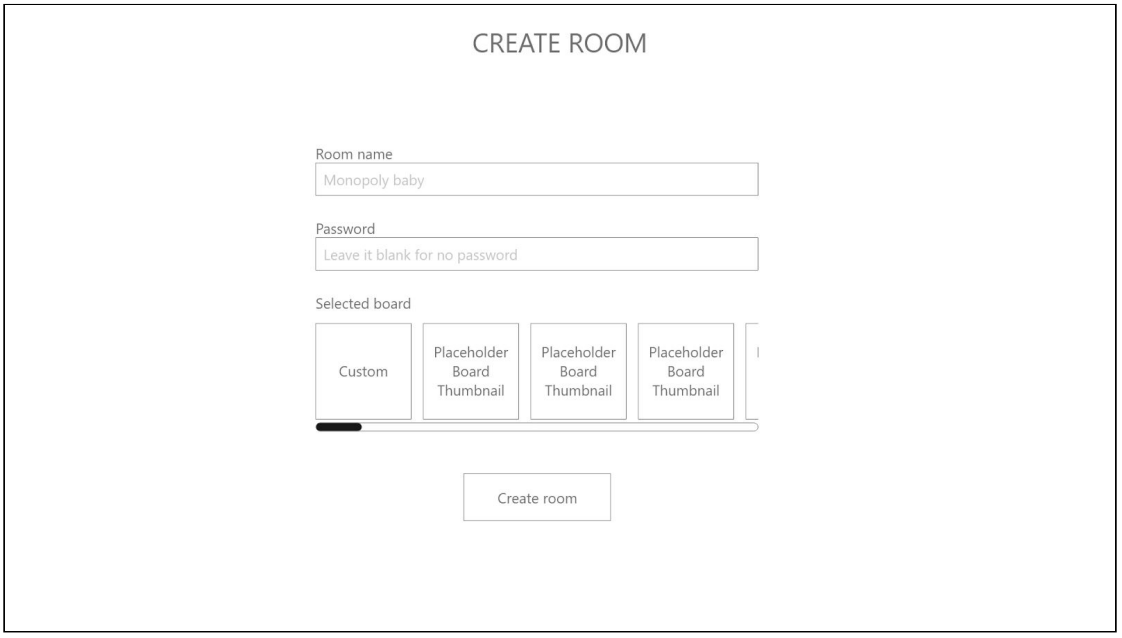

Figure 35: Create Room screen

The template above is used to create a new room. It consists of two text areas, in which the user enters the name of the room and the password for the room. In addition to that, users can select from the predefined templates while creating the room. The Create Room button below completes the process and creates the room according to the prompted settings from the user.

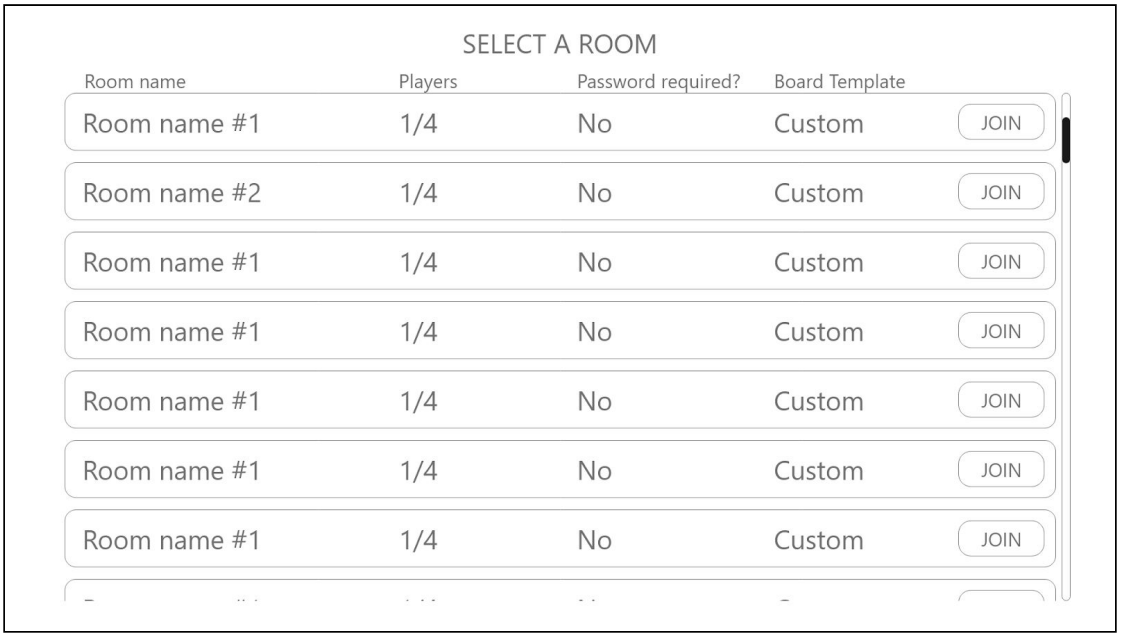

Figure 36: Join Room screen

While joining an existing room, users can select the room they want to join. From the menu, users can see the current situation of the rooms, such as the name of the rooms, how many players are in the room, and etc. After deciding on which room to join, the user can click on the Join button and can join the room.

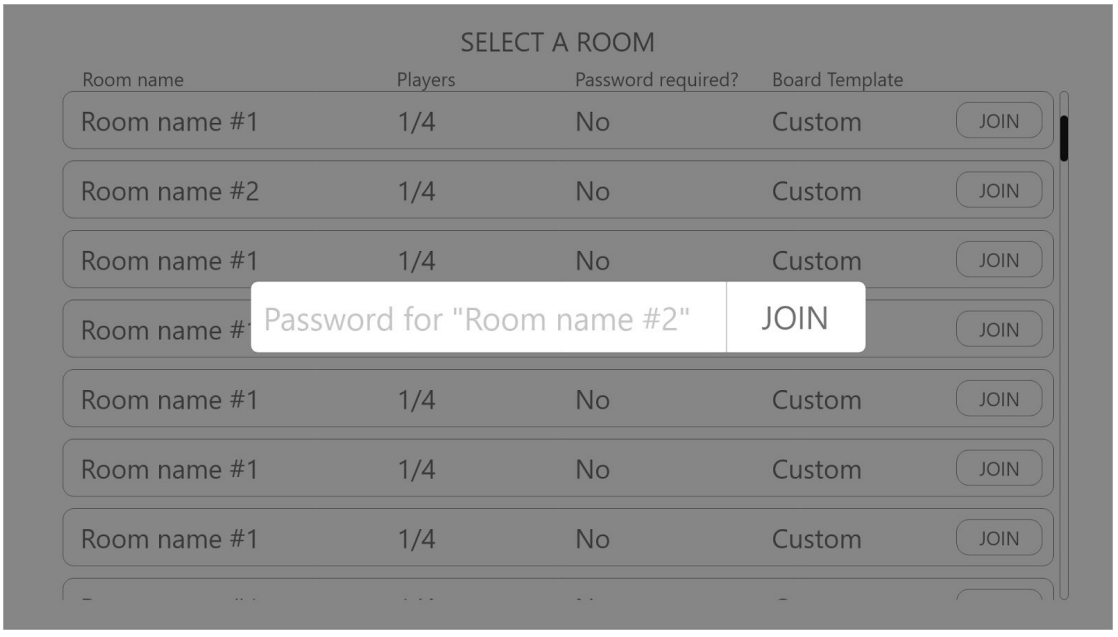

Figure 37: Ask password when joining a room

After pressing the join button, the player must enter the password of the room that would play.

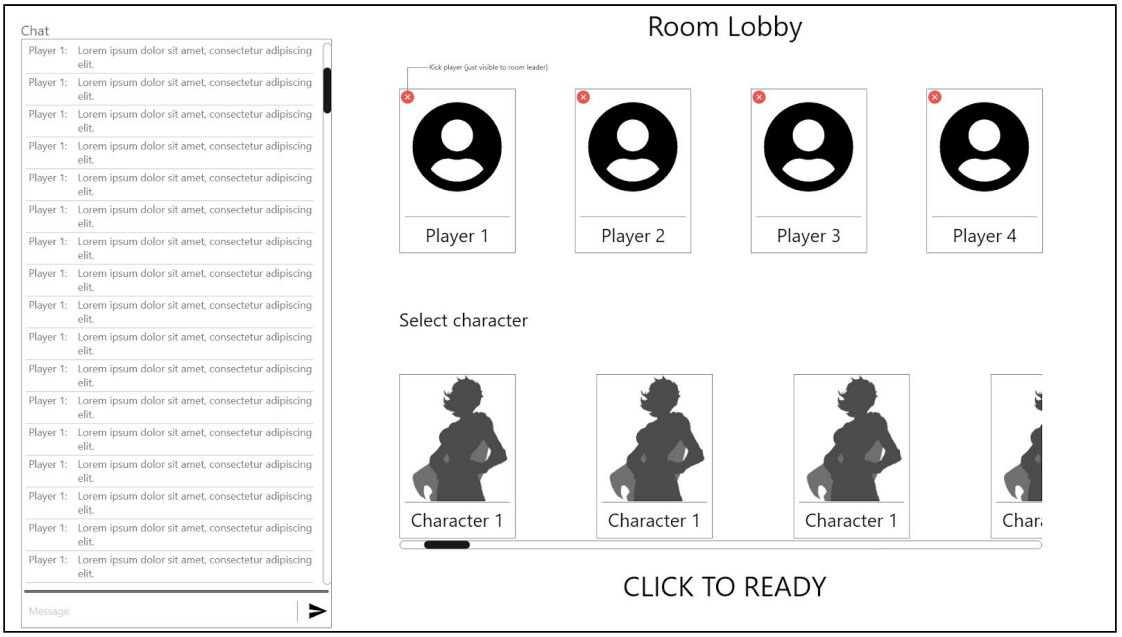

Figure 38: Room lobby screen

In the lobby screen, players can select their characters that have special skills. On the left side of the screen, players can chat with each other. After all, players state that they are ready, the game will initialize.

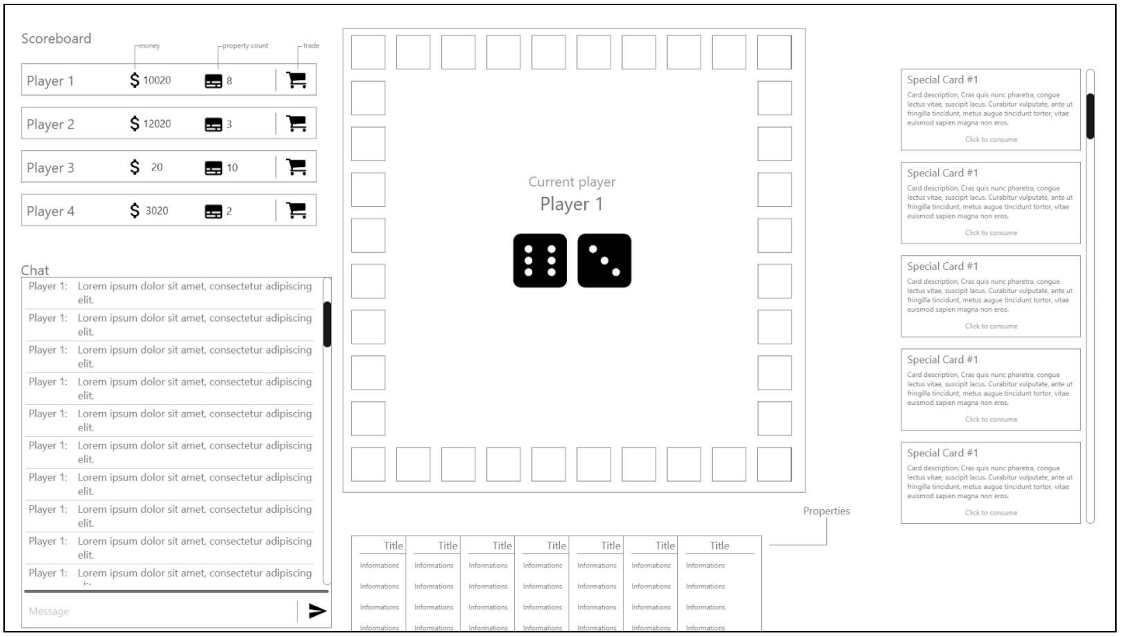

Figure 39: Game screen

This is the Game Screen mockup. The bottom part of the screen represents the cards of the user, the left side is a scoreboard and chat, the right side is the special cards the user earned. Players who are waiting for their turns would see this screen. In the middle of the board, 2 dices and a Current Player are shown.

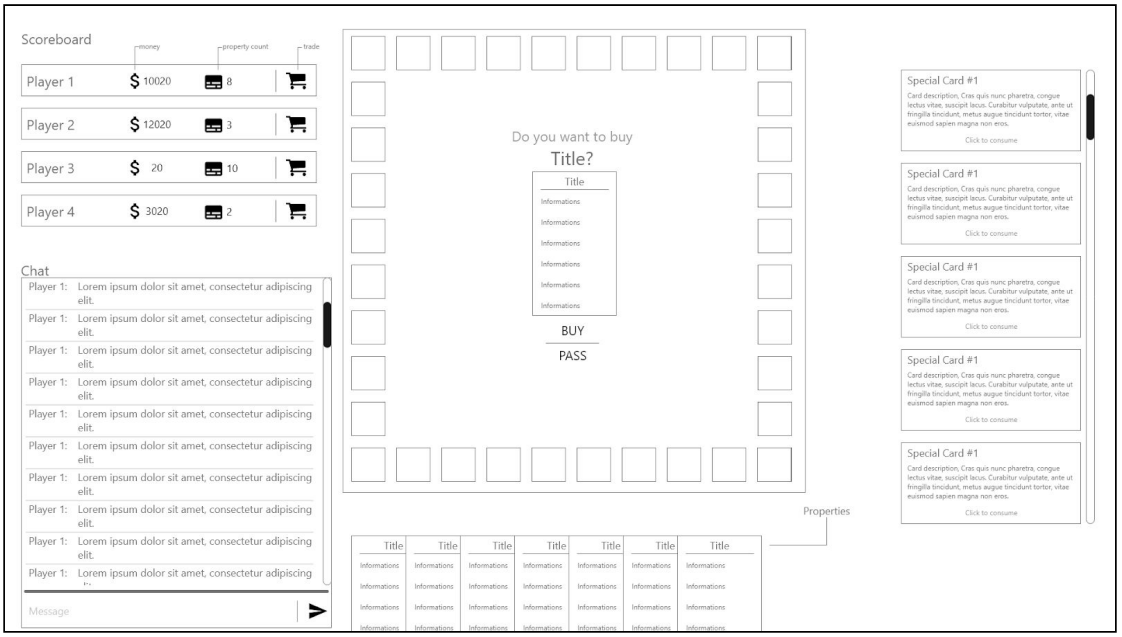

Figure 40: Buying a property

This is buying a property mockup. The bottom part of the screen represents the cards of the user, the left side is a scoreboard and chat, the right side is the special cards the user earned. In the middle of the board, there is a pop-up and asks “Do you want to buy tile?” and the user can select BUY or PASS.

References

[1] (n.d.). Retrieved October 28, 2020, from https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-state-machine-diagram/

[2] (n.d.). Retrieved October 29, 2020, from https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/

[3] Bruegge, B. (2014). *Object-oriented software engineering using UML, Patterns, and Java*. Pearson.