



# CS 319 Term Project

## Analysis Report

*Bilpoly (Monopoly Bilkent)*

Section 3

Group 3H

### Project Group Members

- |                    |          |
|--------------------|----------|
| 1. Ömer Ünlüsoy    | 21702136 |
| 2. Arda Akça Büyük | 21802835 |
| 3. İrem Tekin      | 21803267 |
| 4. Ece Ünal        | 21703149 |
| 5. Furkan Ahi      | 21501903 |

## Table of Contents

<b>1.</b>	<b><i>Introduction</i></b>	<b>4</b>
<b>2.</b>	<b><i>Overview</i></b>	<b>4</b>
<b>2.1.</b>	<b>Gameplay</b>	<b>4</b>
<b>2.2.</b>	<b>Game Board</b>	<b>5</b>
<b>2.3.</b>	<b>Player's Initial Items</b>	<b>5</b>
<b>2.4.</b>	<b>Modes</b>	<b>5</b>
<b>2.4.1.</b>	Time Modes	6
<b>2.4.2.</b>	Game Modes	6
<b>2.5.</b>	<b>Pawns</b>	<b>7</b>
<b>2.6.</b>	<b>Credit Cards</b>	<b>7</b>
<b>2.7.</b>	<b>Properties</b>	<b>7</b>
<b>2.8.</b>	<b>Cafes, Lands, Functional Places</b>	<b>8</b>
<b>2.9.</b>	<b>Chance &amp; Rector's Whisper Cards</b>	<b>8</b>
<b>2.10.</b>	<b>Game rules including added and changed features</b>	<b>8</b>
<b>2.11.</b>	<b>Options</b>	<b>11</b>
<b>3.</b>	<b><i>Functional Requirements</i></b>	<b>12</b>
<b>3.1.</b>	<b>Play Game</b>	<b>12</b>
<b>3.1.1.</b>	Pre-game Settings	12
<b>3.1.2.</b>	Bilkent Buildings Board	13
<b>3.1.3.</b>	Bilkent CS Board	13
<b>3.2.</b>	<b>Options</b>	<b>14</b>
<b>3.3.</b>	<b>How to Play</b>	<b>14</b>
<b>3.4.</b>	<b>Credits</b>	<b>14</b>
<b>3.5.</b>	<b>Quit</b>	<b>14</b>
<b>4.</b>	<b><i>Functional Additional Requirements</i></b>	<b>15</b>
<b>4.1.</b>	<b>Chance and Rector's Whisper Cards</b>	<b>15</b>
<b>4.2.</b>	<b>Credit Card</b>	<b>15</b>
<b>4.3.</b>	<b>Mortgage</b>	<b>15</b>
<b>5.</b>	<b><i>Nonfunctional Requirements</i></b>	<b>15</b>
<b>5.1.</b>	<b>User Interface and Human Factors</b>	<b>15</b>
<b>5.2.</b>	<b>Game Content</b>	<b>16</b>
<b>5.3.</b>	<b>Usability</b>	<b>17</b>
<b>5.4.</b>	<b>Reliability</b>	<b>18</b>

<b>5.5.</b>	<b>Compatibility .....</b>	<b>18</b>
<b>5.6.</b>	<b>Extendibility.....</b>	<b>18</b>
5.6.1.	Player No and Game Boards.....	18
5.6.2.	Single Player Mode.....	19
5.6.3.	Chatroom .....	19
<b>6.</b>	<b><i>Nonfunctional Additional Requirements.....</i></b>	<b>19</b>
<b>6.1.</b>	<b>Performance.....</b>	<b>19</b>
<b>7.</b>	<b><i>System Models .....</i></b>	<b>20</b>
<b>7.1.</b>	<b>Use-Case Model .....</b>	<b>20</b>
<b>7.2.</b>	<b>Dynamic Models.....</b>	<b>25</b>
7.2.1.	Sequence Diagrams.....	25
7.2.2.	Activity Diagram.....	31
7.2.3.	State Diagram .....	33
<b>7.3.</b>	<b>Object and Class Model .....</b>	<b>36</b>
7.3.1.	MainMenu.....	37
7.3.2.	Options.....	37
7.3.3.	Credits.....	37
7.3.4.	HowToPlay .....	37
7.3.5.	PlayerSelection.....	37
7.3.6.	PreGameSettings .....	38
7.3.7.	SoundManager .....	38
7.3.8.	PauseMenu .....	38
7.3.9.	AssetManager.....	38
7.3.10.	GameManager.....	39
7.3.11.	Player.....	39
7.3.12.	ChanceCard.....	39
7.3.13.	RectorsWhisperCard .....	39
7.3.14.	Pawn .....	40
7.3.15.	Location.....	40
7.3.16.	Landable.....	40
7.3.17.	Land .....	40
7.3.18.	CardPlace .....	40
7.3.19.	Cafe.....	41
7.3.20.	AtalarsRoom.....	41
7.3.21.	FunctionalPlace.....	41
<b>7.4.</b>	<b>User Interface - Navigational Paths and Screen Mockups.....</b>	<b>42</b>
7.4.1.	Navigational Path .....	42
7.4.2.	Screen Mock-ups.....	43
<b>8.</b>	<b>Conclusion.....</b>	<b>63</b>
<b>9.</b>	<b>Improvement Summary .....</b>	<b>64</b>
<b>10.</b>	<b>Glossary.....</b>	<b>65</b>
<b>11.</b>	<b>References.....</b>	<b>66</b>

# **1. Introduction**

As a group 3H, we decided to design and implement the “Monopoly” game. Monopoly is an old real estate board game in which the players goals are to dominate the real estate market [1]. Players try to be the wealthiest “through buying, renting, and trading with the intention of bankrupting all opponents” [2]. The motivation behind choosing this game as our project was due to the fact that we have played this game since we were children. In addition, Monopoly can be implemented using object-oriented programming which we want to learn more about this semester.

It is normally played by 2-8 players and it has different versions. We will take the last version of the original monopoly [2] and change some of its features, also add new rules and features. A new name was given to the game as “Bilpoly” and it will take place in Bilkent neighborhood or in a game board consists of Bilkent CS courses. Current game will have the following features which will be explained further in the report:

- Two game board modes: Bilkent Buildings Mode and Bilkent CS Mode
- Normal Mode and Timed Mode
- Sound Options

The game will be implemented using Java and JavaFX with object-oriented programming practices.

# **2. Overview**

## **2.1. Gameplay**

Bilpoly is a computer game that is an adaptation board game. It can be played by two to four players on the same computer. Players are represented by pawns [see section 2.5] and their money is represented by credit cards [see section 2.6]. There are two game board modes [see section 2.4.2] which are Bilkent Buildings Mode and Bilkent CS Mode. Bilkent Buildings Mode is the priority while building the game and further explanations about the game will

be based on this mode unless it is stated as Bilkent CS Mode rules. Also, there are two game modes according to time which have different ending conditions [see section 2.4.1]. However, the common aim of each time mode is to dominate the whole Bilkent by opening Bilkas and Starbucks to places and pushing the other players to go bankrupt (Starbucks and Bilka are equivalent to house and hotel in classic Monopoly respectively). In Bilkent CS Mode, the game finishes when all players except one are out of “hours” which is their balance, or if the game is in the Timed Mode when the time is out the player that has the most “hours” wins. The game is played with turns and each player rolls dice when it's their turn. The sum of the dice's outcomes is the move count of the player. Moves are done by the player's pawn. Player takes action according to the place which his/her pawn lands. Places and actions according to them will be explained further in the report.

## **2.2. Game Board**

On the game board there are lands, Chance and Rector's Whisper places, Atalar's Room (jail), Go to Atalar's Room Place, Nizamiye (starting point), Free Parking, Dormitory Fee Place, Tuition Fee Place, Book Fee Place, Food Fee Place, and Cafes. In addition, on the center of the game board there are Chance and Rector's Whisper Cards.

## **2.3. Player's Initial Items**

At the beginning of the game, players will enter a nickname and select a pawn. Furthermore, each player will have an initial money which can be determined before starting the game.

## **2.4. Modes**

Before starting the game, time and board modes must be selected by the players. Selected modes cannot be changed once the game starts. To play with different modes players must stop the game and start a new game.

#### **2.4.1. Time Modes**

There are two modes in Bilpoly which are Normal Mode and Timed Mode. The mode selection is done before starting the game. In Normal Mode people get bankrupt and the last player left in the game is the winner. In Timed Mode, players choose the game duration before starting the game and when time's up the player who has the most money wins the game.

#### **2.4.2. Game Modes**

There are two game board modes in Bilpoly which are Bilkent Buildings Mode and Bilkent CS Mode. The mode selection is done before starting the game. In the first mode, the lands on the game board are real places that belong to the Bilkent neighborhood such as buildings (like EA, EE, B, FF buildings...), dormitories (like 76th Dormitory, 81th Dormitory...) and iconic places (like Cafeteria, Mayfest, Library...) that are located in Bilkent University. Besides these, some cafes and restaurants in Ankuva and Bilkent Center (like Piel Roja, Bluejay...) are the cafes of the game. Buildings (lands) are in groups and every group has its own color. For example, B Building, SA Building and SB Building are in a group and all of them are represented with green.

In the second mode every land is a course that is taken by CS students in Bilkent University. Same as the Bilkent Buildings Mode, courses are grouped, and each group has its own color. Instead of money, there are "hours" in this mode as the currency. "Hours" is spent or gained in the game. When a player's pawn lands on a course, s/he could take the course by spending his/her "hours". In this mode, assignments are equivalent to Starbuckses and A+ is equivalent to Bilka. Players have to spend their "hours" to help the player that takes the course when their pawn is on that course. Mortgaging is equivalent to dropping the course.

## **2.5. Pawns**

Pawns are the representation of the players. Every player has to select a pawn before starting the game and cannot change their pawns afterward.

Players move on the game board according to the dice outcomes and these movements are done with pawns. Pawns will be moved automatically by the game after a player rolls the dice, no external action input is needed to move the pawns manually. In Bilpoly the pawns are:

- TMD (Tunus Ring)
- Ferrari
- BMW
- TAXI

## **2.6. Credit Cards**

In Bilpoly, credit cards are used to represent players' money. Before starting the game, the initial money for each player is chosen by the players, and the amount is shown on the credit cards. Each player has its own credit card, credit card colors are default and cannot be changed. The amount of money a player has is written on the card and will change during the game when money is spent or earned. When clicked on a credit card, the player's information is shown which includes the picture of the card and the lands, Starbucks, and Bilkas the player owns.

## **2.7. Properties**

In Bilpoly, Starbucks and Bilkas are the properties that can be bought and built when all the lands (buildings in Bilkent) on a color set are owned.

Properties of different lands have various costs. Also, the rent of a land changes when a property is built on it. Properties can be sold when the owner doesn't have enough money to execute a necessary action such as paying rent.

## **2.8. Cafes, Lands, Functional Places**

There are three main place groups on the game board.

### **Cafes are**

- equivalent to Railroad places in the US Monopoly Standard American Edition.

### **Functional places are**

- “Fee Places” which are “Tuition Fee”, “Dormitory Fee”, “Book Fee”, “Food Fee”. Fee places replaced the taxes, electric company and water works of the US Monopoly Standard American Edition.
- “Nizamiye”, “Go to Atalar’s Room place”, “Atalar’s Room”, “Free Parking”.

## **2.9. Chance & Rector’s Whisper Cards**

On the game board, there are places which make a person draw a card.

There are two types of cards: Chance and Rector’s Whisper. Each card has an action that the player who draws it does. An action can make the player pay money, earn money, move his/her pawn, etc. Chance Cards will randomize the numbers in the card as keeping it in limits such as 3000-20000 for money. Rectors Whisper Cards, the player does the instruction that is written on the cards that make the player pay or earn money, however, the amounts vary according to the current balance of the player.

## **2.10. Game rules including added and changed features**

The rules are given for Bilkent Buildings Mode which is our priority for implementation. CS Mode rules are similar to these except few name changes.

- + Credit cards will be used instead of paper money in the classic Monopoly.

- + Each player is given the same initial money. Currency unit is Bilcoin. Initial money of each player will be determined by the players at the beginning of the game.
- + There isn't a role for a banker, banker's responsibilities are done by the program.
- + The first player to start the game will be determined randomly by the game.
- + Player turn passes to the next player when a player finishes their operation (the turns of the players are predetermined at the initialization of the game by a shuffle operation to the players).
- + In a player's turn, the player rolls the dice, and according to the place where her/his pawn has landed the following can happen:
  - ✓ The player can buy land if the land doesn't have an owner or if the land is mortgaged and the owner who mortgaged the land is willing to sell it.
  - ✓ The player can build property if the player is on a color set that all the lands on it were bought by that player.
  - ✓ The player can pay rent if the land has an owner other than that player.
  - ✓ The player can pay a fee when his/her pawn lands on a Tax Place.
  - ✓ The player can draw Chance or Rector's Whisper card.
  - ✓ The player can go to Atalar's Room
  - ✓ The player can get money if the pawn lands on Free Parking.
  - ✓ The player can do nothing.
- + As a new feature, we will have the rule known as the "Free Parking Rule" to our game. In this rule, all money paid by the players to the bank (excluding the land sales) goes to the middle area. While the game continues, whoever comes to the "Free Parking" point gets all the money saved in the middle.

- + When a player builds three Starbuckses in a place, s/he will be allowed to open a Bilka in that place.
  
- + When a player buys all the land of a color set, s/he can start building properties on them. When the player's pawn lands on a land that player owns and in a completed color set, the player can build properties on any of the lands in the color set. The following rule in the original game "Building must be equal on all properties in a group. You may place a single building on a single property, but you may not place two buildings on one property unless all other properties in the group have one building present (even build rule)" [4] doesn't apply. Instead, the player can build properties (max 3 Starbuckses and 1 Bilka) as s/he wishes and can afford.
  
- + If a player doesn't have enough money, player can get money in the following ways:
  - ✓ Can sell the properties on his/her lands for half price.
  - ✓ Can mortgage a land if there is no property on it.
  - ✓ Can mortgage cafes
  
- + When a player wants to mortgage a land s/he has, properties on the land must be sold to the bank then the mortgage process starts.
  
- + When a land is mortgaged, the player gets money from the bank (half price of the land). If a player has enough money to lift the mortgage s/he pays an amount of mortgage price + mortgage price \* mortgage multiplier (in our case, mortgage multiplier is 10%).
  
- + If a land is mortgaged and another player's pawn lands on it, the game will ask the player if s/he wants to buy the land and then ask the owner if s/he wants to sell the land. If both sides agree, the mortgaged land can be sold.

- + If a land is bought by a player, other players cannot buy the land unless the owner mortgaged the land and agrees to sell it.
  
- + Rolling double dice doesn't give a player privilege unless the player is in Atalar's Room. If a player rolls double dice, his/her pawn moves accordingly, and the turn goes to the next player.
  
- + When it is the player's turn who is in Atalar's Room, the player rolls the dice. If the player can roll double dice, s/he is out of Atalar's Room, else the player waits for his/her next turn. If the player cannot roll double dice at the end of the third turn, s/he has to pay the Atalar's Room fee to get out of Atalar's Room. If the player doesn't have enough money to pay the fee, the player goes into the Mortgage Process and if still cannot afford the fee, go bankrupt and get out of the game.
  
- + Cafes can be bought by players, but players cannot build properties on them.
  
- + Cafes, lands and lands with properties have specific rents.
  
- + In the normal mode game finishes when everyone except one player goes to bankruptcy. The last player is the winner.
  
- + In the timed mode game finishes when the preset time ends, and the winner is the one with the highest score. Score calculation is done considering money, lands and property that players have.

## **2.11. Options**

Bilpoly has an Options screen which includes a music selection unit, music volume, sound effects volume, game background selection unit. The players

can select the default music provided by the game or can upload another music. Gamers enjoy listening to music while playing [3], because of this the selected music plays during the game. Music selection can be done at the beginning of the game. Sound effects and music volume can be adjusted during or before the game. Sound effects include money sounds when money is earned or lost, dice roll sound, click sounds, etc. Game background is a picture which is in the back of the game board and can be selected from the given options or can be uploaded by the players at the beginning of the game.

### **3. Functional Requirements**

#### **3.1. Play Game**

When a user wants to play the game, s/he should be able to encounter the pre-game settings. On the screen, the user is asked to make customization for all players and the game.

##### **3.1.1. Pre-game Settings**

In "Pre-Game Settings" users should be presented with 3 customizations to enter:

- Number of players must be selected,
  - Name of players must be entered and
  - The pawns of players must be chosen.
- 
- Player boxes should be able to open according to the number of players.
  - Each player must have a unique color and
  - Players should not be able to choose the same pawns.

Depending on previous choices of the "Pre-game settings", users must be able to make settings for the game and the board. There should be 3 selectable/adjustable features:

- Initial money must be selected
- Board type must be selected (Bilkent Buildings Mode or Bilkent CS Mode)
- Game mode must be selected. An adjustable time scroll can appear according to the user's choice.

### **3.1.2. Bilkent Buildings Board**

This board should include Bilkent buildings instead of cities/districts, unlike the classic Monopoly.

In addition to the classic Monopoly rules, the extra rules specified in the report are valid:

- The currency in this mode is Bilcoin.
- Instead of building houses and hotels on the land, the player must be able to open Starbucks and Bilka to the buildings.
- The community chest cards have been changed to Rector's Whisper cards.
- The chance cards remain the same, but their contents have been changed according to the theme.
- If the timed game mode is selected, the player with the most property + cash at the end of the time, should be able to win the game.
- In normal mode, the game must continue until all players except one player are bankrupt.

### **3.1.3. Bilkent CS Board**

In this board, instead of the lands, there are going to be Bilkent CS courses. The rules in the report are applied here, too. The currency in this mode is

Hour. Houses and hotels have been changed to assignments and A+. Chance and Rector's Whisper cards have been changed to suit the theme of the board. Game mode rules are the same as for other boards.

### **3.2. Options**

There should be 3 adjustable sections for the Options menu:

- Music volume
- Sound FX volume
- Game Background

Also, the user should be able to add music and game backgrounds.

### **3.3. How to Play**

This section should contain some information for the user:

- Purpose of the game
- Some basic rules
- Free Parking Rule

### **3.4. Credits**

This section should include the names and pictures of the game developers.

### **3.5. Quit**

It must allow the user to exit the game.

## **4. Functional Additional Requirements**

### **4.1. Chance and Rector's Whisper Cards**

- If a player draws a chance card, the value in the card must be randomized. Thus, a more dynamic game can be offered to players.
- If a player draws a Rector's Whisper card, the value in the card should change in proportion to the player's instant balance. Therefore, the snowball effect can be avoided in the game and a more competitive environment can be provided.

### **4.2. Credit Card**

- The cash flow should be carried out directly on credit cards with player information.

### **4.3. Mortgage**

- Players must be able to mortgage their properties when their money is insufficient.
- In order to open the mortgaged land, a certain penalty payment must be made.

## **5. Nonfunctional Requirements**

### **5.1. User Interface and Human Factors**

- For the game board, vibrant and highly contrasted colors must be used to make the places more visible on the game board [5].
- The background images must be lighter to make the game board more visible and make the screen less tiring to look at [6].

- Also, with the game board, credit cards, stop button, roll dice button, next turn information, remaining time information and information pop-ups about the current turn, the number of items on the game screen must not exceed seven to make the gameplay less confusing.
- At the Main Menu screen, more vibrant shade of the most used color of the image background must be used for buttons.
- For the screen pop-ups, everything must be lightened except the window that popped up, to make the pop-up more visible and make the game screen less eye-straining.
- Button names and other clickable features must have labels on them since they must be self-explanatory.
- Only mouse must be used as the input device while playing the game, since using more devices will make the game harder to learn.
- Only time that any other device other than a mouse must be used is the process of initialization of the game, where users enter their names with a keyboard.

## 5.2. Game Content

- While developing this game, the priority should be to reveal our own ideas by changing and developing some of its features, similar to the classic monopoly.

- The game to be made should be more appropriate to the Bilkent theme, containing traces from the CS department, changing some rules and points of the game, in a situation where the user is more committed to the game and a more competitive environment is created.
- In addition to all these, the **usability** of the game should be increased with the sound, music and backgrounds added to the game and it should be made more user friendly.

### 5.3. Usability

One of the most time-consuming places in the game while playing Monopoly is the money flow between players or between bank and game. Although it feels like real money offline and provides an interactive environment, money exchange is not a very user-friendly and practical phenomenon for a digital Monopoly-like game we are working on.

- Therefore, game currency must be kept digitally, and all money transfers must be made automatically by the game system.

One of the subtitles that increase usability in games is how competitive this game provides. Many new features have been added to the game to bring this environment to a much higher level than Monopoly.

- All payments from players to the bank should be collected at a point called "Free-Parking" instead of going directly to the bank vault. The main reason this was done was that this point did not fit well into the competitive environment in the classic version and was not considered necessary by many players.
- The user must come to any point on that color set to be able to make Starbucks or Bilka.

## **5.4. Reliability**

While playing this game, the player / players must not intentionally or unknowingly make various mistakes.

- All money transactions must be made directly by the game.
- None of the players should be able to access to funds.
- “Loading the dice” trick must become impossible to get on the digital platform.

## **5.5. Compatibility**

Although the game was intended for the Bilkent community, the change in the theme did not change the gameplay and rules much.

- The developed game should be easily played by someone who has played Monopoly before.
- Players should be able to easily access the new rules developed within the game.
- Players should learn these rules and be able to easily synchronize with the game.

## **5.6. Extendibility**

### **5.6.1. Player No and Game Boards**

In the current version of the game, a maximum of 4 players can play and players play the game on a single computer. In addition, the game has 2 boards named Bilkent buildings and Bilkent CS Mode.

- The number of players can be easily increased to over 4.
- Players can play the game online from different devices at the same time.

- Numerous boards for all kinds of different themes can be added to the game.

### **5.6.2. Single Player Mode**

The game is currently designed to be played with multiple players. However, users may also want to play the game alone.

- A bot system can be developed for the game.
- This system can be supported with Artificial Intelligence technology.  
Thus, bots can be played in a human-like way. This provides a more pleasant experience to the user in the single player mode.

### **5.6.3. Chatroom**

In the current state of the game, communication between players is limited. A money transfer takes place only when players come to each other's buildings, and this happens automatically in the design.

- A real-time chat room can be integrated into the game for more interaction between players.
- This improvement will directly contribute to the addition of the bargain feature in the game.

## **6. Nonfunctional Additional Requirements**

### **6.1. Performance**

- Application response time must be less than 0.5 seconds.

# 7. System Models

## 7.1. Use-Case Model

Visual Designer Standard Edition (Windows/Mac)

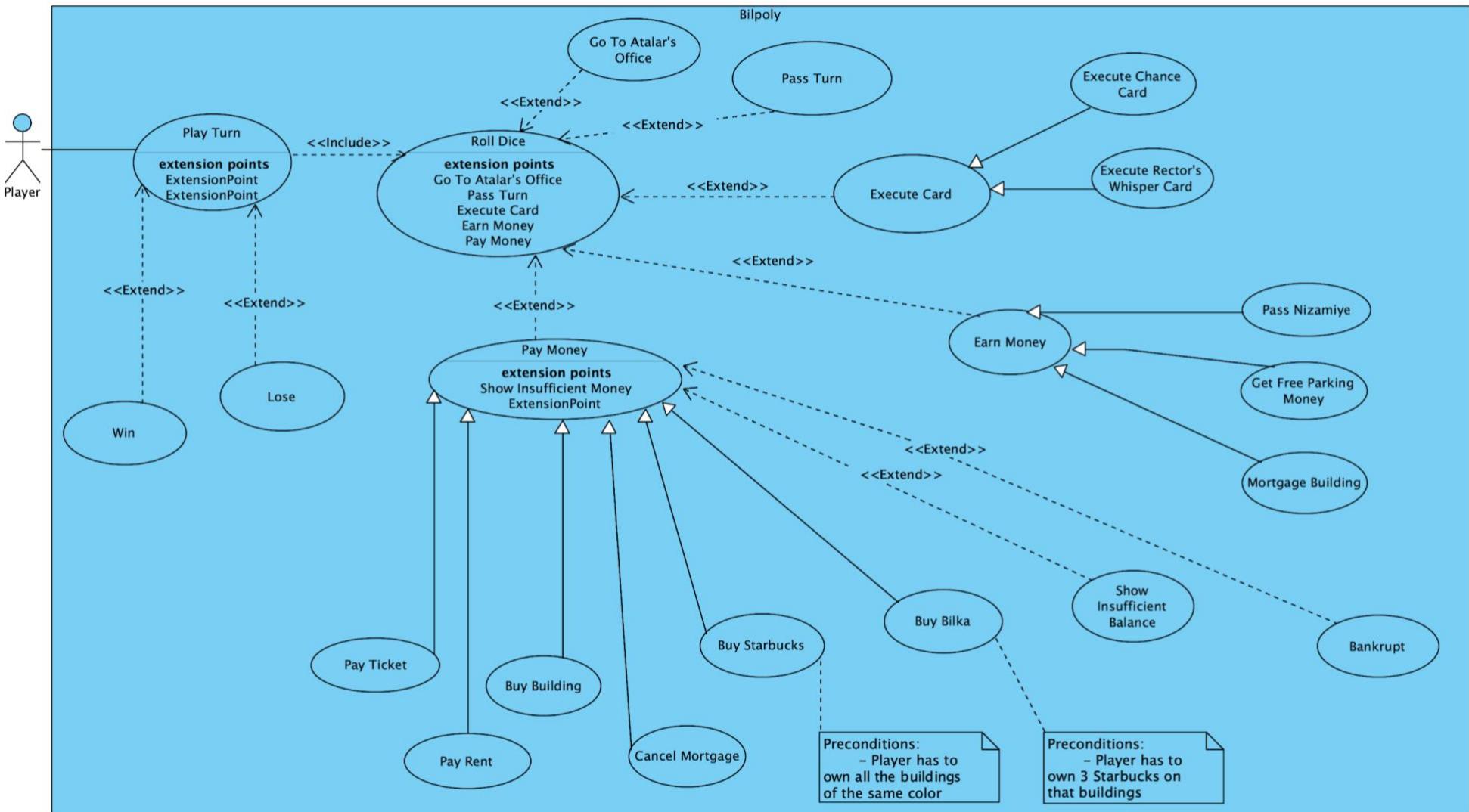


Figure 1. Use-Case Diagram

## **Use Case #1:**

### **Use Case: Players Starts the Game**

Primary Actor: Players

Stakeholders and Interests:

- Players want to play the game.

Pre-conditions:

- Players must be in the main menu.

Post-conditions:

- The game is on.

Entry-conditions:

- Players click the “Play Game” button on the main menu.
- Players adjust the setups of the game at two-screen Setup Menu.
- Players hit the “Play” button to start the game.

Exit conditions:

- Players click the “Pause” button during the game and the “Quit” button on the Pause Menu.
- One player wins the game by bankrupting the rest
- In the Timed Mode, the timer is over and the player in the best situation wins.

Main Event Flow:

- Players select the “Play Game” button on the Main Menu.
- Players select the “2 Players” option and type their names with their desired pawns.
- Players hit the “Next” button to view the second screen.

- Players select “\$100,000” initial money, “Buildings” Board mode, and “Normal” game mode.
- Players hit the “Start” button and initialize the game.
- Players play their turns in the order and one player goes bankrupt.
- Other player wins the game and the “Game Over” screen is displayed.
- Player hits the “Congratulations” button to go back to the Main Menu.
- Player quits the game by selecting the “Quit” button on the Main Menu.

Alternative Event Flows:

- During the game, players decide to end the game.
- One player hits the “Pause” button.
- Player selects the “Main Menu” option to end that game and go to the Main Menu.
- In the Main Menu, players start a new game or select “Quit” to quit the application.

## **Use Case #2**

### **Use Case: One Player Pays a Rent to Another**

Primary Actor: Player

Stakeholders and Interests:

- One player has to pay a rent to another.

Pre-conditions:

- Player’s pawn must be on a land owned by the other player.

Post-conditions:

- The player who paid the rent has less money.
- The owner of the land has more money.

Entry-conditions:

- Player rolls the dice and goes to the land owned by the other player.

Exit conditions:

- Player pays the rent and passes his/her turn.

Main Event Flow:

- In his/her turn, the player rolls the dice and moves the pawn.
- The pawn is on the land owned by the other player.
- Player pays the rent (Player's money decreases, other player's money increases, and an animation is shown).
- Player passes his/her turn.

Alternative Event Flows:

- Player takes a "Chance Card".
- Chance Card says player to go to the land owned by the other player.
- Player pays the rent (Players money decreases, other players money increases, and an animation is shown).
- Player passes his/her turn.

## Use Case #3

### Use Case: Player Buys a Bilka for a Land

Primary Actor: Players

Stakeholders and Interests:

- Player want to buy a Bilka for his/her land.

Pre-conditions:

- Player must be on a land of the same color set.
- Player has to have the all color set.

- Land has to have 3 Starbucks.

Post-conditions:

- The player has a Bilka on his/her land.

Entry-conditions:

- Player rolls the dice and goes to the land of the same color set he/she wants to buy a Bilka for.

Exit conditions:

- Player passes his/her turn.

Main Event Flow:

- Player rolls the dice and goes to the land of the same color set.
- Player selects Bilka.
- Player's money decreases.
- 3 Starbucks on the land turn into a Bilka.
- Player passes his/her turn.

Alternative Event Flows:

- Player rolls the dice and goes to the land.
- First, the player buys the 3. Starbucks for the land.
- Then, the player buys the Bilka.
- 3 Starbucks turn into a Bilka.
- Player passes his/her turn.

## 7.2. Dynamic Models

### 7.2.1. Sequence Diagrams

#### 7.2.1.1. Player Starts Game

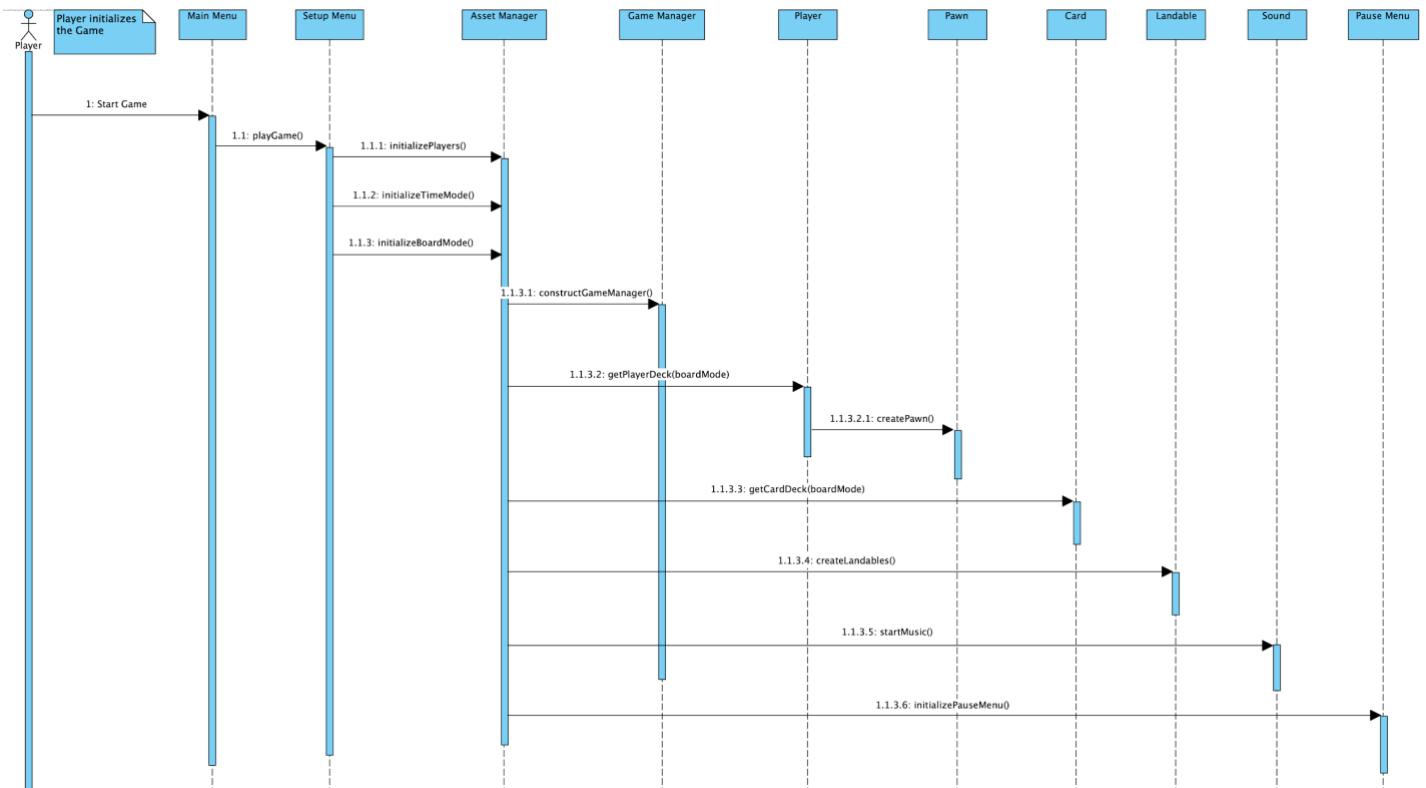


Figure 2. Player starts the game

**Scenario:** A player opens and starts the game

When the player runs the executable, he/she will see the Main Menu screen first. The player selects the Play Game option and goes to the setup process which consists of 2 screens, Player Selection Screen, Pre-Game Settings. On those screens, the player has to select initial settings and modes to start the game. When the player hits the Start button in the second screen, the mode and settings data is sent to Asset Manager which initializes the game and board. Asset Manager initializes the board by calling `startGame()` function of `GameManager`. Then, it calls `createPawn()`, `createCards()`, and `createLandables()` to initialize the game components. Finally, Asset Manager calls `startMusic()` of Sound Manager to start the background music and `initializePauseMenu()` to create Pause Menu at the background.

### 7.2.1.2. Player Picks Chance Card

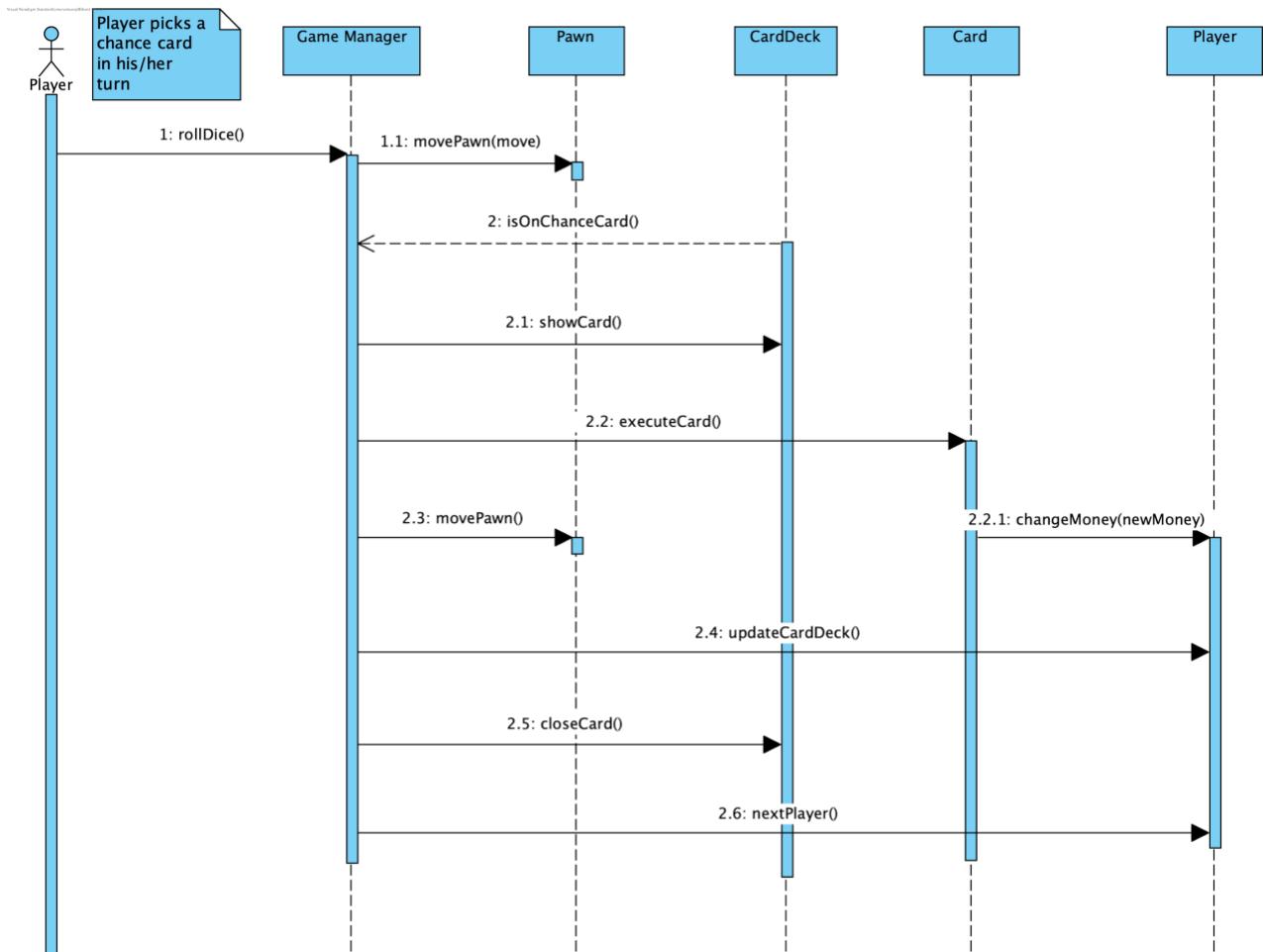


Figure 3. Player picks chance card

**Scenario:** A player plays his/her turn and comes to a Chance Card place

Player rolls the dice (`rollDice()`) and moves his/her pawn. Game Manager calls `movePawn()` of Pawn instance and pawn goes to a Chance Card place. Card Place instance gives feedback to the Game Manager indicating the pawn is on a Chance Card place and Game Manager calls `showCard()` with a Chance Card argument to show to Chance Card Pop-up. Player hits the Okay button on the Pop-up to execute the card. Game Manager calls `executeCard()` of Card instance and this function handles the Card's action (it could require `movePawn()` to send the player to another place or `changeMoney()` to give or take some money to the player). Then, Game Manager calls `closeCard()` to close the Pop-up and `nextPlayer()` for the next turn.

### 7.2.1.3. Player Buys Land

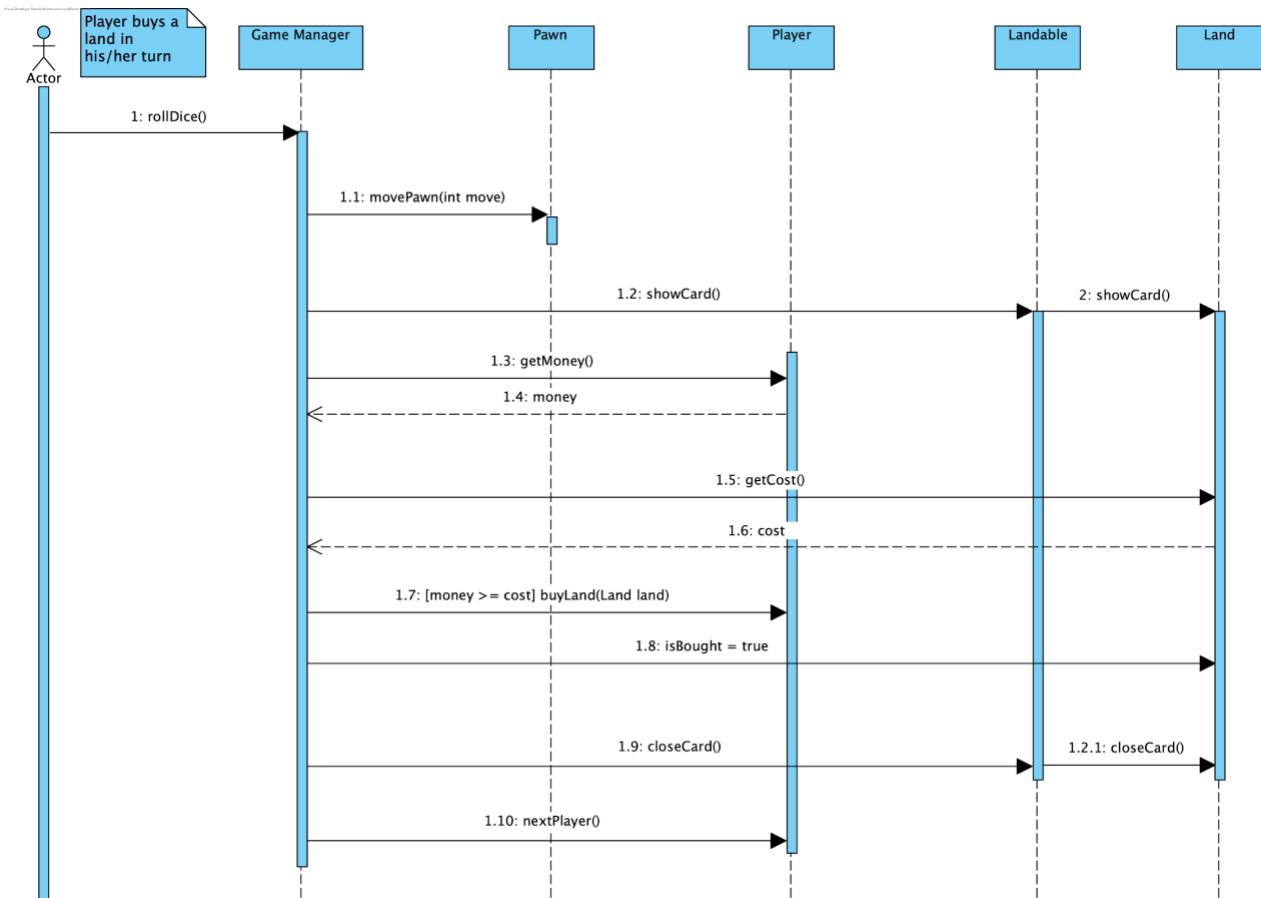


Figure 4. Player buys land

**Scenario:** A player plays his/her turn and buys the land

Player rolls the dice (`rollDice()`) and moves his/her pawn. **Game Manager** calls `movePawn()` of **Pawn** instance and pawn goes to a **Land** which is not bought yet. Game Manager asks Player his money and to Land its cost by calling `getMoney()` and `getCost()`. If the player wants to buy the land and his/her money is enough to buy it, Game Manager calls `buyLandOrCafe()` function of Player to add the land to his/her `lands[]` array. Then Game Manager assigns `isBought` boolean of **Land** to true which indicates that the land is not available anymore. Finally, Game Manager closes the Land card and calls `nextPlayer()` for the next turn.

#### 7.2.1.4. Player Goes to Atalar's Room

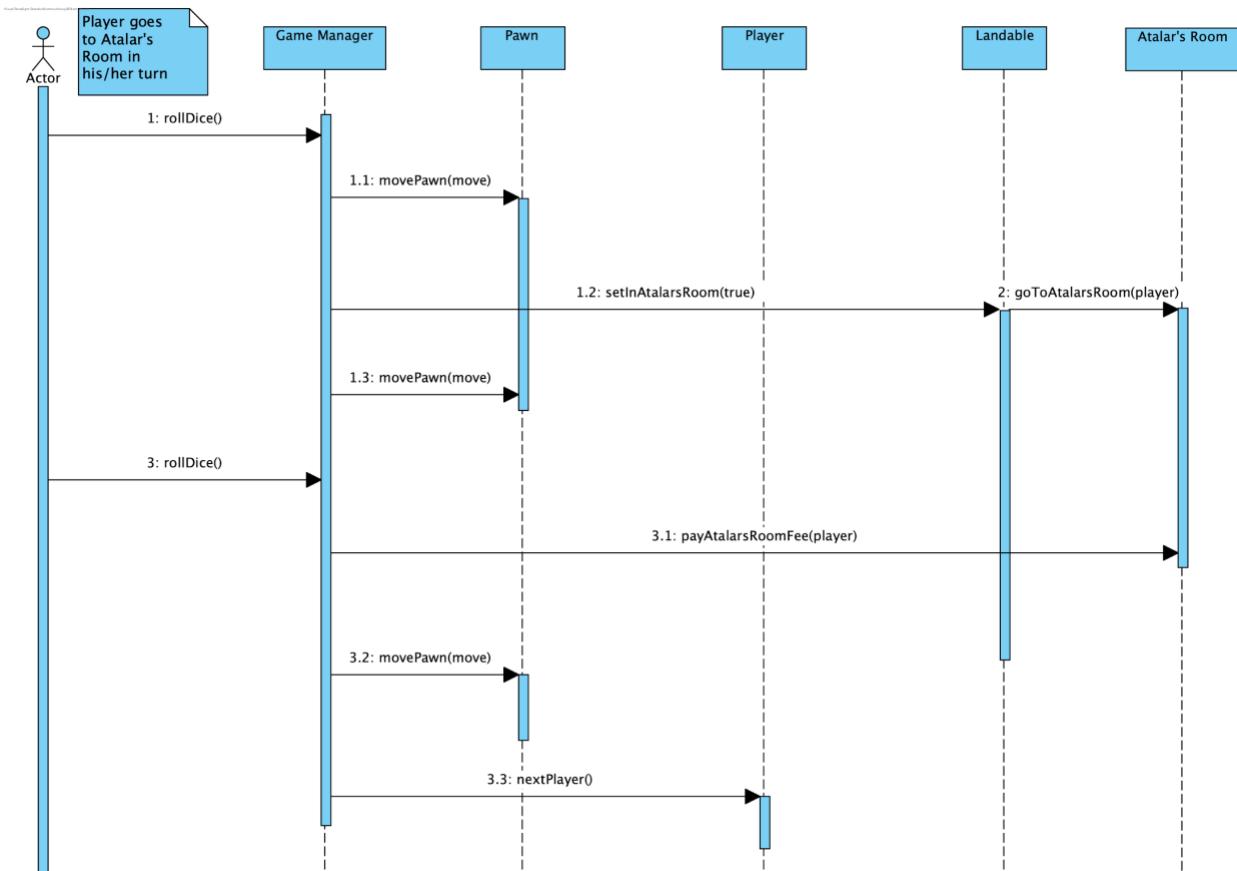


Figure 5. Player goes to Atalar's Room

#### Scenario: A player goes to Atalar's Room in his/her turn

The player rolls the dice, **rollDice()** method is called. Then the pawn of the player is moved with the **movePawn()** method. The player lands on “Go to Atalar’s Room” landable, which makes him/her go to “Atalar’s Room” landable. **goToAtalarsRoom()** method is called and the player’s pawn is moved to Atalar’s Room with the **movePawn()** method. For the next two turns, player rolls the dice; if the dice are equal, player can leave Atalar’s Room. If not, player has to pay the fee and leave at the end of the third turn.

### 7.2.1.5. Player Pauses Game

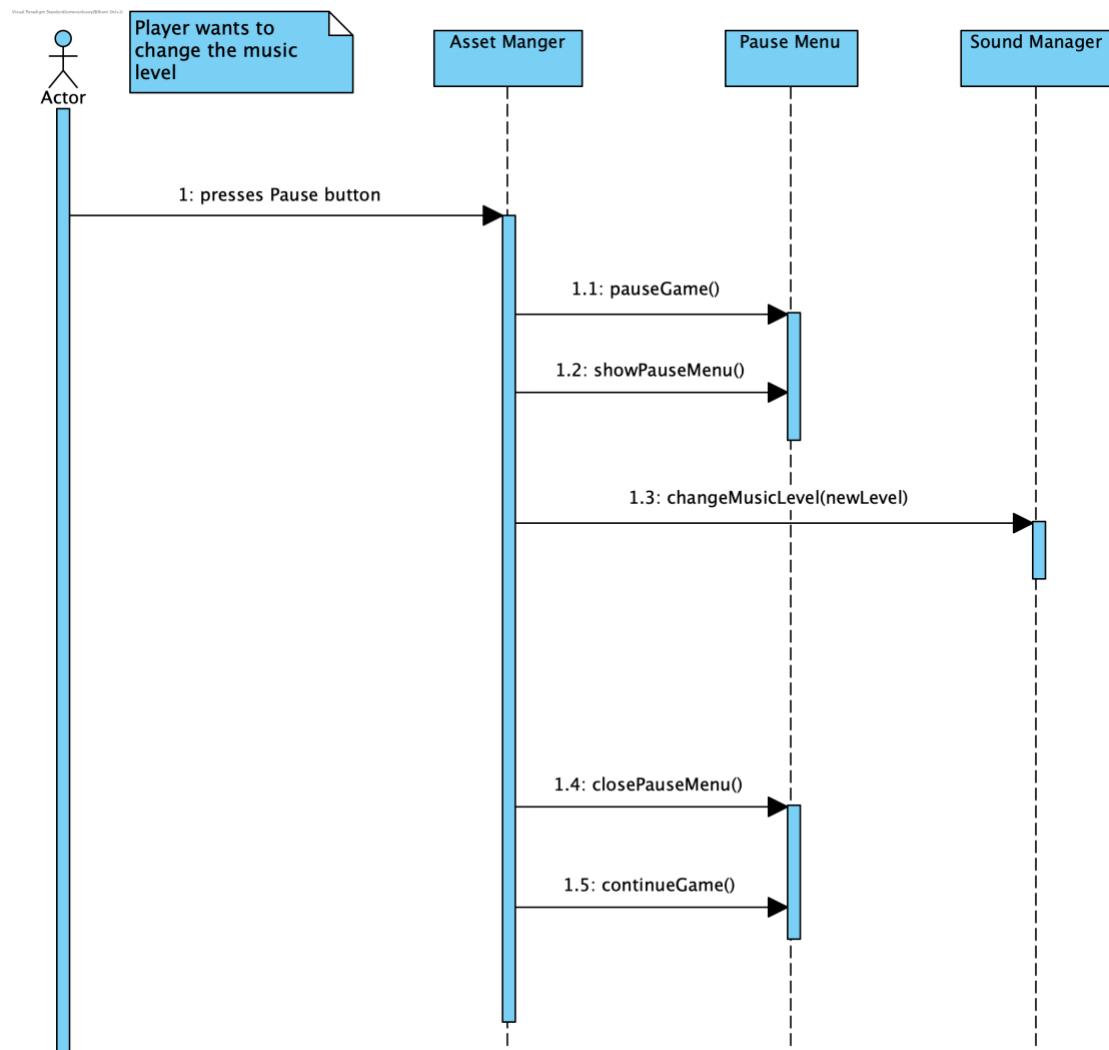


Figure 6. Player pauses game

**Scenario:** Player pauses the game in mid-game and changes the music level

The player clicks on the pause button at the top-right of the screen. The Asset Manager calls the `pauseGame()` and `showPauseMenu()` functions. Pause menu is displayed on the screen. The player adjusts the music level on the pause menu which makes the Asset Manager call the `changeMusicLevel()` function. Then, the player closes the pause menu, `closePauseMenu()` function is called, and the game continues.

### 7.2.1.6. Game Over

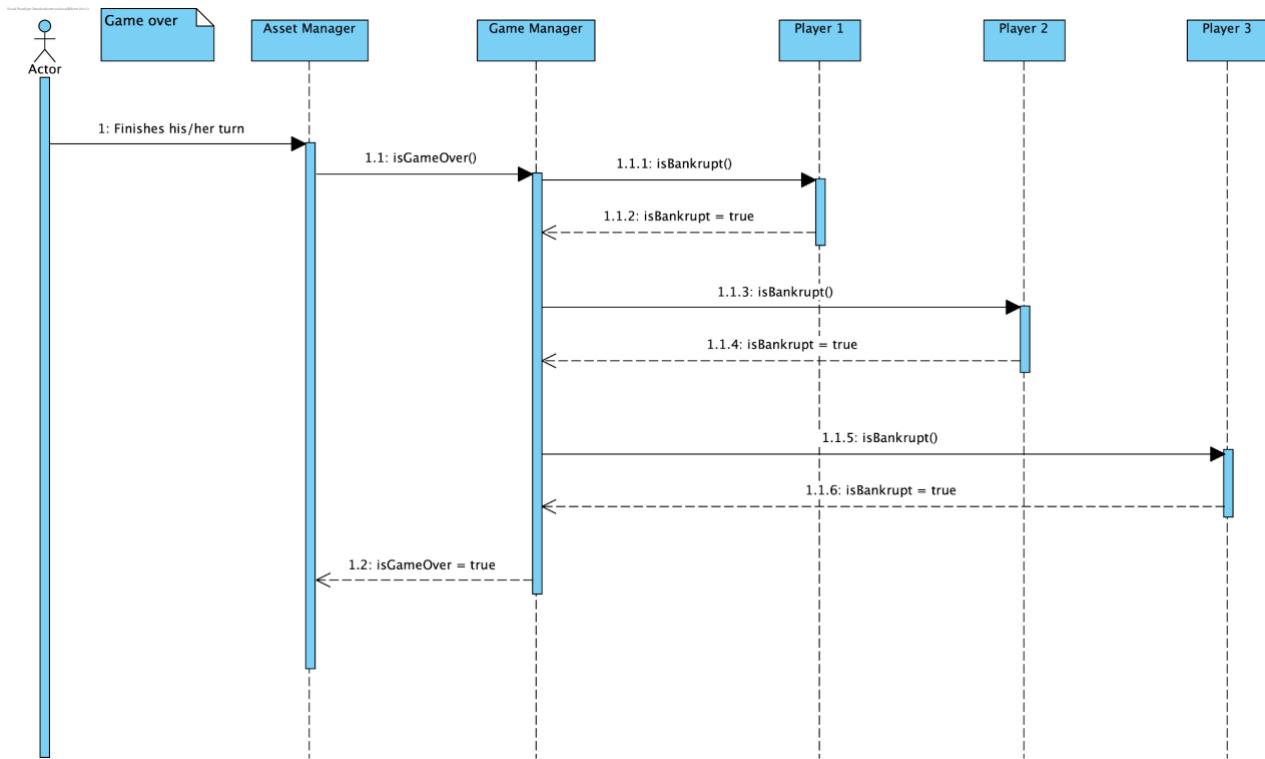


Figure 7. Game over

**Scenario:** Everyone except one player bankrupts and the game ends

At each turn, after a player plays his/her turn, `isGameOver()` function is called to check the game state. In this scenario, a player plays his/her turn and `isGameOver()` function is called. `isGameOver()` function calls `isBankrupt()` function for each player. Three players are bankrupt on this turn. Game over condition is satisfied and the game finishes.

## 7.2.2. Activity Diagram

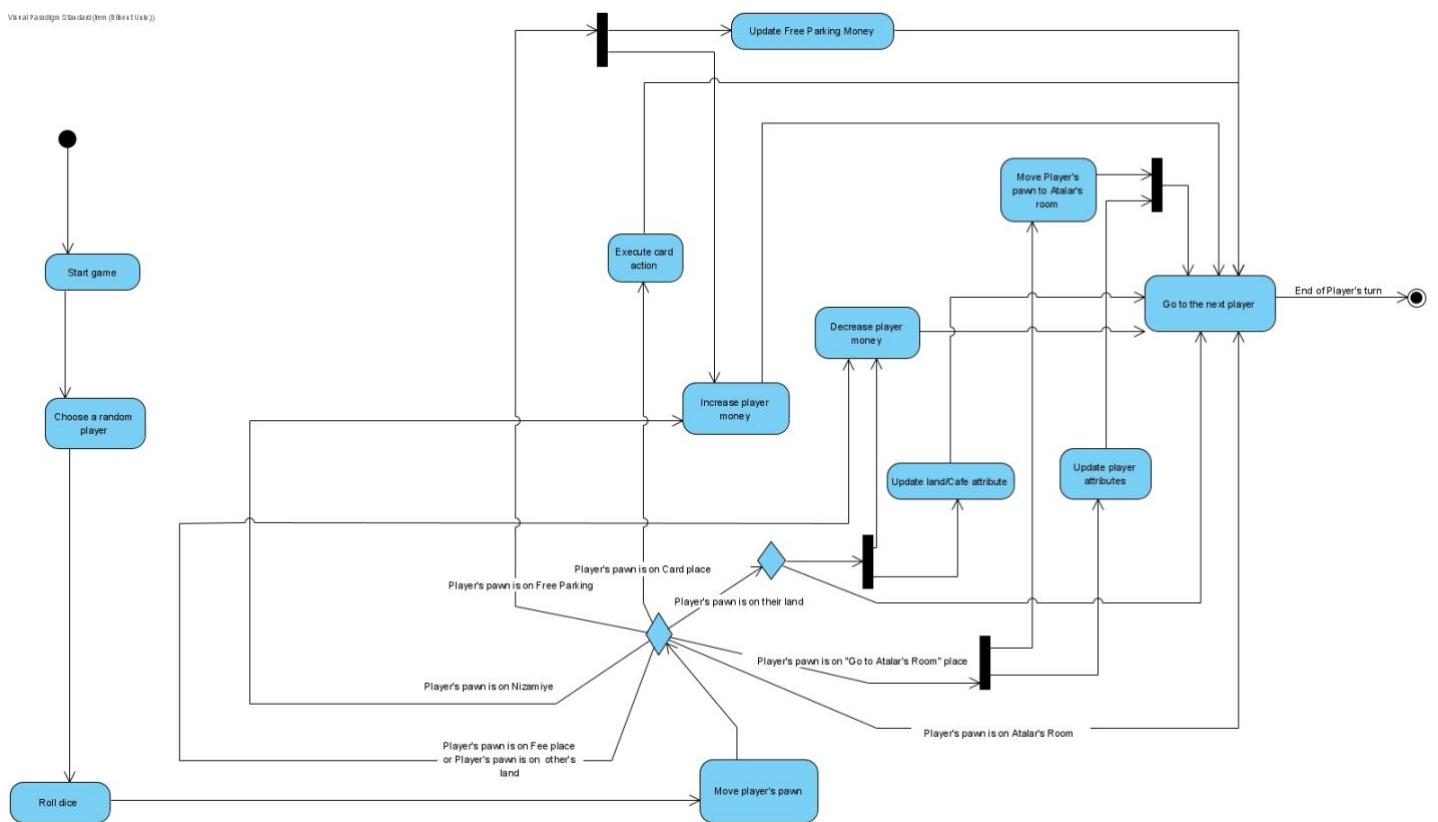


Figure 8. Activity diagram of turn of a player

This diagram shows the system's activity during the turn of a player. After the game starts a player is randomly selected as the first player and rolls the dice. Player's pawn moves to a place on the game board according to the dice value. The pawn may land on a land, cafe, card place, go to a functional place such as "Go to Atalar's Room", "Nizamiye" etc. Player takes action according to the landed place. Then, it is the next player's turn. If a player lands on another player's land, he/she pays rent.

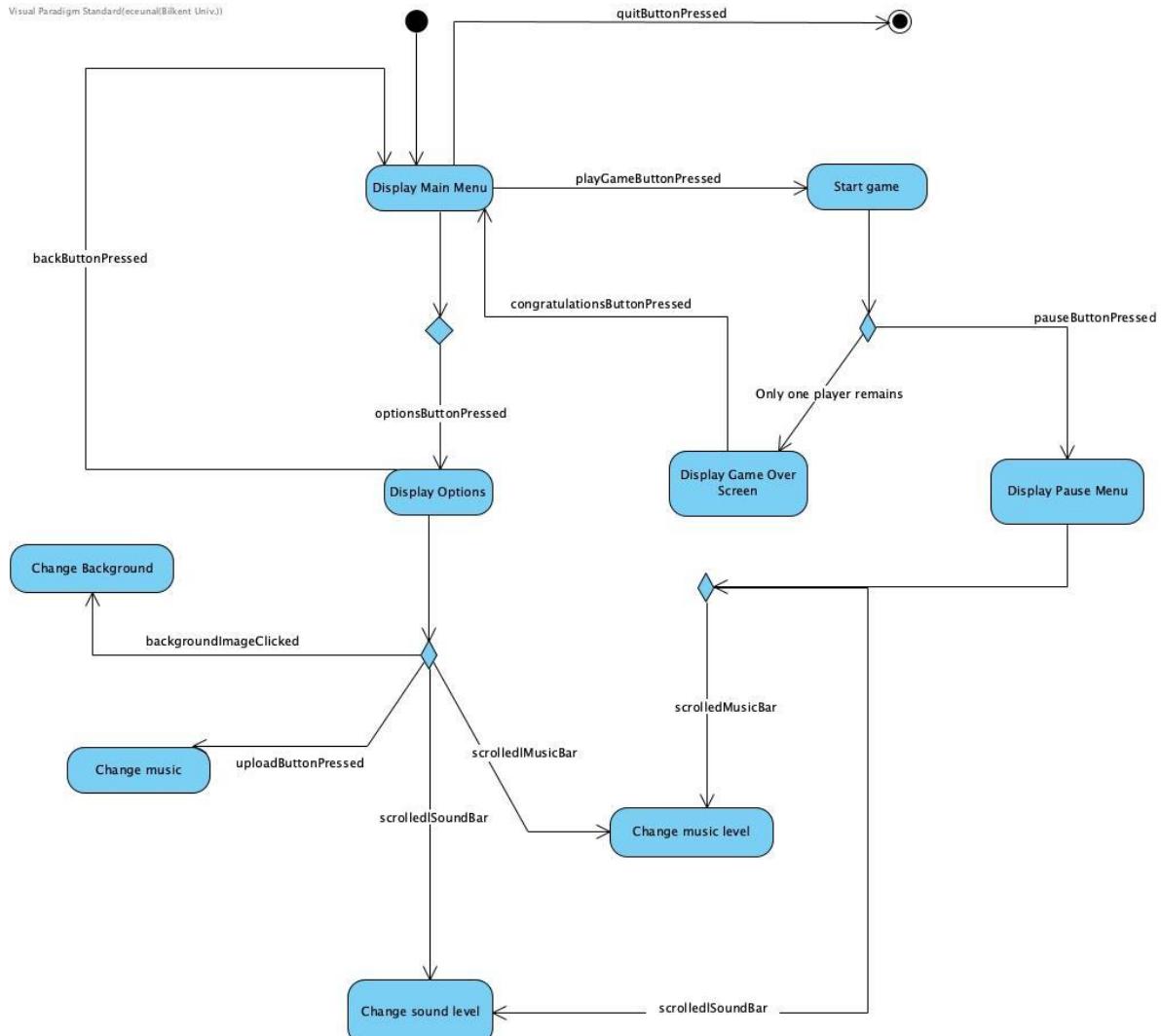


Figure 9. Activity diagram of SoundManager

This diagram shows the system's activity of the Sound Manager. In the Main Menu there is an “Options” button which leads to “Options” screen that includes; background options, music selection and music level options, sound level options. Music and sound level options can also be reached through Pause Menu which is opened with the “Pause” button on the game board.

### 7.2.3. State Diagram

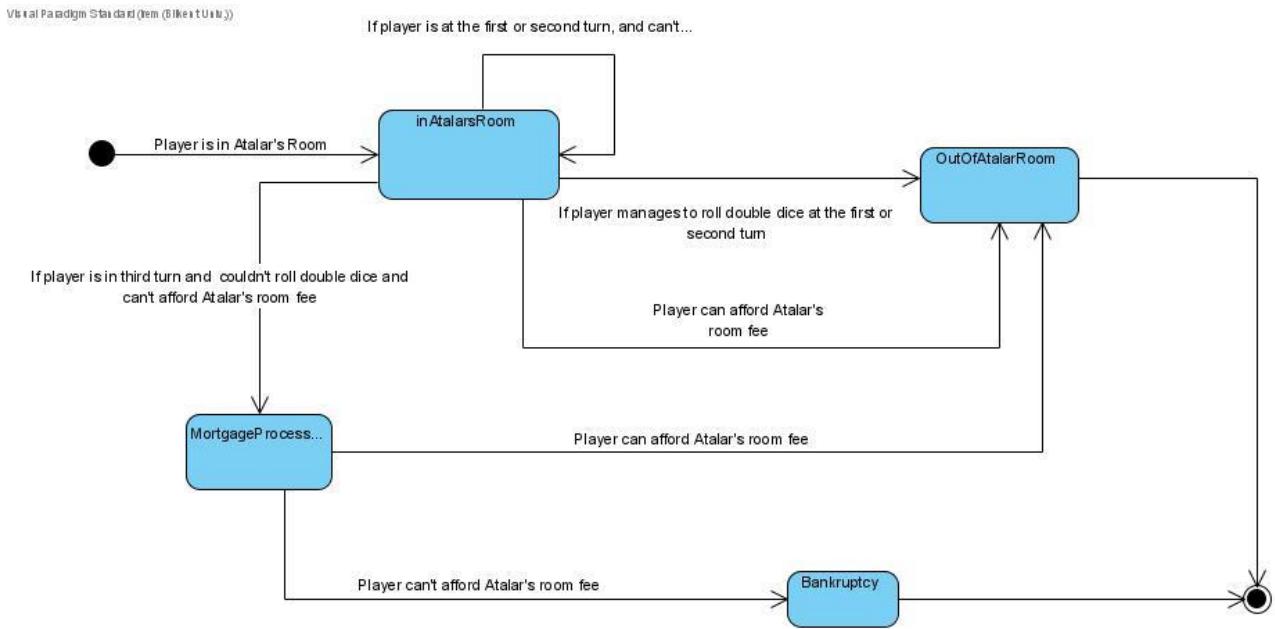


Figure 10. State diagram of Atalar's Room

This is the state diagram of Atalar's Room. If a player goes into Atalar's Room in his/her turn, turn goes to the next player. When it is the player's turn who is in Atalar's Room, the player rolls the dice. If the player can roll double dice, s/he is out of Atalar's Room, else the player waits for his/her next turn. If the player cannot roll double dice at the end of the third turn, s/he has to pay the Atalar's Room fee to get out of Atalar's Room. If the player doesn't have enough money to pay the fee, player goes into the Mortgage Process and if still cannot afford the fee, go Bankruptcy and get out of the game.

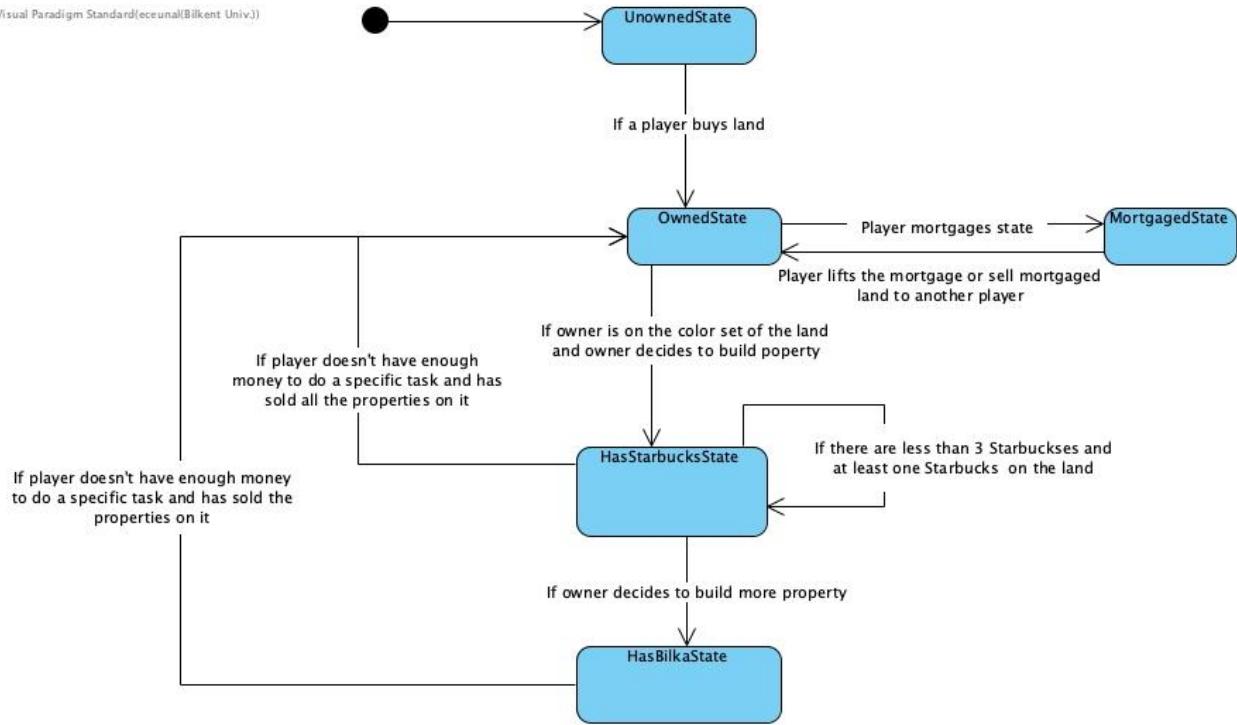


Figure 11. State diagram of a land

This is the state diagram of a land. First it is in an unowned state when nobody has bought it yet. When a player buys it, it is in owned state. If the owner' pawn is on the color set of a land and all the lands of the land's color set is bought by the same player, the player can build properties on the land. Maximum three Starbukses can be built on the land then the owner can build one Bilka. If the player doesn't have enough money to do a specific task, the player may sell the properties on the land then may mortgage the land. When the player has enough money s/he can lift the mortgage. If another player's pawn lands on the mortgaged land and that player wants to buy the property, if the owner who mortgaged the land allows it, can buy the property.

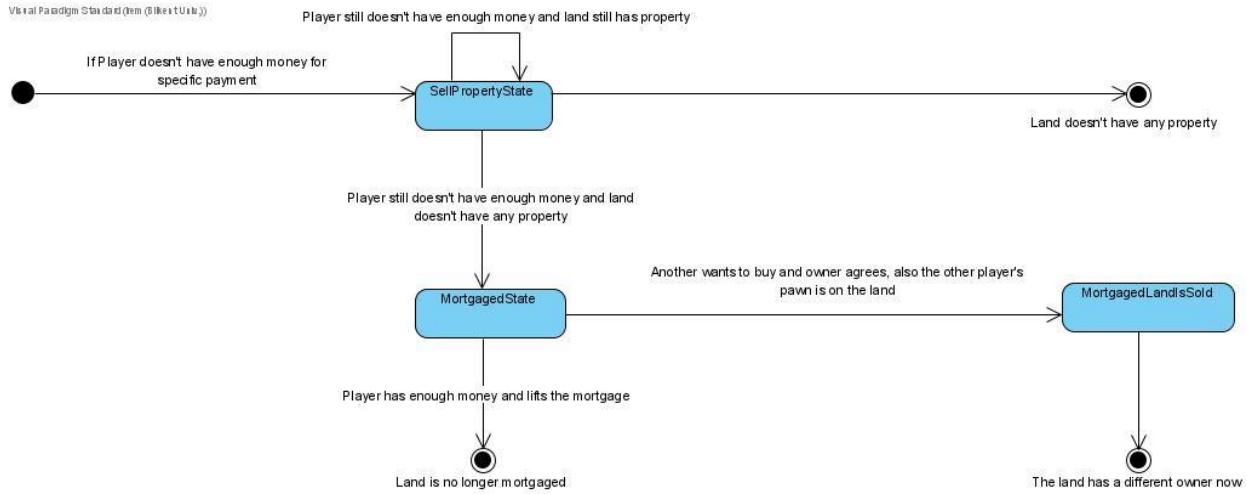


Figure 12. State diagram of mortgage process for a Land

This is the state diagram of the Mortgage Process for a single land. Since the player cannot mortgage their land with properties on it, he/she has to sell the properties first. After that, if they still don't have enough money for paying the specific amount for rent, Atalar's Room fee, card penalty etc., they can mortgage their land, and pay money with the income of it. Also, if land is mortgaged it can be sold to another player if buyer and seller agree on it while buyer's pawn in on the land.

## 7.3. Object and Class Model

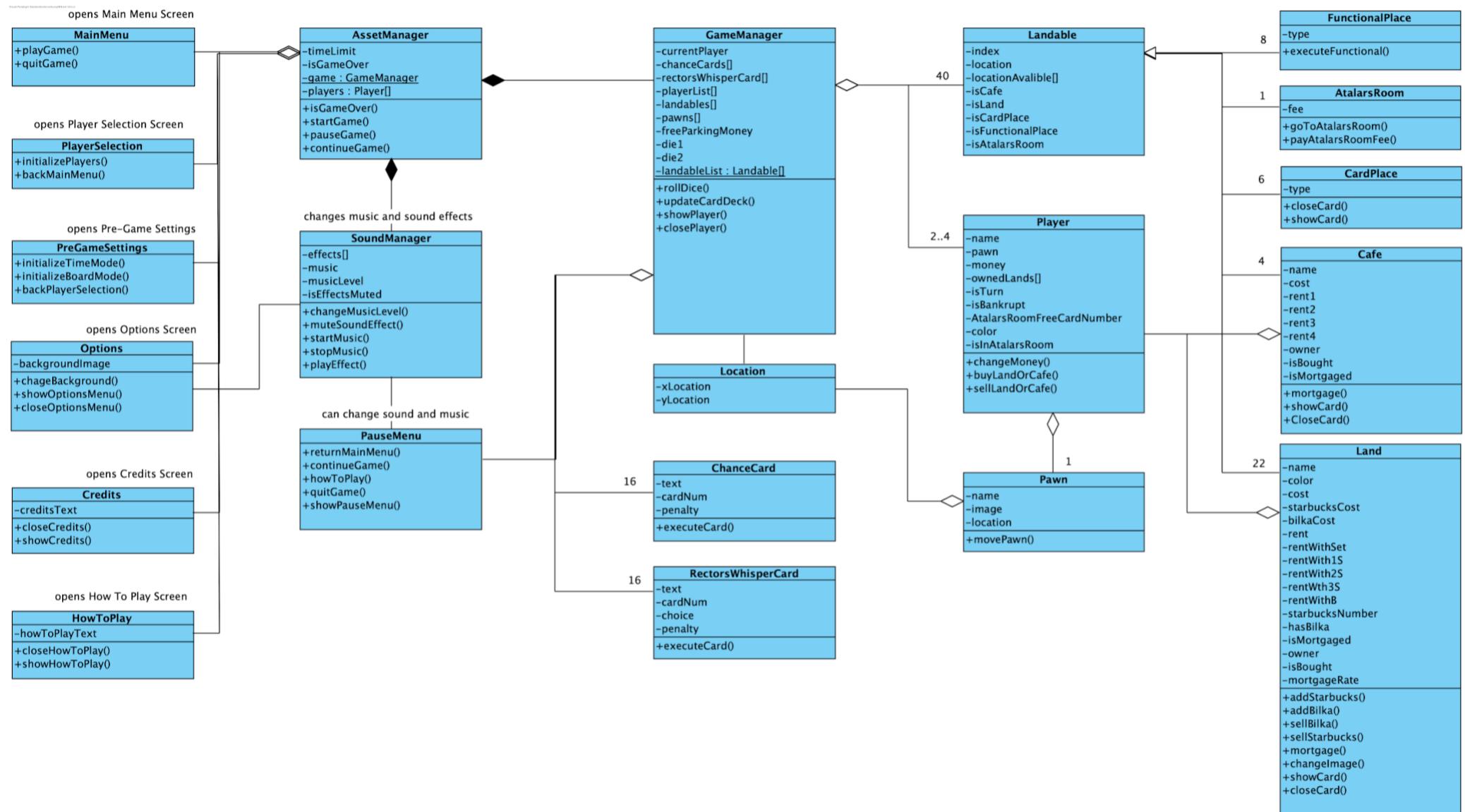


Figure 13. Object and class model

As illustrated above, Bilpoly game has 21 classes. While some of these classes serve as instances, rest is for game management or inheritance.

### **7.3.1. MainMenu**

This class constructs the first window, the Main Menu, the player sees. Main Menu has several options such as “Play Game”, “Options”, “How To Play”, “Credits”, and “Quit”. This class calls proper functions according to the player’s choice.

### **7.3.2. Options**

This class enables players to choose their own background image. To do that, this class constructs a window for players.

### **7.3.3. Credits**

This class is responsible for the credits window that shows creditsText and our images.

### **7.3.4. HowToPlay**

This class enables players to learn or remember the game rules by showing the “How to Play” window. Since the original Monopoly has 4+ pages of rules, we probably show the fundamental rules and the changed rules which is special for Bilpoly.

### **7.3.5. PlayerSelection**

This class constructs the first one of two consecutive windows (views) for the player to set up a game with desired modes and options. Player can access this setup interface by selecting the “Play Game” option in the Main Menu. In the first window, Player Selection, players should decide the number of people that will play the game; 2, 3, or 4 players are the options. After deciding the player number, each player should enter a unique nickname and

select a pawn. Then, players can go to the second window by selecting the “Next” button.

### **7.3.6. PreGameSettings**

This class constructs the second one of two consecutive windows for the player to set up a game with desired modes and options. In this window, players can decide the initial money, Board Mode, and Time Mode. Finally this class calls startGame() function from GameManager to start the game.

### **7.3.7. SoundManager**

This class handles the sound settings during the game. These settings can be adjusted from the Pause Menu. Bilpoly has two types of sounds, sound effects and music that plays in the background. Players can change the background music as well as the sound level of effects. When a player plays his/her turn, a proper effect will be played.

### **7.3.8. PauseMenu**

This class enables players to pause the game and change some settings such as music level, sound effects level. Players also can open the “How to Play” window to check some rules or go back to the Main Menu which will close the game immediately. A player can go back to the game by pressing the back button. This class calls functions of SoundManager and GameManager.

### **7.3.9. AssetManager**

AssetManager class is the top class that operates during a game. It allows other classes to communicate with each other during the game. It knows the game modes and settings and it initiates the game. It creates the board and the GameManager. It also decides to end the game if the time limit is reached or one player wins by bankrupting others. It decides the winner and shows the “Game Over” screen.

### **7.3.10. GameManager**

GameManager class is the most complex class of this project. It handles each player's turn and decides who is next. It rolls the dice and maintains cash flow during a turn. It knows the Chance and Rector's Whisper Cards, and all landables (lands, cafes, Atalar's Room, and other Functional Places) to deal with all possible actions during a turn. It also shows Player Info and updates Player Cards (Credit Cards) accordingly.

### **7.3.11. Player**

Each player has a Player class instance that holds their information. Each Player instance is initialized with a player name, player's pawn, initial money and a color that represents his/her Credit Card. Each turn GameManager calls Player.changeMoney() to update that player's money. When a player buys a land or cafe, his/her land array is updated. If a player goes to Atalar's Room (jail), an instance (isInAtalarsRoom) is updated.

### **7.3.12. ChanceCard**

Each Chance card has its own Card instance. This class holds the card's text. When its function (executeCard()) is called by GameManager, card's actions are executed (player could earn money, lose money, or other mumbo-jumbos can happen).

### **7.3.13. RectorsWhisperCard**

Each Rector's Whisper card has its own Card instance. This class holds the card's text and the player's choice (some cards have options). When its function (executeCard()) is called by GameManager, card's actions are executed (player could earn money, lose money, or other mumbo-jumbos can happen).

### **7.3.14. Pawn**

Each player has a pawn that specifies his/her location on the board. Each pawn has a unique name and image. It also has a location information of players. GameManager can move a pawn by calling its movePawn() function.

### **7.3.15. Location**

Each point on the board is specified by a location information. Each location information has an x and y values according to the right bottom corner of the window.

### **7.3.16. Landable**

Landable class specifies each place on the board. A place could be a land such as SA-Building or a card place like Chance Card. It can also be a cafe, Atalar's Office, or other Functional Places such as taxes. Land, CardPlace, Cafe, FunctionalPlace, and AtalarsRoom classes inherit this class.

### **7.3.17. Land**

Each land with color code has its own Land class instance that shows its information. Each land has several constant properties such as name, color, its cost, 1 Starbucks cost, 1 Bilka cost, and its rent. Besides that, each land has variable properties such as its owner, its current Starbucks number, whether it has a Bilka, and whether it is mortgaged. GameManager can change these properties accordingly by calling its proper functions.

### **7.3.18. CardPlace**

Each card place on the board has this class's instance that shows its type. GameManager can show or close this card by calling its functions.

### **7.3.19.     Cafe**

There are 4 cafes on the board that players can buy. These cafes act like transportation in the original Monopoly. When a player has more than one cafe, the rent other players have to pay increases. However, players cannot buy Starbucks or a Bilka for cafes.

### **7.3.20.     AtalarsRoom**

Atalar's Room is the jail of Bilpoly that players have to go to if they come to the "Go to Atalar's Office" place or their chance card says so. A player try to roll double dice.

### **7.3.21.     FunctionalPlace**

Functional Places are the other places left on the board. "Nizamiye", the starting point and Fee places have their FunctionalPlace class instances. GameManager can call its executeFunctional() function to take their actions.

## 7.4. User Interface - Navigational Paths and Screen Mockups

### 7.4.1. Navigational Path

Navigational path of Bilpoly is illustrated in the following figure.

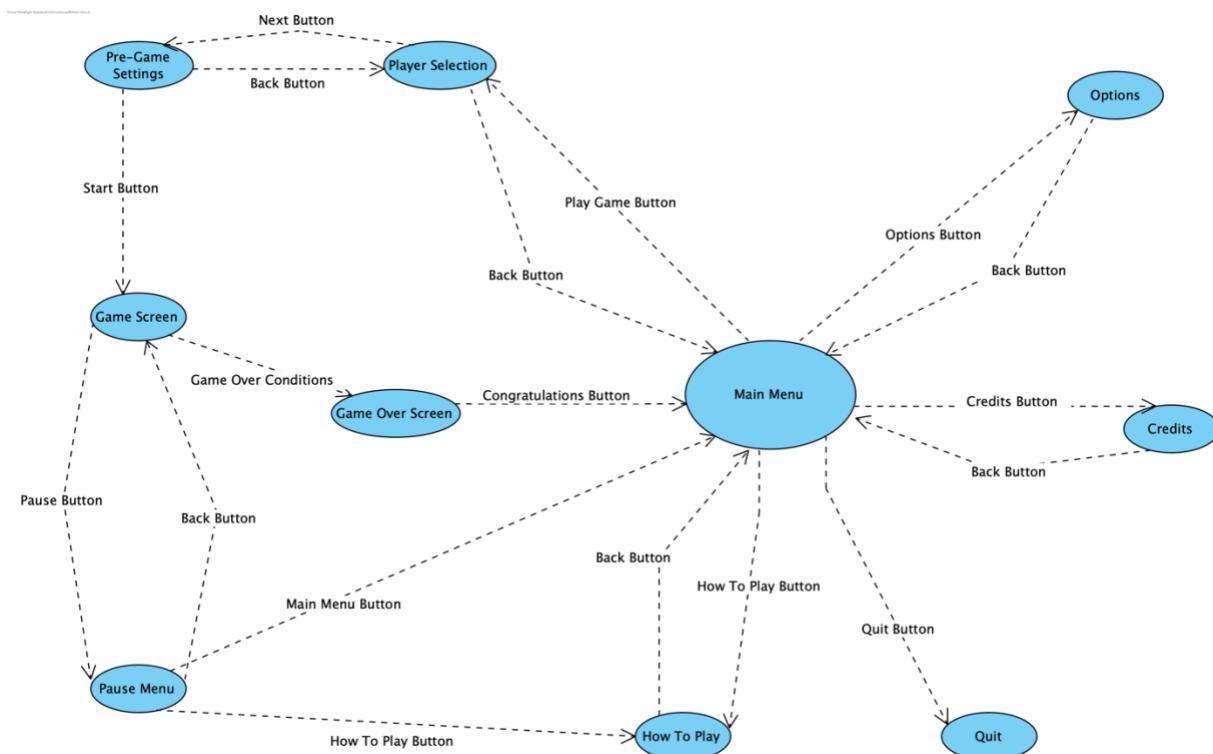


Figure 14. Navigational path

When the game is opened, the “Main Menu” screen is displayed on the screen. The user can navigate into the “Options Menu”, “How to Play” screen, and the “Credits” screen and go back to the main menu again. Also, the user can quit the game or click on the “Play Game” button and start to initialize the game.

After clicking the “Play Game” button, the “Player Selection” screen comes to the screen. The user can either initialize players or go back to the main menu.

Having initialized the players, the “Pre-Game Settings” screen comes to the screen. The user can either choose game modes and set initial money or go back to the player selection screen.

After the game starts, it could be paused and the “Pause Game” screen appears on the screen. On this screen, either the players can quit the game, or they can check out the “How to Play” screen. Also, the game could be resumed on this screen.

When the game ends, “Game Over” screen comes to the screen and the players can go to the main menu.

## 7.4.2. Screen Mock-ups

### 7.4.2.1. Navigation

#### 7.4.2.1.1. Main Menu



Figure 15. Main Menu

The main menu of Bilpoly is composed of the name of the game, a beautiful picture of Bilkent in the background, and 5 buttons which are “Play Game”, “Options”, “How to Play”, “Credits”, and “Quit”. The user can navigate through screens or quit the software via this menu.

#### 7.4.2.1.2. Options

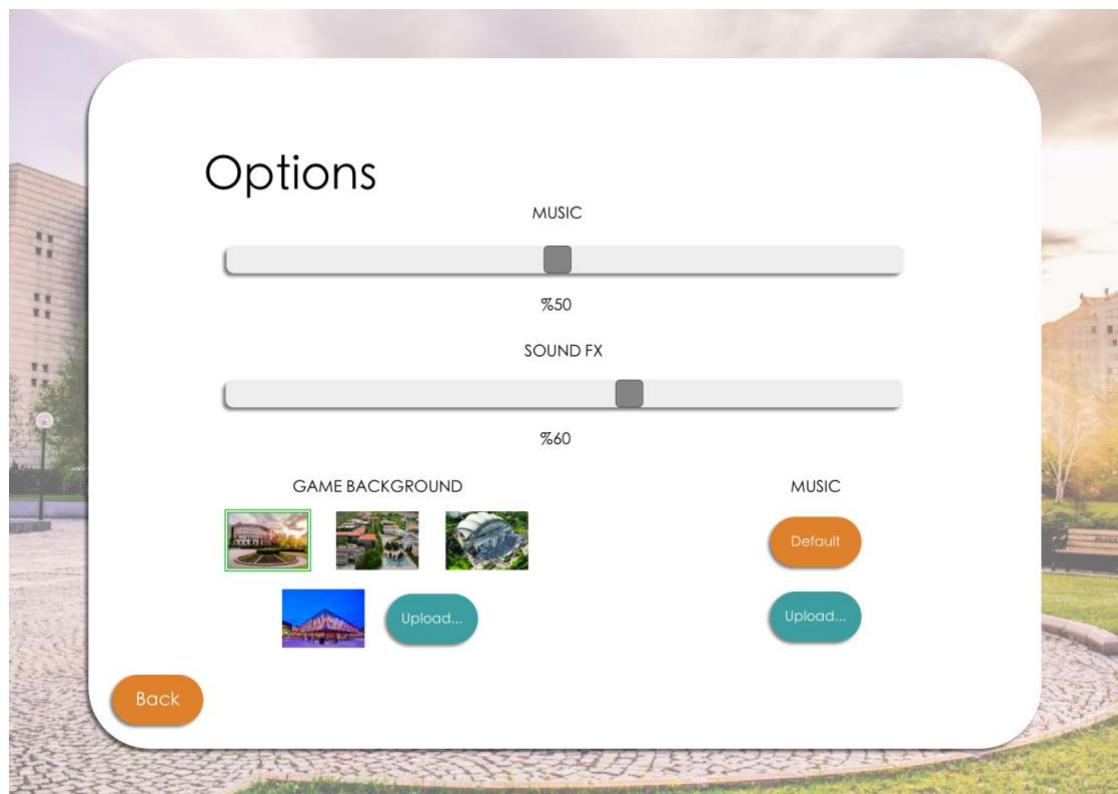


Figure 16. Options Menu

Options Menu of Bilpoly makes it able to change music and sound effects level. The user can also change the background of the game by choosing one of the preloaded backgrounds or uploading from local. Background music can also be chosen default or uploaded from local. By clicking “Back”, the user can go back to the main menu.

#### 7.4.2.1.3. How to Play

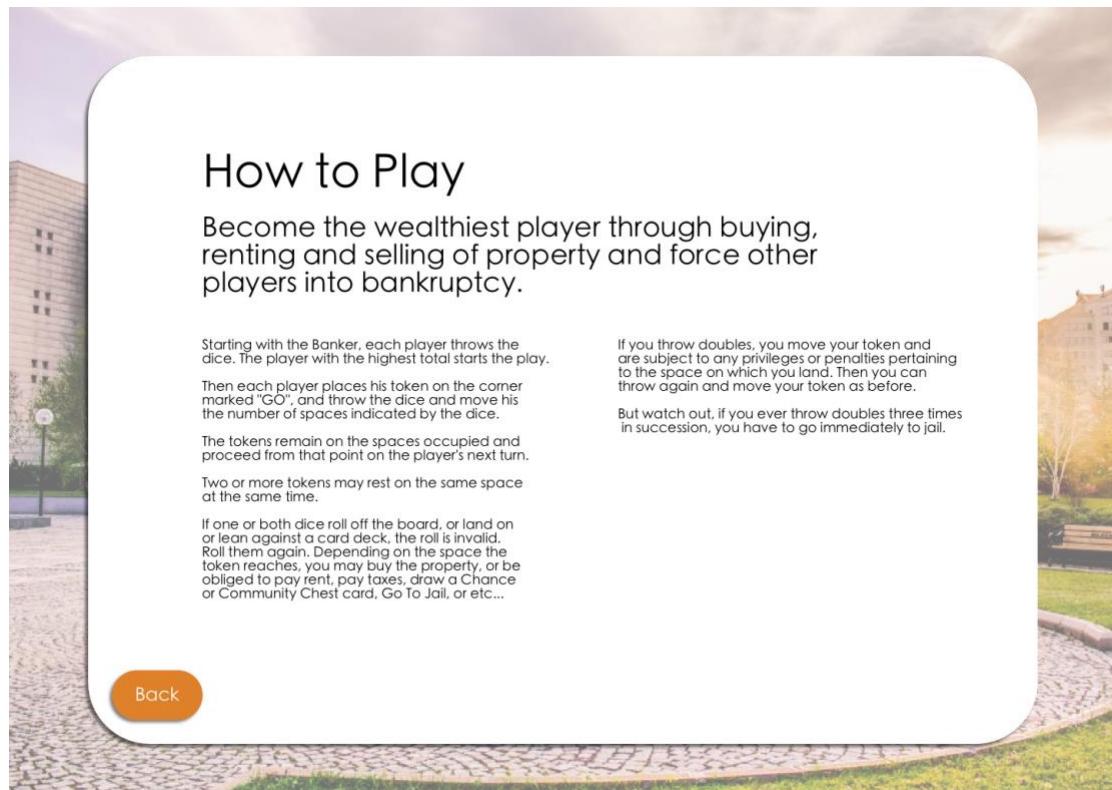


Figure 17. How to Play

The “How to Play” screen provides the user the rules of Bilpoly for a better game experience. “Back” button sends the user back to the main menu.

#### 7.4.2.1.4. Credits

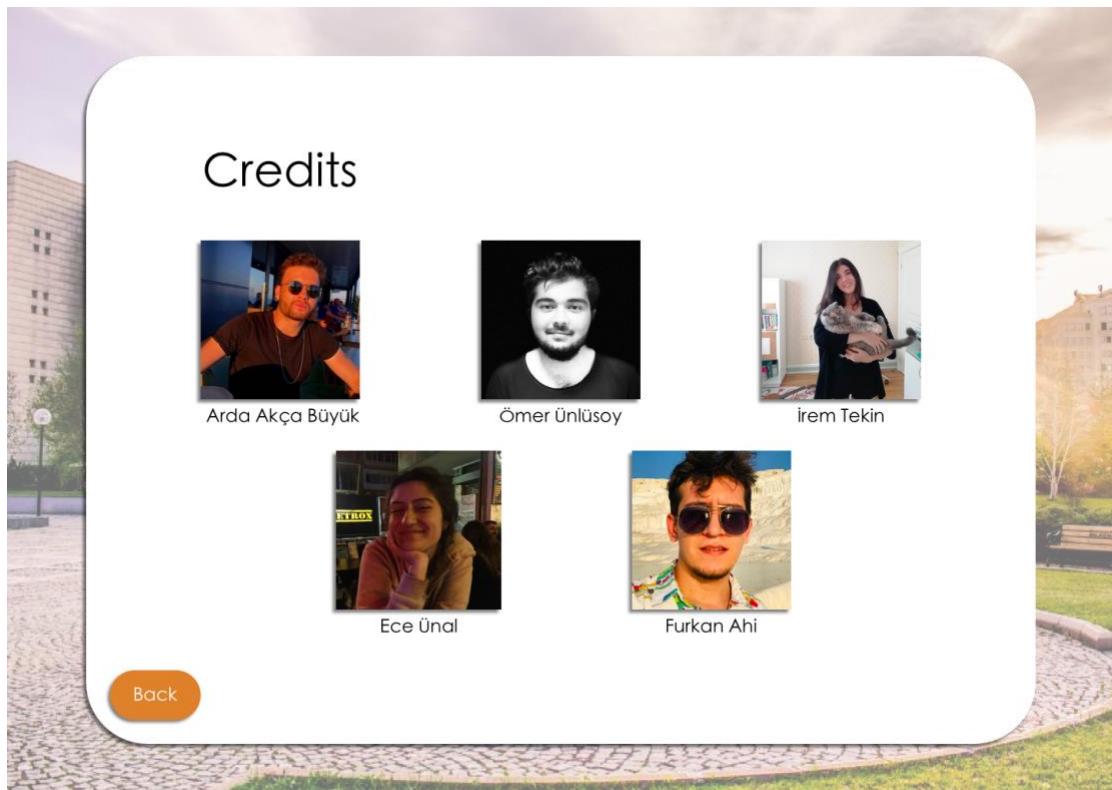


Figure 18. Credits

Credits screen consists of the pictures of the developer team. Main Menu can be reached via the “Back” button.

#### 7.4.2.1.5. Player Selection

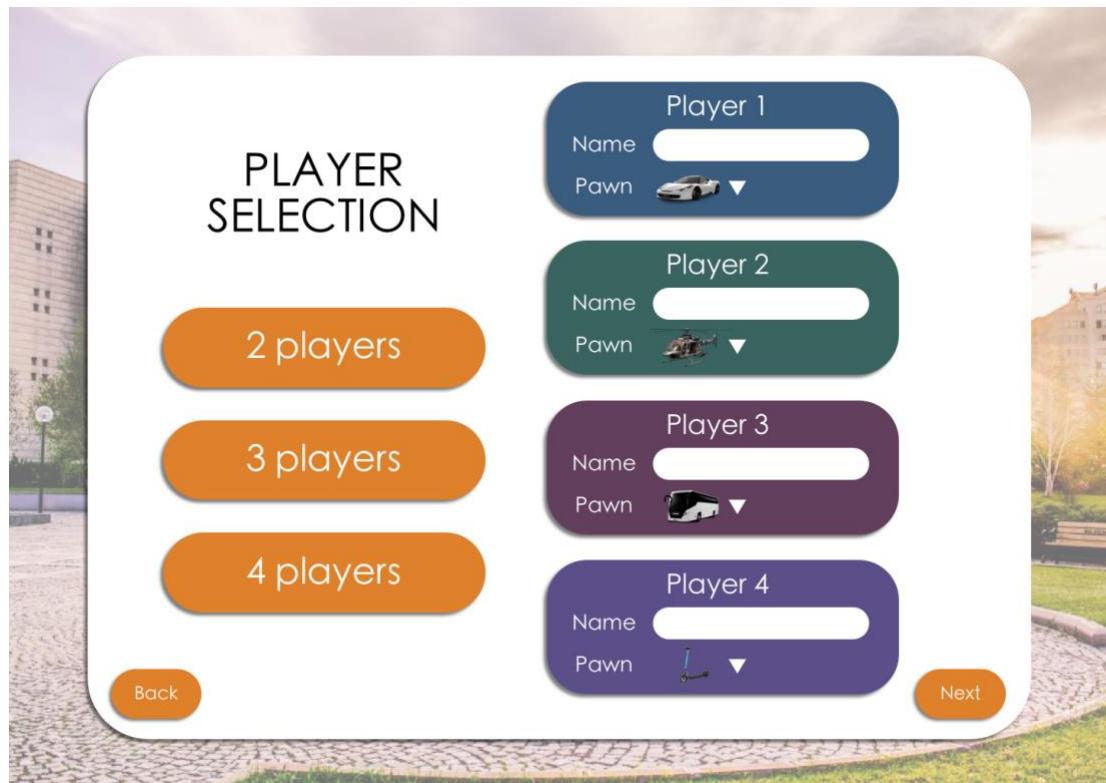


Figure 19. Player Selection

Player Selection Screen has a dynamic structure. Number of players is chosen from the left. According to the number of players, entries at the right are generated. The players can choose their names and pawns on their entries. Main Menu can be reached via the “Back” button. After initialization, players can proceed to the “Pre-game Settings” screen by clicking the “Next” button.

#### 7.4.2.1.6. Pre-game Settings

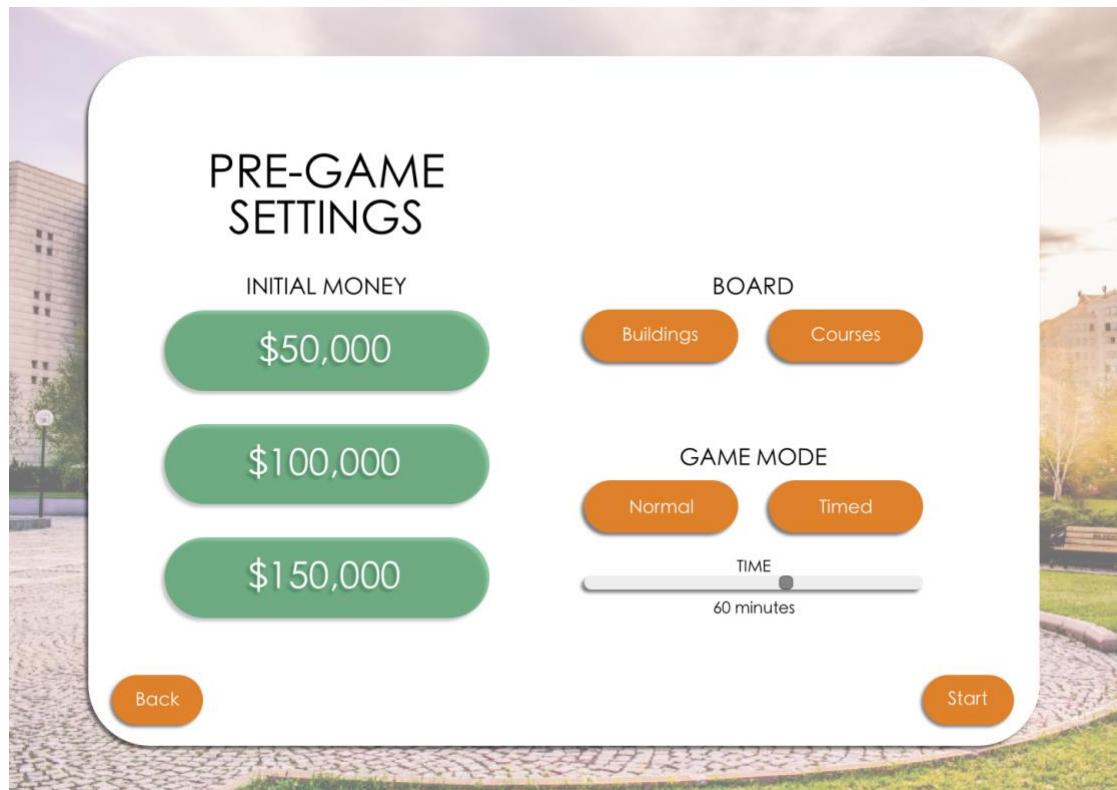


Figure 20. Pre-Game Settings

Pre-game settings screen makes it possible for players to choose the initial money and game board (either Bilkent Buildings Mode or Bilkent CS Mode) for the session. Also, the session can be initialized as normal or timed. In Timed Mode, the duration of the session can be set. “Start” button starts the game, and the “Back” button takes it back to the player selection screen.

## 7.4.2.2. In-Game

### 7.4.2.2.1. Bilkent Buildings Mode

#### 7.4.2.2.1.1. Game Screen

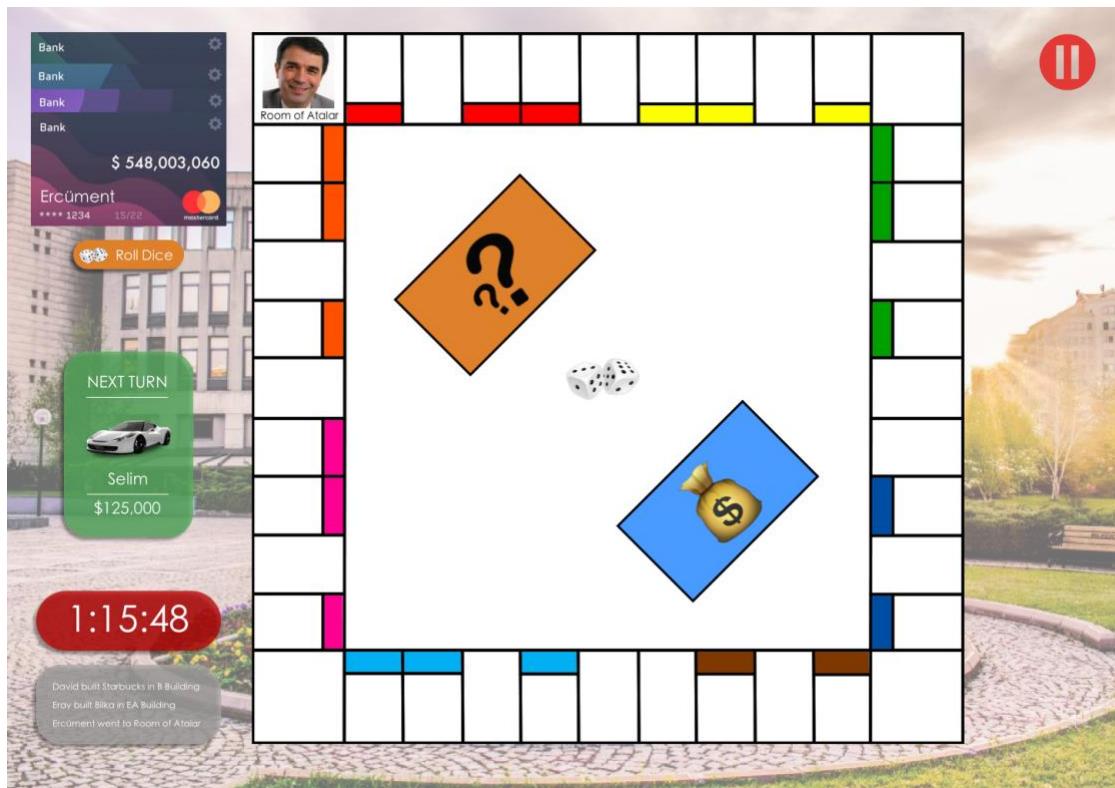


Figure 21. Game Screen of Buildings Mode

The game screen of Buildings Mode consists of numerous features.

At the top-left, players are represented as credit cards of their own. On credit cards, there are the name of the player and the balance of the player. Credit cards are sorted based on turns of the players. The credit card of the player that has the turn comes at the front.

The “roll dice” button under the credit card deck appears when a player takes the turn.

At the mid-left there is a card that shows the next player that will have the turn after the current player.

At the bottom-left there is the time left for the session in Timed Mode.

At the bottommost-left there are the last three actions taken by the players. When the current player plays turn, the uppermost action fades out and the new action is written on the bottom.

The game board consists of the buildings and cafes in Bilkent University. Each three buildings are grouped into a single color as in the original Monopoly game. Starting point is Nizamiye, the jail of the original Monopoly game is Atalar's Room, and the Free Parking of the original Monopoly game is Mayfest Parking Lot.

Dice rolling animation is in the middle of the board. Also, there are representations of Chance and Rector's Whisper Cards in the middle of the board as well.

The pause button is at the top-right of the screen.

#### 7.4.2.2.1.2. Player Info

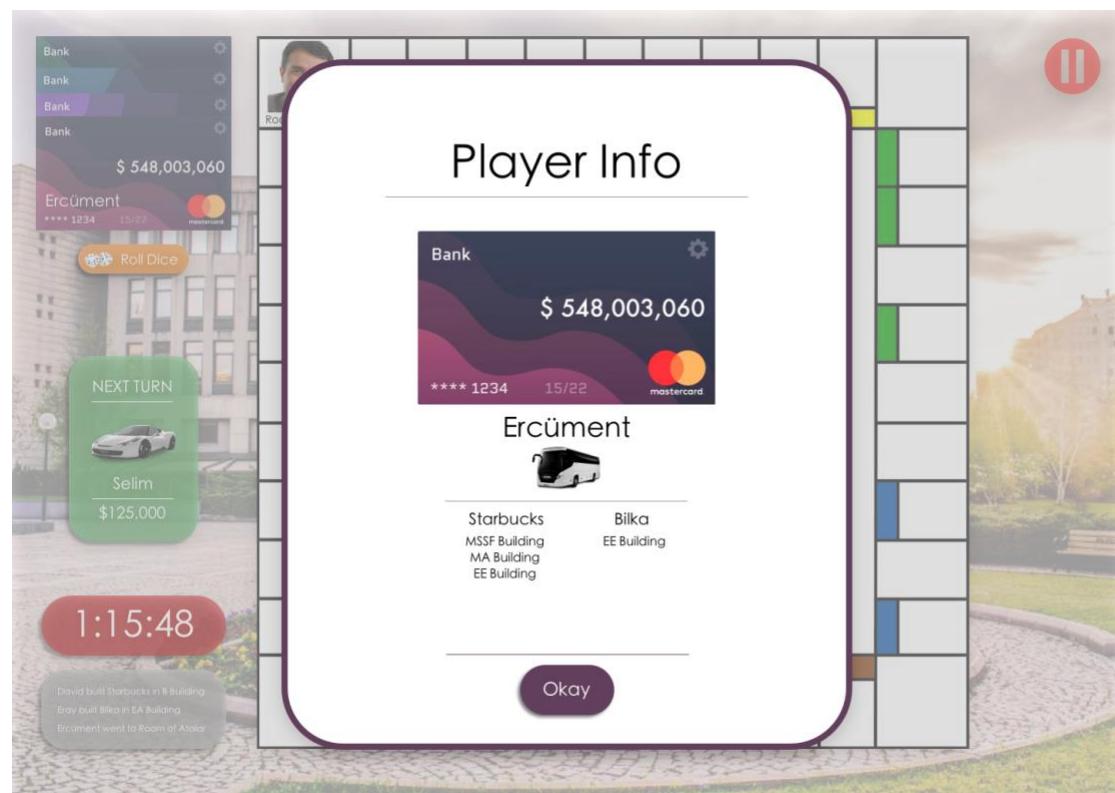


Figure 22. Player Info of Buildings Mode

By clicking the credit card of any player, information about that player can be displayed on the screen. Balance, name, pawn and owned buildings of the desired player is presented on this screen. The places that the player built

Starbuckses or Bilkas are categorized as well. The player can go back to the game by clicking the “Okay” button.

#### 7.4.2.2.1.3. Chance Card

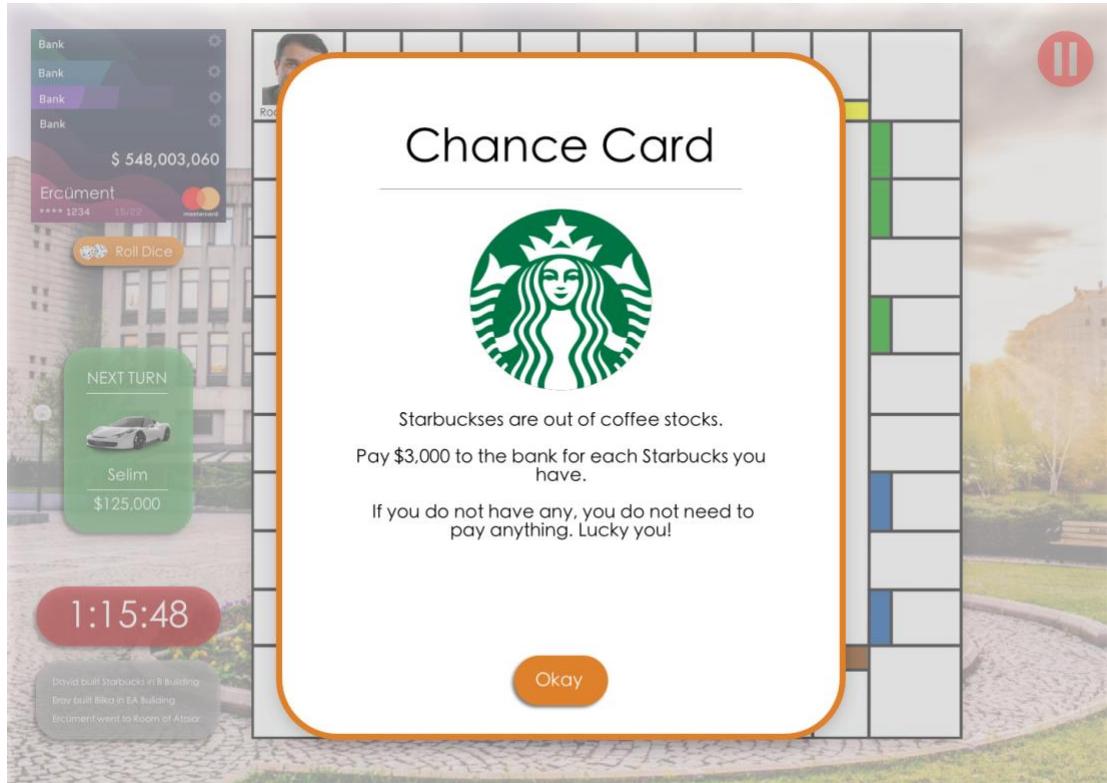


Figure 23. Chance Card of Buildings Mode

An example of a Chance Card is on the mockup above. There is an orange border outside of the card since Chance Cards are represented by the color orange. Each chance card has a picture that is associated with the content of the card. When the player clicks the “Okay” button, the chance card is executed according to its action.

#### 7.4.2.2.1.4. Rector's Whisper Card

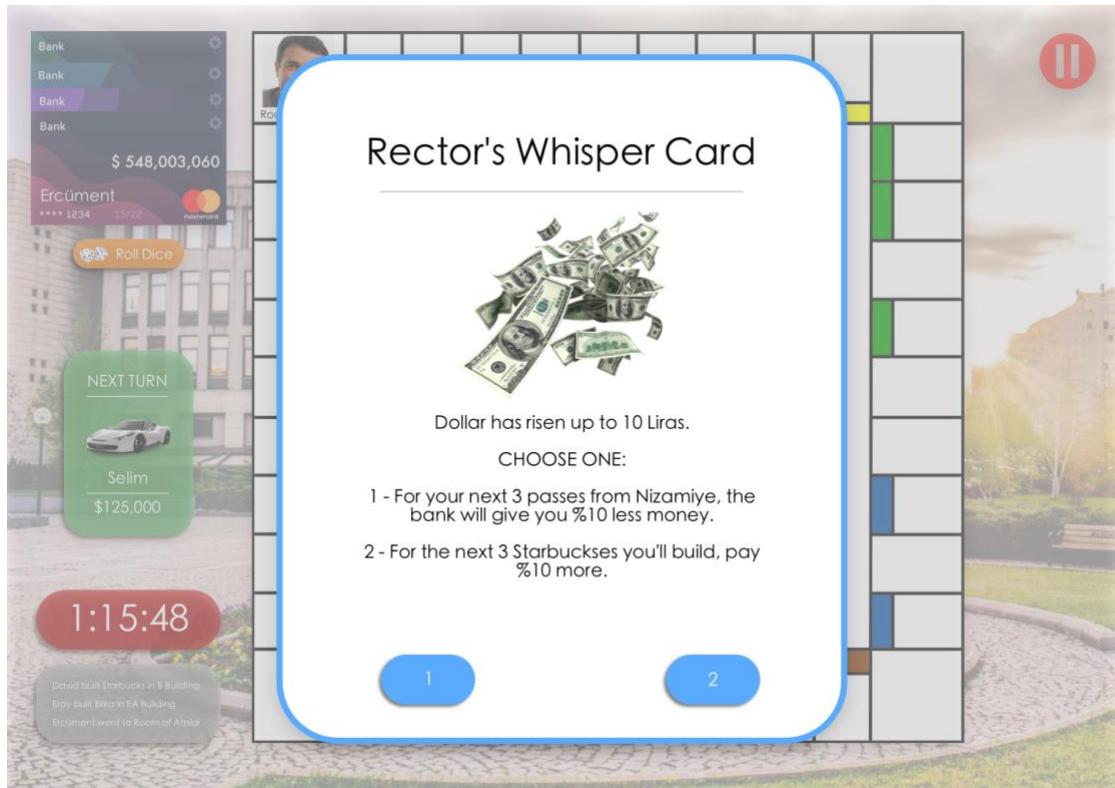


Figure 24. Rector's Whisper Card of Buildings Mode

An example of a Rector's Whisper Card is on the mockup above. There is a blue border outside of the card since Rector's Whisper Cards are represented by the color blue. Each Rector's Whisper Card has a picture that is associated with the content of the card. The player can choose one of the two options offered by the card. Both options have different actions. After selection, the card is executed.

#### 7.4.2.2.1.5. Building Card

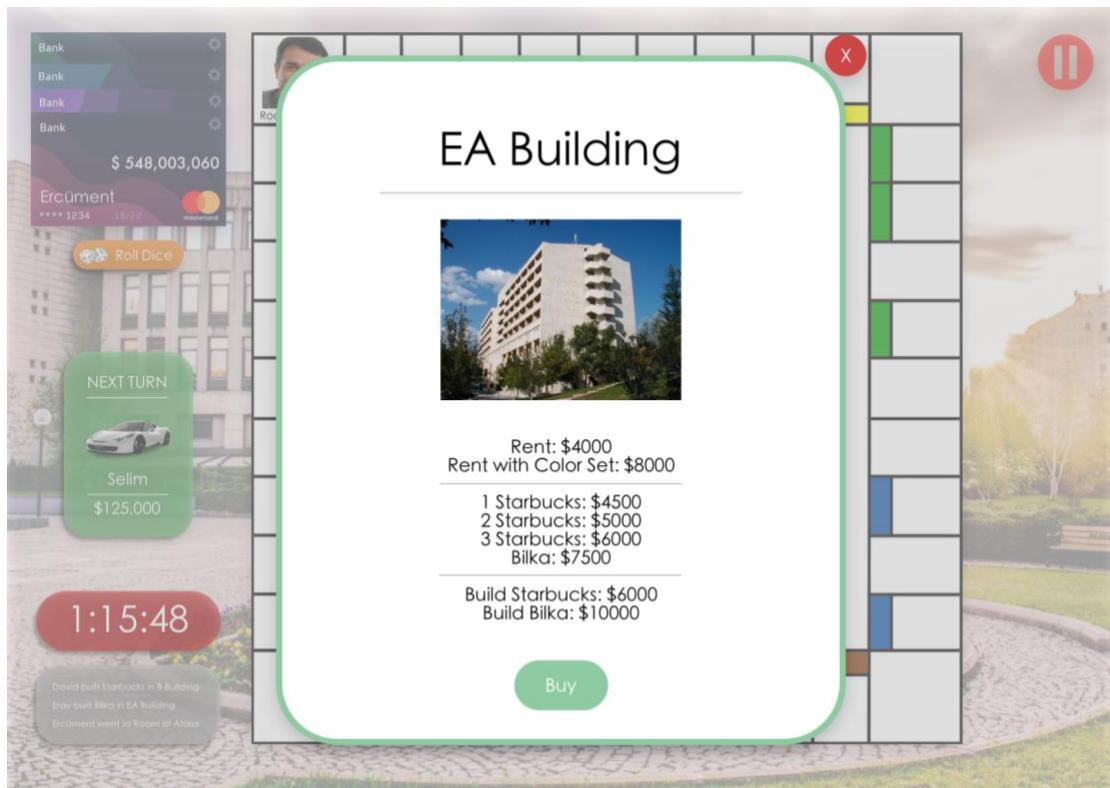


Figure 25. Building Card

If nobody owns the building that the player landed, this card appears. Building Cards have a green border outside them which is associated with the color of money. There is a picture of the building on the card. Below the picture of the building, there is information about the building.

#### 7.4.2.2.1.6. Building Cards (Player Bought All Buildings with the Same Color)

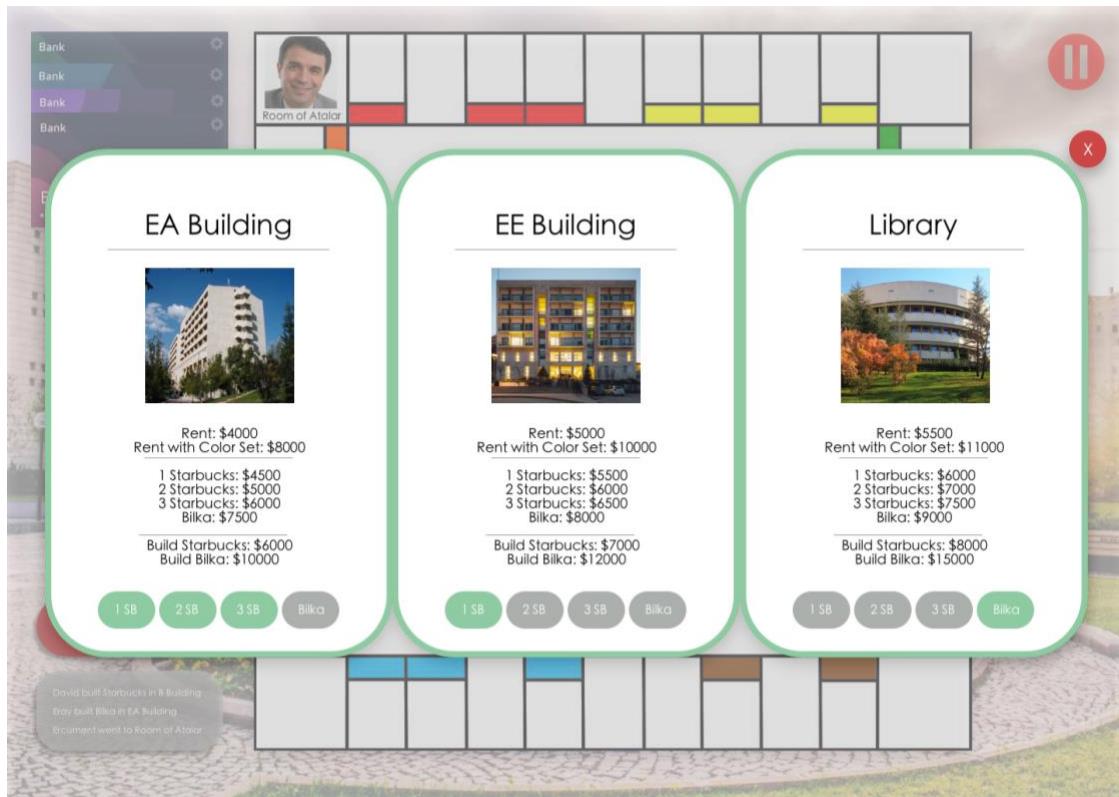


Figure 26. Building Cards

If the player owns all the buildings that have the same color, these cards of the buildings appear. The user can build Starbucks or Bilkas if they want to. The limit is at most 3 Starbucks and 1 Bilka in a single building. If a player already has, for example, 2 Starbucks, only 1 Starbucks can be bought. Other options are locked and indicated with grey.

## 7.4.2.2.2. Bilkent CS Mode

### 7.4.2.2.2.1. Game Screen

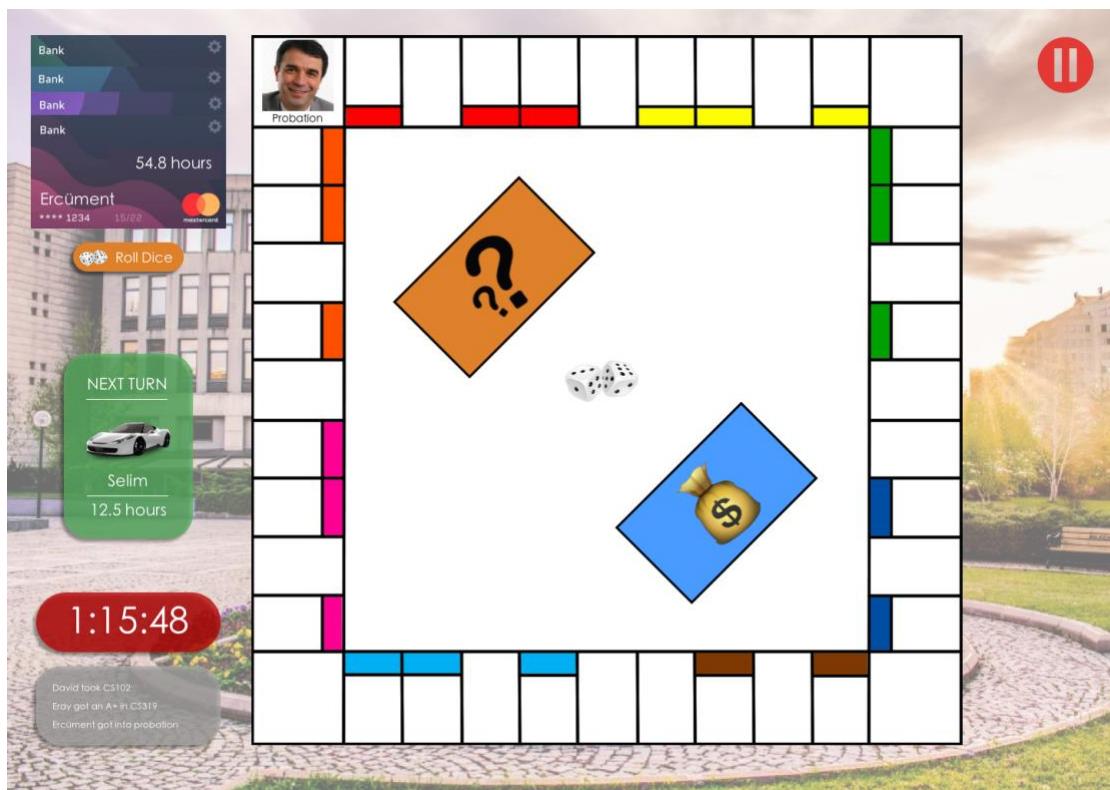


Figure 27. Game Screen of CS Mode

This board is very similar to the game board in Bilkent Buildings Mode. In this board, instead of buildings, there are courses on the regions of the board. Moreover, instead of money, the balances are in hours. Also, the history section at the bottom left is different according to the mode.

#### 7.4.2.2.2. Player Info

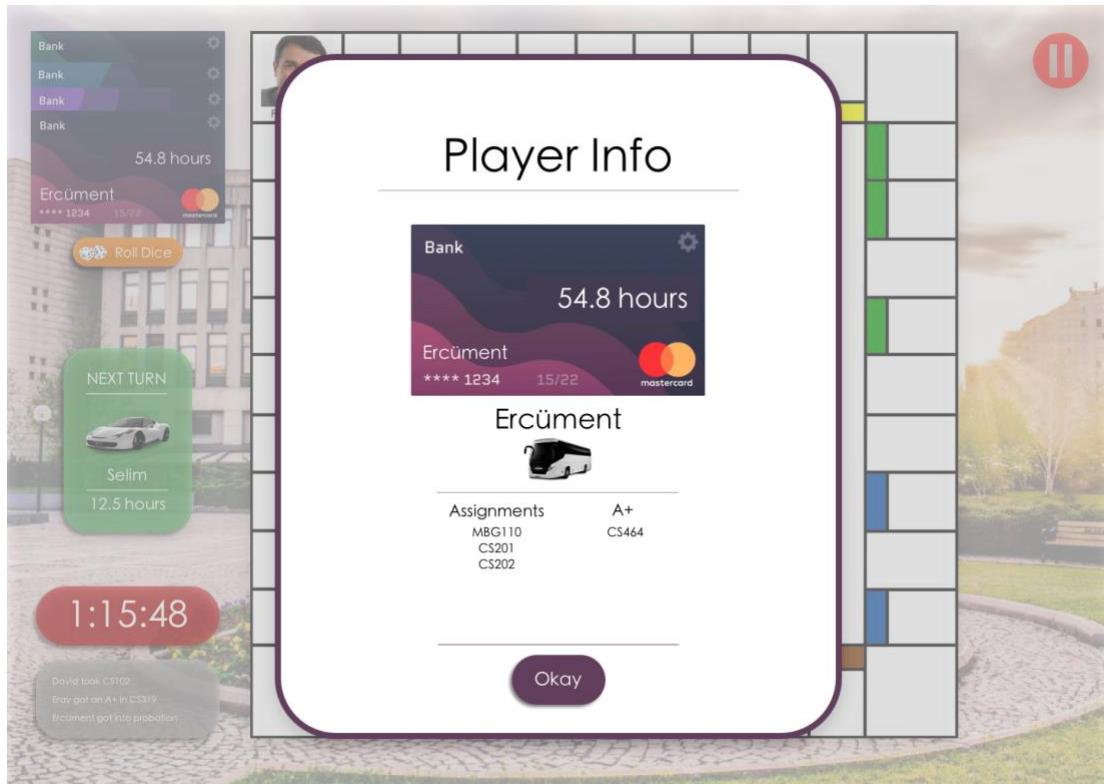


Figure 28. Player Info of CS Mode

By clicking the credit card of any player, information about that player can be displayed on the screen. Balance in hours, name, pawn and owned buildings of the desired player is present on this screen. The courses that the player did assignments or taken A+ are categorized as well. The player can go back to the game by clicking the “Okay” button.

#### 7.4.2.2.2.3. Chance Card

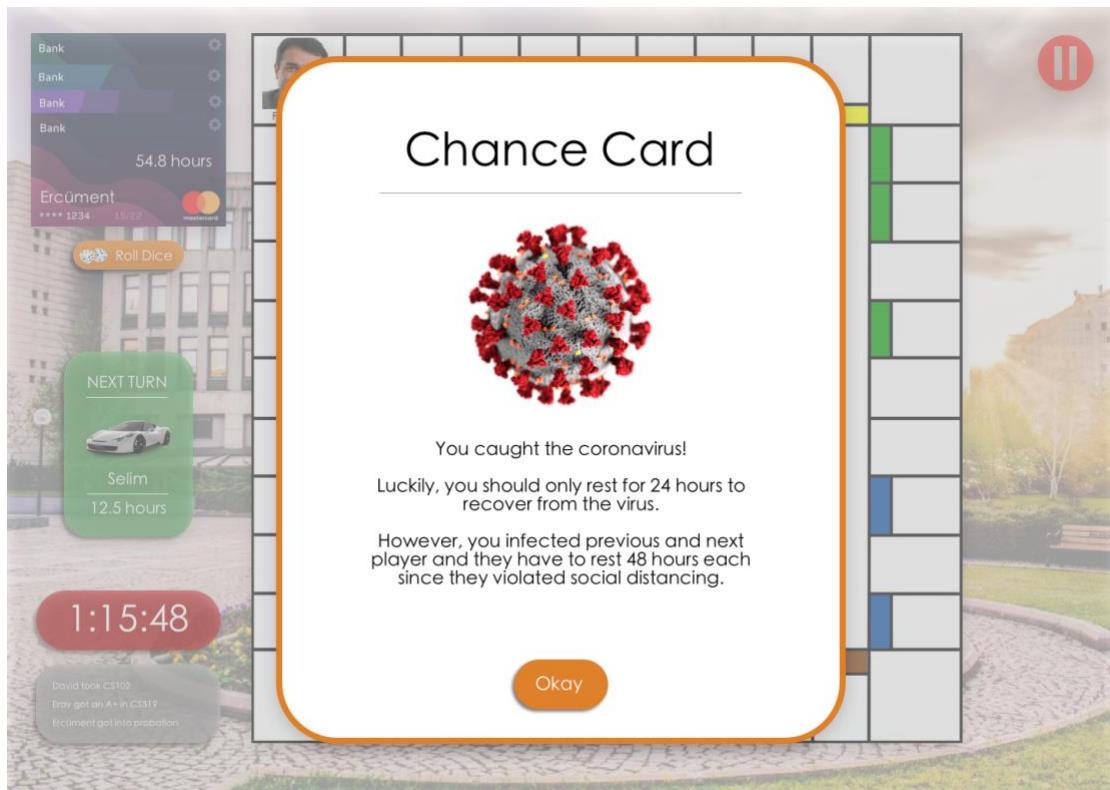


Figure 29. Chance Card of CS Mode

An example of a Chance Card is on the mockup above. There is an orange border outside of the card since Chance Cards are represented by the color orange. Each chance card has a picture that is associated with the content of the card. When the player clicks the “Okay” button, the chance card is executed according to its action.

#### 7.4.2.2.2.4. Rector's Whisper Card

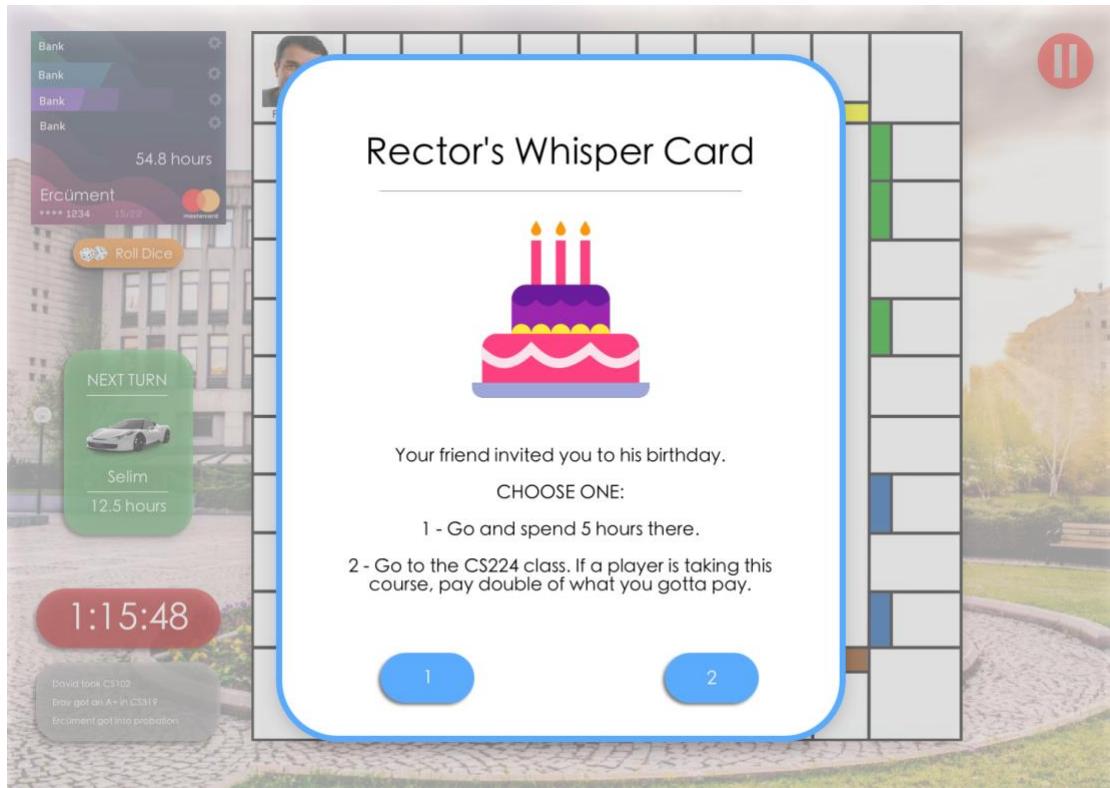


Figure 30. Rector's Whisper Card of CS Mode

An example of a Rector's Whisper Card is on the mockup above. There is a blue border outside of the card since Rector's Whisper Cards are represented by the color blue. Each Rector's Whisper Card has a picture that is associated with the content of the card. The player can choose one of the two options offered by the card. Both options have different actions. After selection, the card is executed.

#### 7.4.2.2.2.5. Course Card

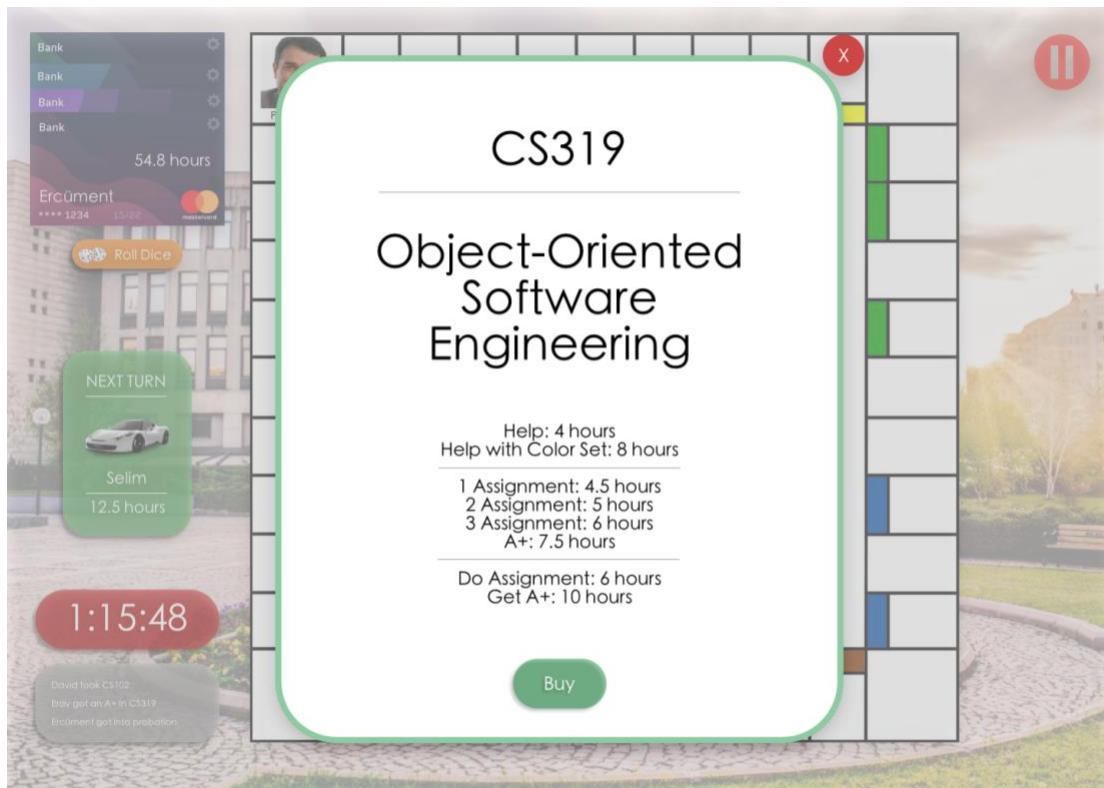


Figure 31. Course Card

If nobody takes the course that the player landed, this card appears. Course Cards have a green border outside them. There is the name of the course on the card. Below the name of the course, there is information about the course.

#### 7.4.2.2.6. Course Cards (Player Bought All Courses with the Same Color)

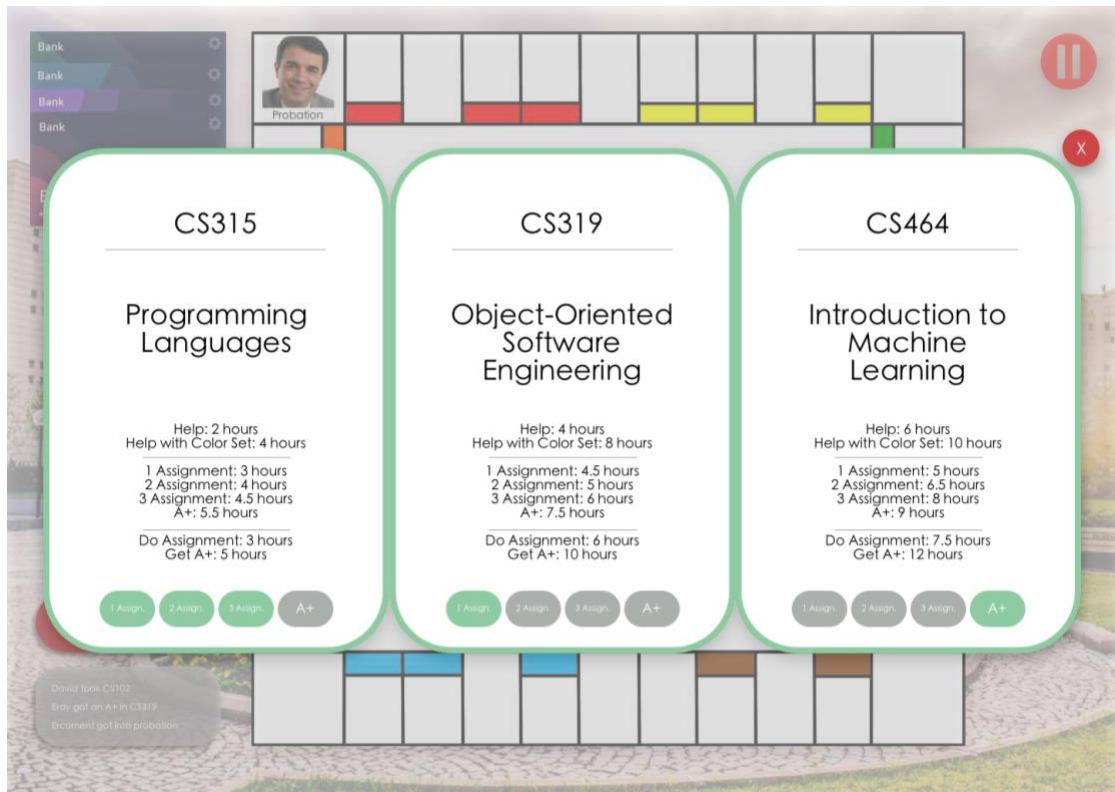


Figure 32. Course Cards

If the player takes all the courses that have the same color, these cards of the courses appear. The user can do assignments or get an A+ if they want to. The limit is at most 3 assignments and 1 A+ in a single course. If a player already did, for example, 2 assignments, only 1 assignment can be done. Other options are locked and indicated with grey.

### 7.4.2.2.3. Common Screens

#### 7.4.2.2.3.1. Pause Screen

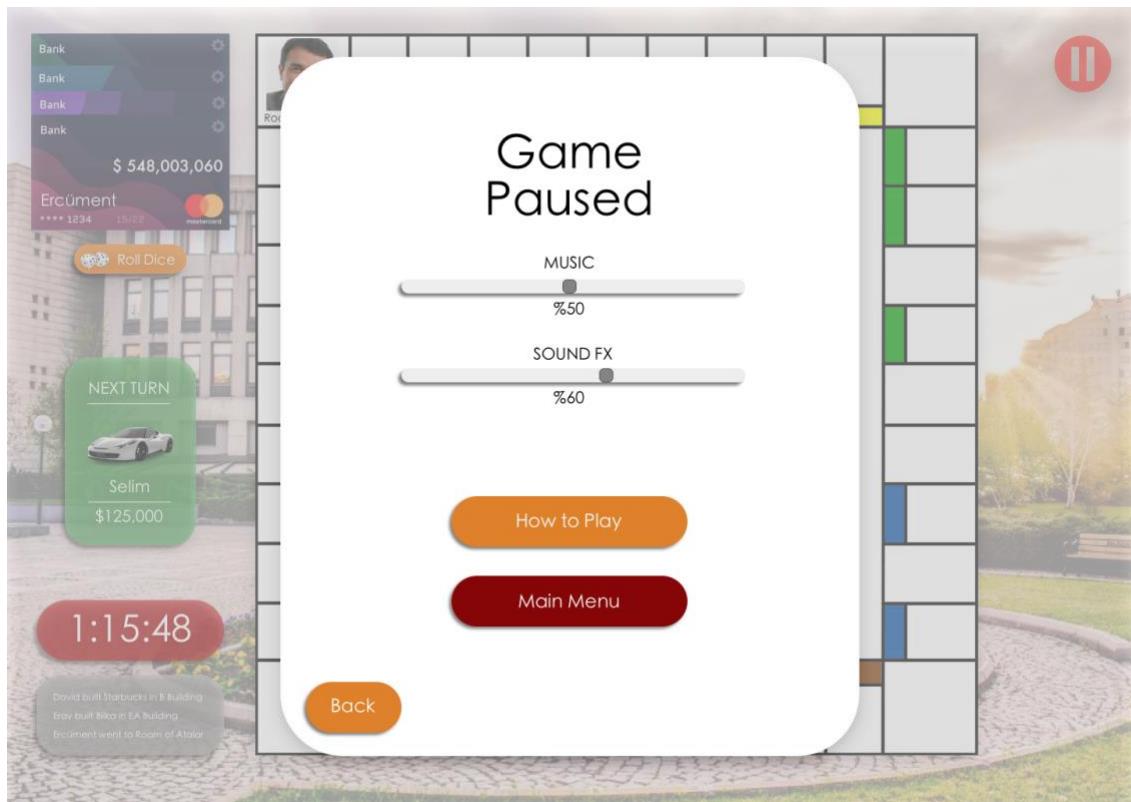


Figure 33. Pause Screen

When the game is paused, this screen pops up. The players can adjust the music and sound effect level on this screen. Or the players can open the "How to Play" screen and check out the game rules. As another option, the players can decide to go back to the main menu and end this session by clicking the "Main Menu" button. The game can be resumed via the "Back" button.

#### 7.4.2.2.3.2. Game Over - Bankruptcy

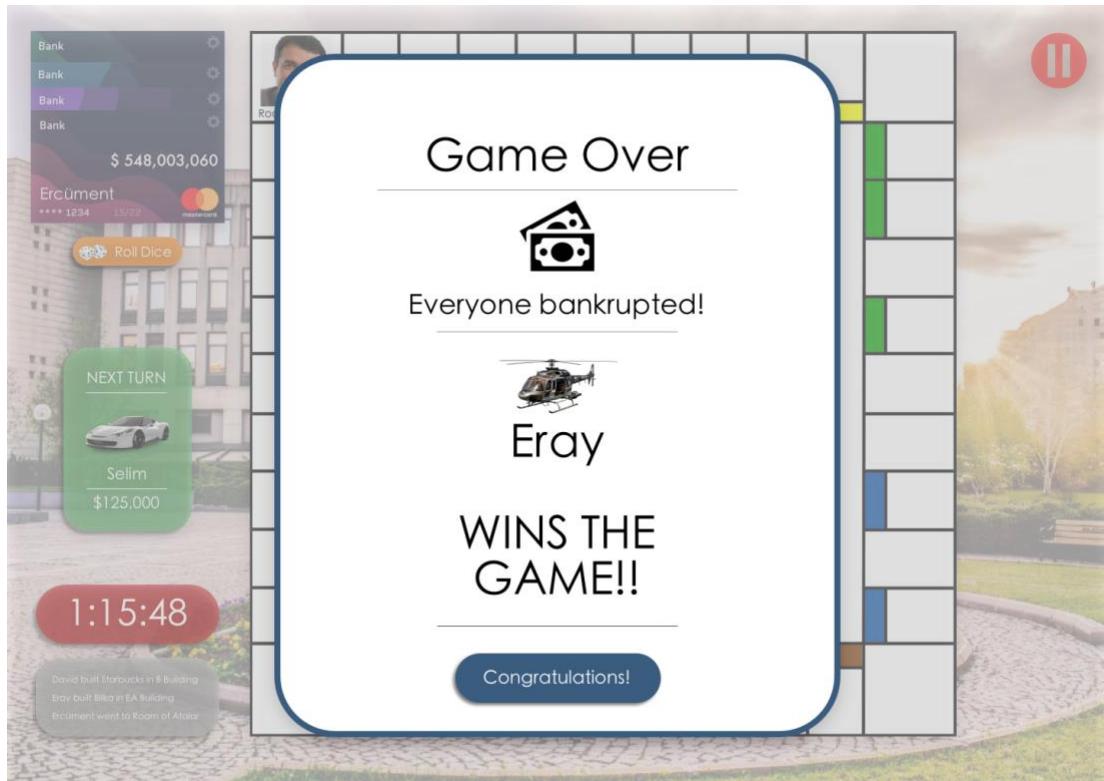


Figure 34. Game Over (Bankruptcy)

In Normal Mode, when everyone except one player bankrupts, the game is over. When the game is over, this screen pops up. The winner of the game is announced on this screen. The only option on this screen is to click the “Congratulations” button, which ends the session and takes it back to the main menu.

#### 7.4.2.2.3.3. Game Over - Time's Up

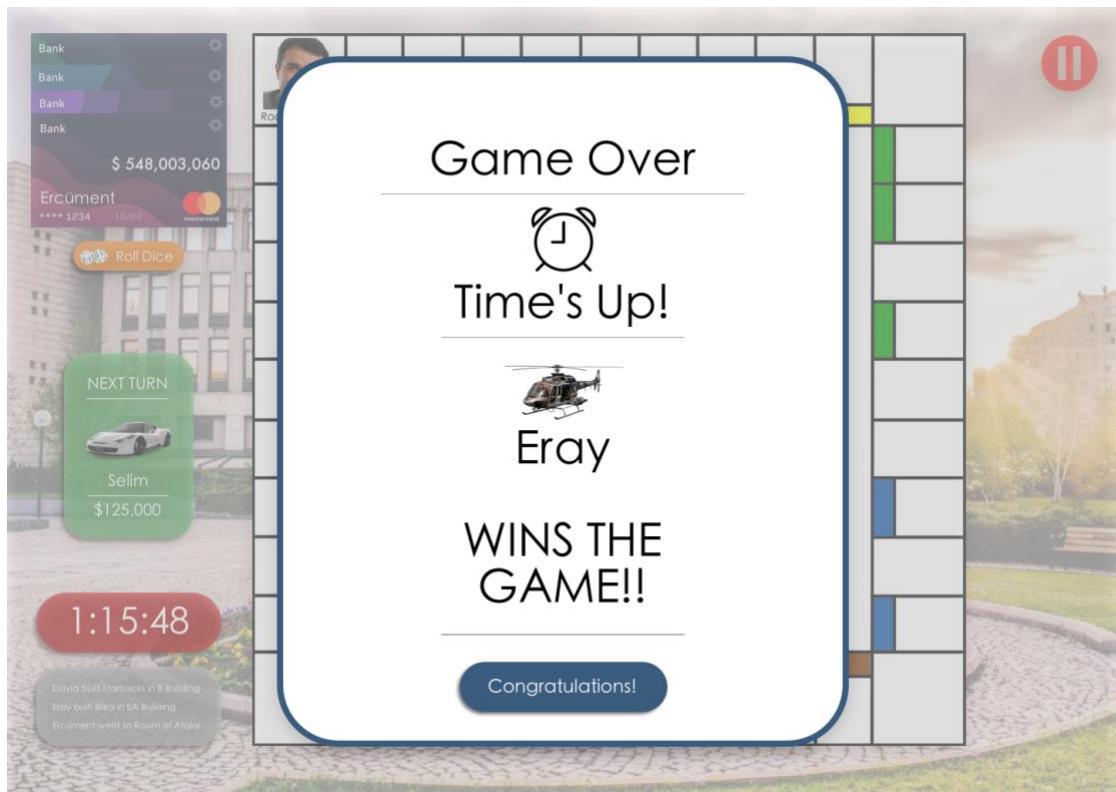


Figure 35. Game Over (Time's Up)

In Timed Mode, when the time runs out, the game is over. When the game is over, this screen pops up. The winner of the game is announced on this screen. The only option on this screen is to click the “Congratulations” button, which ends the session and takes it back to the main menu.

## 8. Conclusion

In the analysis report, "Bilpoly", which is the modified and improved version of the world-famous Monopoly game, was the subject. In the report, functional requirements, non-functional requirements and system models were discussed in detail and explained.

In the functional requirements section, many actions that the user will take after entering the game are explained. The options on the game screen are clearly expressed.

At Nonfunctional requirements section, describe aspects of the system that are not directly related to its functional behavior. It was also explained in the extendibility section how the game can gain extra features in its possible future.

In the System Models section, the system of the whole game was examined in 3 main sections:

- In the use case section, possible situations that may occur in the game with various scenarios are explained in detail with the help of diagrams. How the system works was shown with the activity diagrams. The functioning of the decision-making mechanism has been concretized in the sequence diagrams.
- In the Object and Class model, all classes of the game were written as much as possible, and it was tried to add attribute and methods as best we can.
- In the last part, information about the game was given with the help of screenshots and diagrams.

## 9. Improvement Summary

Considering the incoming feedback, changes and improvements made, the following were added to the second iteration:

- Additional functional requirements have been added: Some features developed differently from the classic Monopoly game have been explained without overly entering the game details.
- Attention was paid not to write the functional requirements based on UI.
- Performance has been added to the Nonfunctional Requirements section.

- Several diagrams have been updated according to feedback:
  - State diagram changed.
  - Parameters have been added to the sequence diagram.
  - Errors in the activity diagram have been fixed.
- Captions added to figures
- The game rules are explained in the overview section.

## 10. Glossary

**Land:** A land is a building such as MSSF, EA Building.

**Mortgage:** When a player does not have enough money to accomplish a certain task the player can sell the properties on a land and then take money from the bank in return.

**Property:** A property is either a Starbucks or Bilka.

**Café:** The cafes are the real restaurant in Bilkent neighborhood except Starbucks and Bilka.

## 11. References

- [1] "History of Monopoly," *Wikipedia*, 17-Oct-2020. [Online]. Available: [https://en.wikipedia.org/wiki/History\\_of\\_Monopoly](https://en.wikipedia.org/wiki/History_of_Monopoly). [Accessed: 22-Oct-2020].
- [2] "Monopoly Rules," *Monopoly Wiki*. [Online]. Available: [https://monopoly.fandom.com/wiki/Monopoly\\_Rules](https://monopoly.fandom.com/wiki/Monopoly_Rules). [Accessed: 22-Oct-2020].
- [3] C. Reporter, "The Importance of Music in Gaming," *Counteract*, 20-Dec-2019. [Online]. Available: <https://counteract.co/blog/the-importance-of-music-in-gaming/>. [Accessed: 23-Oct-2020].
- [4] Database-Control, "World Class Standards, Local Expertise," *Summit Recruitment & Search*, 10-Apr-2020. [Online]. Available: <https://www.summitrecruitment-search.com/classic-board-game-play-monopoly/>. [Accessed: 29-Oct-2020].
- [5] D. Tapaniya, "Colors in UI Design-Theory, Psychology & Practice," *Medium*, 27-May-2019. [Online]. Available: <https://medium.com/iconscout/colors-in-ui-design-theory-psychology-practice-f6d6a5e6e04d>. [Accessed: 29-Oct-2020].
- [6] C. Vizcarra, "All you need to know about colors in UI Design-theory & practice," *Medium*, 26-May-2019. [Online]. Available: <https://uxdesign.cc/all-you-need-to-know-about-colors-in-ui-design-theory-practice-235179712522>. [Accessed: 29-Oct-2020].