Bilkent University

Department of Computer Engineering

CS319 - Object-Oriented Software Engineering

Term Project - Analysis Report
Iteration 2

Project Name: Settlers of Catan

Group No: 3G

Group Members: Mehmet Tolga Tomris
Atakan Arslan
Fatih Çakır
Oğuzhan Dere
Zeynep Berfin Gökalp

Table of Contents

1.Introduction

The **Settlers of Catan** is a famous strategy board game developed and designed by Klaus Terber [1]. The main objective of the game is collecting resources and use them to build roads, settlements or cities to achieve victory. The fun and excitement of the game is, it has features such as, its board is changeable, which makes the new game different from the previous one. The original board game does not have enough various features and it has a complex structure, we decided to add new features, increase the functionality and make it user-friendly and more preferable by implementing the desktop version of the game. In addition to the existing structure of the game, our group decided to implement additional features that will work with the actual game and also advance its object-oriented design.

In our primary design plan, our group is considering to add the following features:

- Sound/Audio Options
- How to Play Option
- Angel Mode
- Judge Development Card

The structure of the game is based on random hexagon grids. The board will vary as starting a new game.

For *Angel Mode*, instead of a robber, there will be an iconic angel who will reward the player as s/he rolls the dice as 7 when s/he is around our settlements and cities.

For *Judge Development Card,* that card collects all resource cards from players and distributes them equally. If the total number of cards cannot be divided by 4, extra cards that are chosen randomly will be deleted.

2.Overview

2.1 Game Grid

The standard game grid consists of 19 hexes, 6 sea frames that are aligned as shown in Figure 1. All of the game elements will be placed on the grid. Roads, settlements and number tokens will be placed on edges. The resource and the number of each hex is randomly determined.

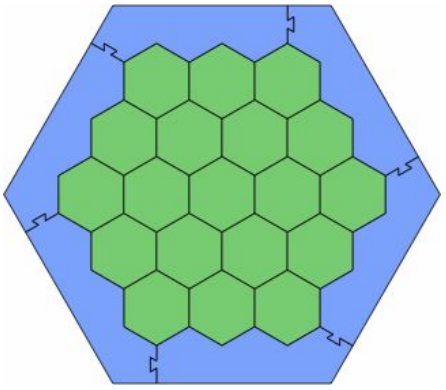

Figure 1

2.2 Settlements

No matter whose turn it is, if a terrain hex produces resources, the player can receive a resource card for each settlement. Settlements require Brick, Lumber, Wool or Grain. Also, each settlement should connect to at least one road. Each settlement is 1 victory point. The **Distance Rule** should be applied when placing the settlements.

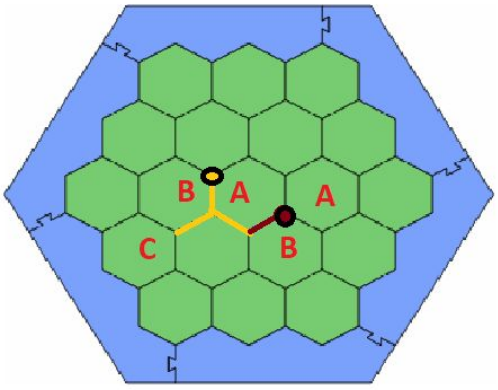

Figure 2

Distance Rule: Any settlement that is placed cannot be less than two “edges” away from another settlement or city.

Figure 2 shows the **Distance Rule** as the yellow player wants to build a settlement. The settlements shown as “A” are already played. Thus, the yellow player can not build on the intersections marked with “B.” The orange player can only build his/her settlement at the intersection “C.”

2.3 Cities

The cities can only be established by upgrading one of the settlements. It requires 3 Ore and 2 Grain. They produce twice as many resources as settlements. The player can get two Resource Cards for an adjacent terrain hex which produces resources.

2.4 Roads

Road requires Brick and Lumber. They should be connected to one of the roads, settlements or cities the player has. Only one road can be built on a given path(**). The first player who builds a continuous road having at least 5 roads, gains the Special Card, the Longest Road. If another player achieves longer than the owner of this special card, the person earns the card.

2.5 Resource Cards

There are 5 different Resource Cards, which are earned after the player rolls the dice. All players who have settlements or cities on the hexes determined by the result of the dice, take the resource cards of that hexes.

2.6 Dice

There are 2 dice. Each dice has 6 faces through 1 to 6. The dice must be rolled for resource production. The sum of the dice determines which terrain hexes produce resources.

2.7 Coast

If the player has a terrain hex that borders on the sea, it is called a “coast.” The player can build a road along the coast and also build settlements and cities too. Coastal sites allows the player to make Maritime Trade(Sec 2.11.2).

2.8 Buying Development Cards

The development cards require Ore, Wool, and Grain. If the player decides to buy a development card, he/she draws the card from the shuffled deck. There are three different cards, Knight, Progress and Victory Frame.

2.9 Special Cards

2.9.1 Rolling 7

If the player rolls the dice a “7”, no player gets a resource. All players who have resource cards more than 7, should return half of their cards to the bank. Afterward, the player must move the robber to the desert or any other hex. The owner of the settlement or city which is adjacent to that hex gives 1 random card to the player. If there is more than one opponent that has an adjacent settlement or city, the player chooses who to rob.

2.9.2 Playing Development Cards

During any time of his/her turn, the player can play 1 Development Card.
Knight Cards: After playing the Knight card, the player moves the robber. The first player who gets the 3 of Knight cards has a special card called “Largest Army” which is 2 VP. If another player has more Knight cards than the current holder, the player takes the Largest Army card. They are red-framed.

Progress Cards: There are 3 different Progress Cards which are green framed.

- **Road Building:** If the player plays this card, he/she can place 2 roads on the board.
- **Year of Plenty:** If the player plays this card, he/she get any 2 Resource Cards from the deck.
- **Monopoly:** If the player plays this card, every other player should give all of their Resource Cards that the player demands.

Victory Cards: It is played when the player has 10 VP. They are yellow framed.

2.10 Game Modes

2.10.1 Robber Mode

The robber will be placed in dessert. It is moved only by rolling a “7” or playing the “Knight Card”. If the robber is placed to any other terrain hex, it restricts that hex from producing resources as long as it remains in that hex.

2.10.2 Angel Mode

The angel will be placed in the grass. It is moved only by rolling a “7”. If the angel is placed to any other terrain hex, it duplicates the resource of that hex as long as it remains in that hex.

2.11 Win Conditions

In order to win, the player must have 10 victory points or more. Each settlement is worth 1 VP, each city is worth 2 VP, and each road is worth 0 VP.

2.12 Trade

Trade can be done to get the required Resource Cards between players or with the bank. There are two types of trade below: domestic trade and maritime trade.

2.12.1 Domestic Trade

The players can trade their Resource Cards with their opponents. The player calls for their needs. Also, opponents can make counteroffers. The ratio is determined by players.

2.12.2 Maritime Trade

The player can trade with the bank by putting 4 exact Resource Cards back and get any 1 Resource Card. If the player has a settlement or city on a dock, he/she can trade with the bank at a 3:1 ratio or at 2:1.

2.13 Settings

The user is able to modify audio settings about the game including the background music and volume of the sound effects.

3. Functional Requirements

3.1 Additional Requirements

Since the game is single-player, artificial intelligence is added to the functional requirements.

3.2 Modes

3.2.1 Robber Mode

The user will access this screen from the Choose mode screen using the “Robber Mode” button. In this mode, the user plays the standard game having the feature of Robber.

3.2.2 Angel Mode

The user will access this screen from the Choose mode screen using the “Angel Mode” button. In this mode, the user plays the standard game having the feature of Angel.

3.3 New Game

The player will access this screen from the home screen, pressing the “New Game” button, which the user can set to start a new game.

3.4 Load Game

The user will access this screen from the home screen, pressing the “Load Game” button, which the user will be asked to select a game to be played.

3.5 How to Play

The player will access this screen from the home screen using the “How to Play” button. This option presents the rules, controls, tips of the game.
Goal of the Game: Description of the aim of the game
Game Rules: Distance rules, resource production, trade, etc.
How to Gain Victory Points: Description of win conditions
Controls: Description of the input controls
Some Tips: Tips of the game

3.6 Settings

The player will access this screen from the home screen using the “Settings” button. On this screen, the player is free to change the sound level of the game.

3.7 Credits

The player will access the screen using the “Credits” button. Developers of the game and the contact information of each developer are listed in addition to the GitHub link.

3.8 Quit Game

The user will access this function from the home screen using the “Quit Game” button. By clicking this button, the user will be asked if he/she is sure to quit the game. After the user selects “Yes”, he/she quits the game immediately. If the selection is “No”, the user accesses the home screen.

3.9 Artificial Intelligence

Our game is a single player game. Therefore, it has artificial intelligence for non-player characters. The non-player characters will be bots and they will be controlled through artificial intelligence.

4. Non-Functional Requirements

4.1 User-Friendly Interface

The Settlers of Catan is designed for 10+, it should satisfy the simplicity of user interface since its also aimed for children too. The actions for playing the cards should be easy to perform. The player should not need to place the roads and settlements pixel-perfect, they should align by themselves. Furthermore, the simplicity of a user-friendly interface also prevents users from making mistakes.

4.2 Performance

In terms of performance, it is important to have a simple and straightforward Settlers of Catan game. The test of win conditions should run in an acceptable time after collecting the points for every turn. Also, rolling the dice and distribution of cards should be fast as well for giving more time to the user so that s\he can concentrate on the game, rather than complex game constraints.

4.3 Extendibility

Settlers of Catan should be extendible since the game is open for new features such as Angel and Robber Modes. Also, the game grid changes as the new game start thus, the algorithms should run and satisfy both of the conditions with different game grids and components. The new components should be changed and added to the game easily.

4.4 Reliability

Game progress of the players will be kept, thus, players can load the game and continue from where they left. The records of the players will be kept in the database. These data(game progresses and records) would not be lost due to a system failure or a power loss.

4.5 Additional Requirements

Since the game records and progress will be kept on database, reliability of the game has changed.

5. System Models

5.1 Use Case Model

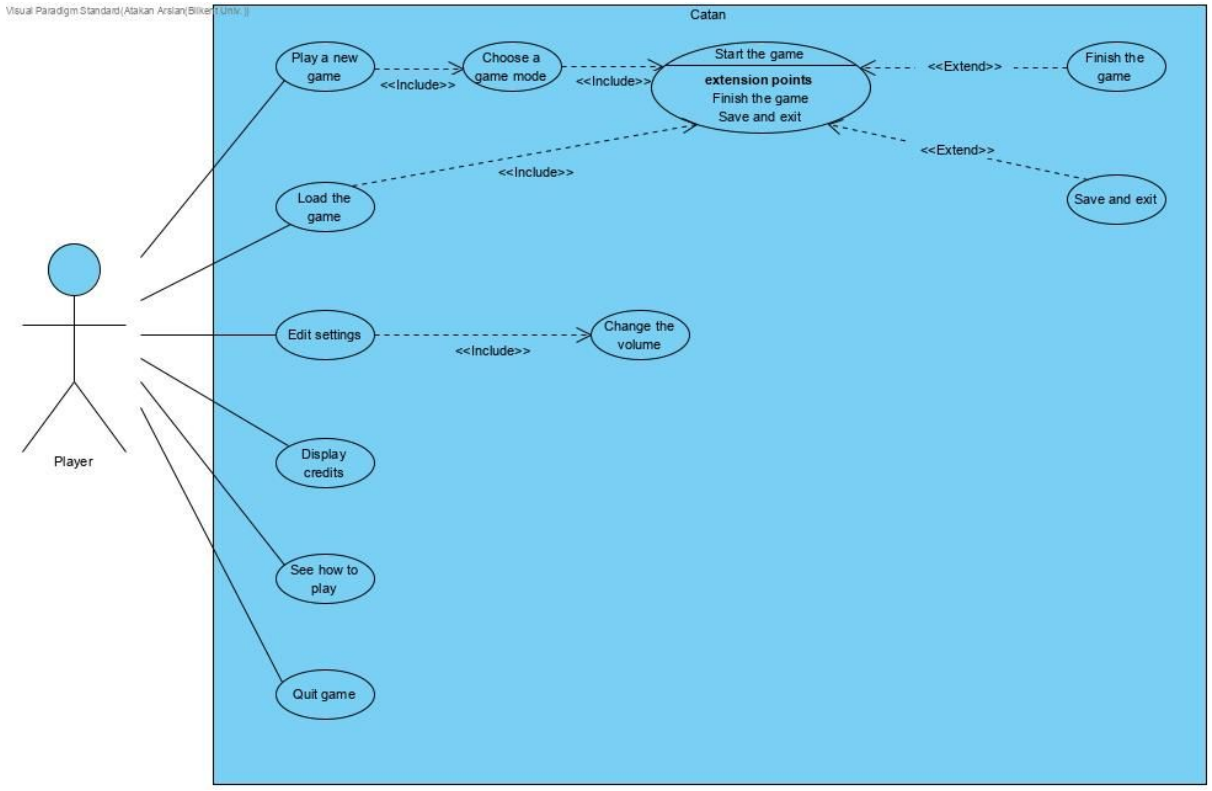

Figure 3: Use Case Diagram of Main Menu

Use Case #1

1. ***Use Case:*** Play a new game
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Conditions:***

1.Player reaches 10 victory points

2.Player selects save & quit option

5. ***Success Scenario Event Flows:***

1.Player chooses one of the game modes

2.Player starts the game

3.Player places the settlements and cities to gain victory points

4.First player who reaches 10 victory points wins the game

5.The game displays a success message for the winner

6.The game returns to the main menu

6. ***Alternative Event Flows:***

1. Player wants to return to the main menu
2. Player selects the back option and the system returns to the main menu

Use Case #2

1. ***Use Case:*** Load the game
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Conditions:***

a.Player reaches 10 victory points

b.Player selects save & quit option

5. ***Success Scenario Event Flows:***

1.Player chooses to load the game

2.Player resumes the game from where s/he left

3.Player places the settlements and cities to gain victory points

4.First player who reaches 10 victory points wins the game

5.The game displays a success message for the winner

6.The game returns to the main menu

6. ***Alternative Event Flows:***

1. Player wants to return to the main menu

2. Player selects the back option and the system returns to the main menu.

Use Case #3

1. ***Use Case:*** Edit settings
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Condition:*** The game performs the changes of settings and returns to the main menu.
5. ***Success Scenario Event Flows:***

1.The game displays the settings option

2.Player changes the volume of the sound

3.Player chooses to return to the main menu

4.The game saves the changes of the settings and returns to the main menu

Use Case #4

1. ***Use Case:*** Display credits
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Condition:*** The game returns to the main menu
5. ***Success Scenario Event Flows:***

1.The game displays the credits

2.Player sees the credits information

3.Player chooses to return to the main menu

4.The game returns to the main menu

Use Case #5

1. ***Use Case:*** See how to play
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Condition:*** The game returns to the main menu
5. ***Success Scenario Event Flows:***

1.The game displays the rules of the Catan

2.Player sees the rules of the Catan

3.Player chooses to return to the main menu.

4.The game returns to the main menu

Use Case #6

1. ***Use Case:*** Quit game
2. ***Participating actor:*** Player
3. ***Entry Condition:*** Player must be in the main menu
4. ***Exit Condition:*** The player decides to not quit the game and returns to the main menu
5. ***Success Scenario Event Flows:***

    1.Player chooses to quit the game

    2.The game is successfully shut down

6. ***Alternative Event Flows:***

    1.Player decides to not quit the game

    2. Player selects the back option

    3.The game returns to the main menu

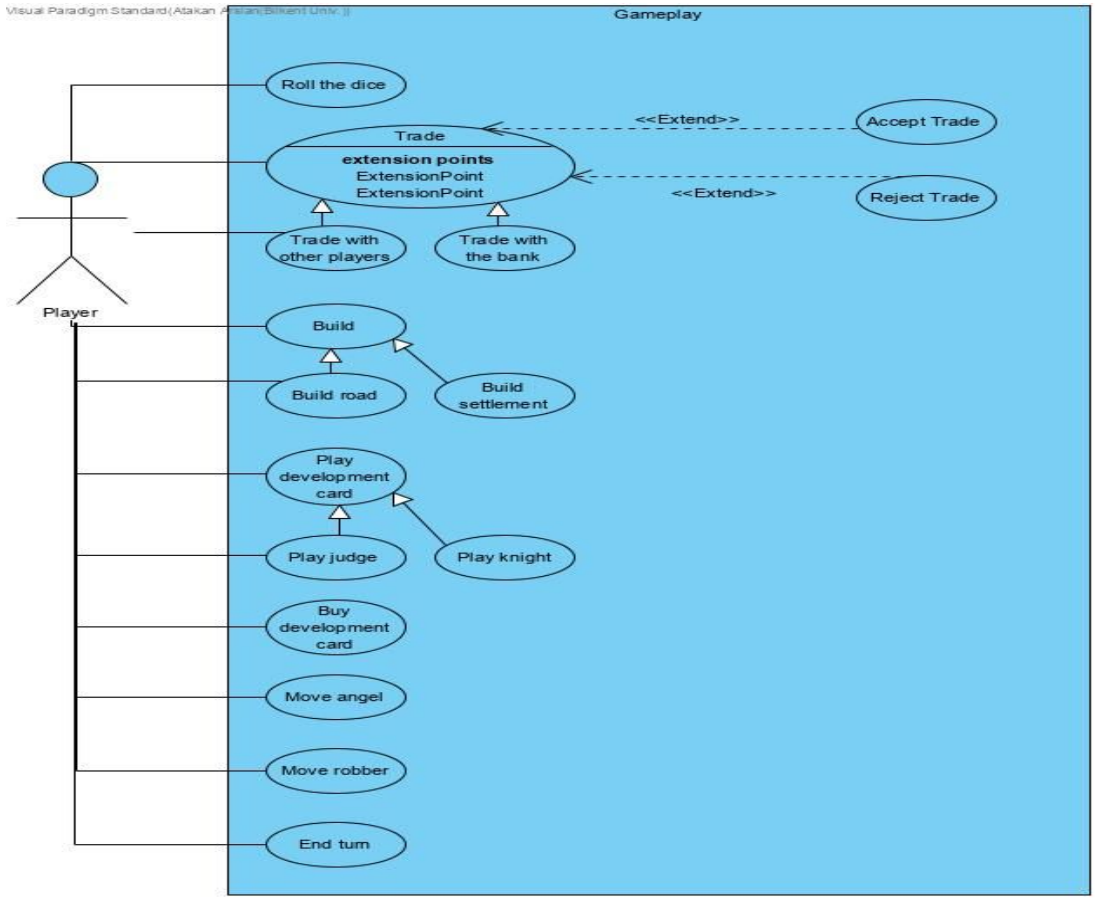

Figure 4: Use Case Diagram of Gameplay

Use Case #1

1. ***Use Case:*** Roll the dice
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Conditions:***

1.Game engine rolls the dice.

5. ***Success Scenario Event Flows:***

1.Player chooses to roll the dice.

2.Game engine rolls the dice.

3.Game displays the result of the dice.

Use Case #2

1. ***Use Case:*** Trade
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Condition:***
    1. Player accepts trade.
    2. Player rejects trade.
5. ***Success Scenario Event Flows:***
    1. Player offers the trade.
    2. Player negotiates about the offer.
    3. Player accept the trade.
    4. The trade is completed.
6. ***Alternative Event Flows:***
    1. Player offers the trade.
    2. Player negotiates about the offer.
    3. Player rejects the trade.
    4. Trade is canceled.

Use Case #3

1. ***Use Case:*** Build
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Condition:*** The player decides to not build anything and returns the game screen.
5. ***Success Scenario Event Flows:***

    1.Player decides to build a road or settlement.

    2.Player chooses to build a road or settlement.

    3.Player gives the required resources.

    4.Player chooses where to build.

    5.The building is placed on the board.

6. ***Alternative Event Flows:***

    1.Player decides to build a road or settlement.

    2.Player chooses to build a road or settlement.

    3.Player does not have the required resources.

    4.Building process cannot be completed.

Use Case #4

1. ***Use Case:*** Play development card
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Condition:*** The player decides to not play any development card and returns the game screen.
5. ***Success Scenario Event Flows:***

    1.Player decides to play a development card.

    2.Player chooses which card to play.

    3.The card is played.

Use Case #5

1. ***Use Case:*** Buy development card
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Condition:*** The player decides to not buy any development card and returns the game screen.
5. ***Success Scenario Event Flows:***

    1.Player decides to buy a development card.

    2.Player should have the required resources.

    3.Game randomly gives a development card to the player.

6. ***Alternative Event Flows:***

    1.Player decides to buy a development card.

    2.Player does not have the required resources.

    3.The buying process is canceled.

Use Case #6

1. ***Use Case:*** Move angel
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game, the result of the dice must be 7 and it should be the player's turn.
4. ***Exit Condition:*** The result of the dice is 7 but the player does not want to move the angel.
5. ***Success Scenario Event Flows:***

    1.Player rolls the dice.

    2.The result of the dice is 7.

    3.Player decides where to move the angel

    4.The angel is moved.

6. ***Alternative Event Flows:***

    1.Player rolls the dice.

    2.The result of the dice is not 7.

    3.Player cannot move the angel.

Use Case #7

1. ***Use Case:*** Move robber
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game, the result of the dice must be 7 and it should be the player's turn.
4. ***Exit Condition:*** The result of the dice is 7 but the player does not want to move the robber.
5. ***Success Scenario Event Flows:***

    1.Player rolls the dice.

    2.The result of the dice is 7.

    3.Player decides where to move the robber

    4.The robber is moved.

6. ***Alternative Event Flows:***

    1.Player rolls the dice.

    2.The result of the dice is not 7.

    3. Player cannot move the robber.

Use Case #8

1. ***Use Case:*** End turn
2. ***Participating actor:*** Player
3. ***Entry Condition:*** The player must be in the game and it should be the player's turn.
4. ***Exit Condition:*** The player decides not to end his/her turn.
5. ***Success Scenario Event Flows:***

    1.Player makes his moves in his turn.

    2.Player decides to end his/her turn.

    3.The turn of the player is ended.

5.2 Dynamic Models

5.2.1 State Diagrams

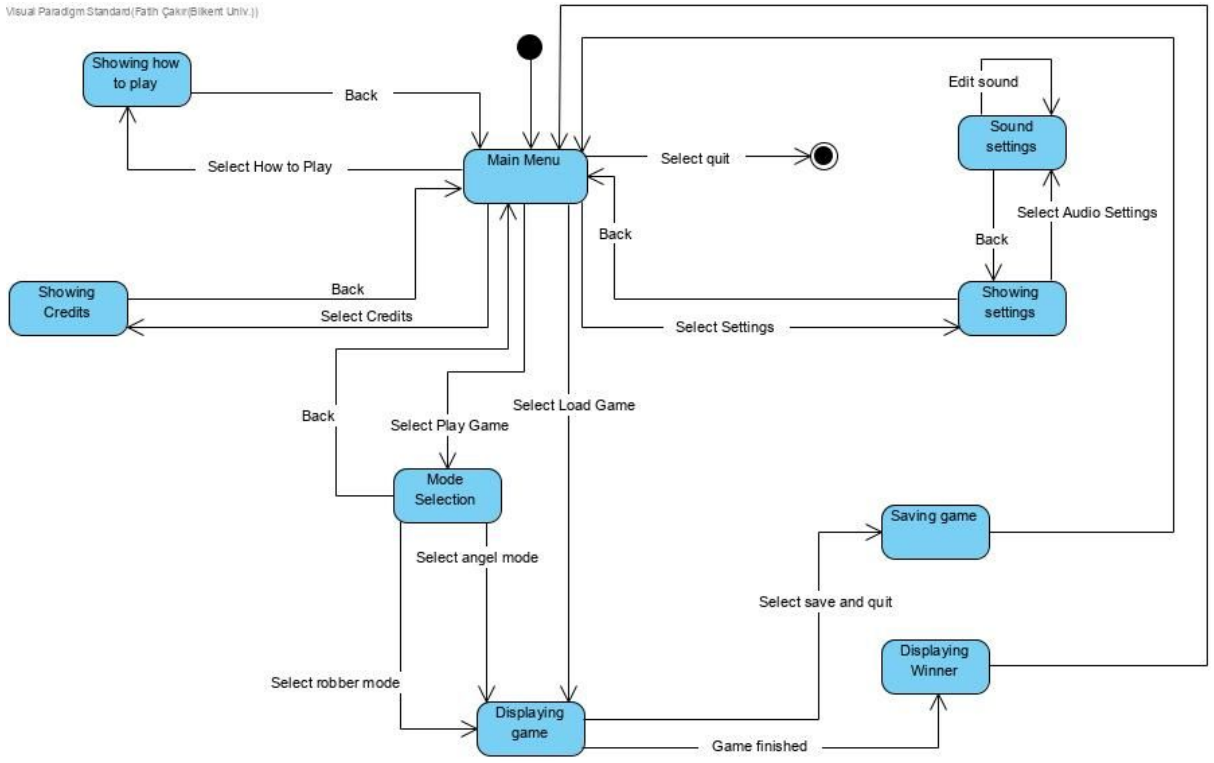

Figure 5: State Diagram of Main Menu

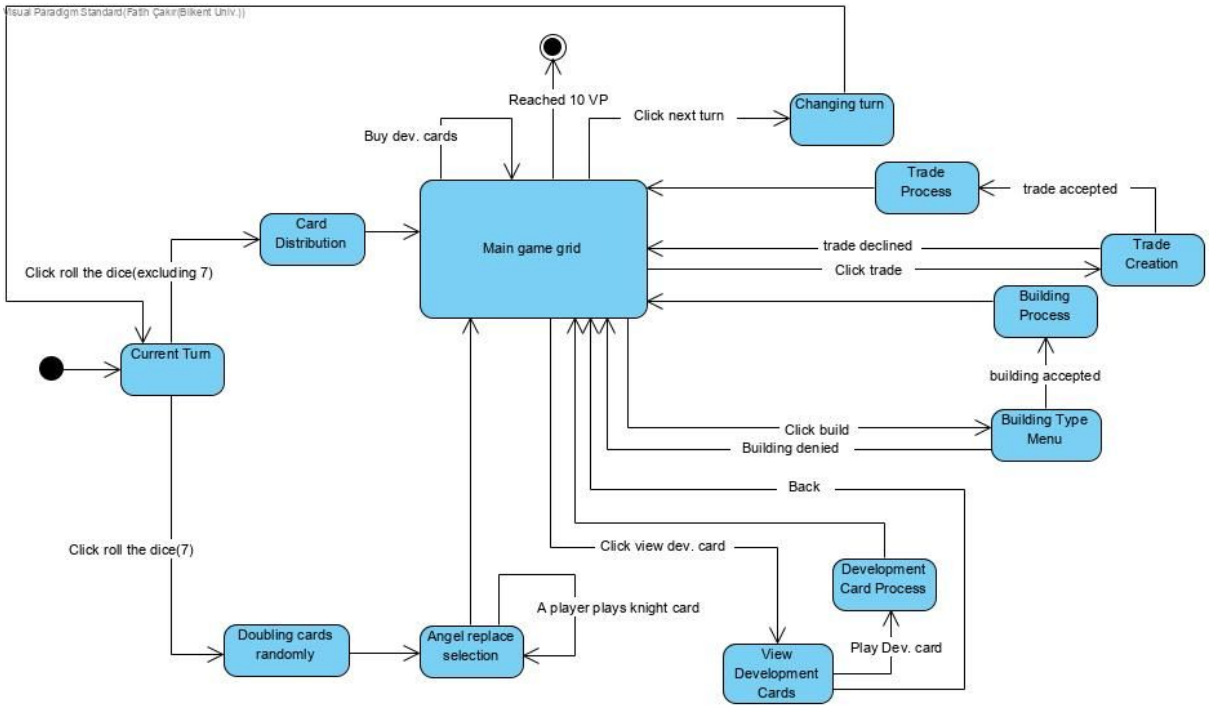

Figure 6: State Diagram of Angel Mode Gameplay

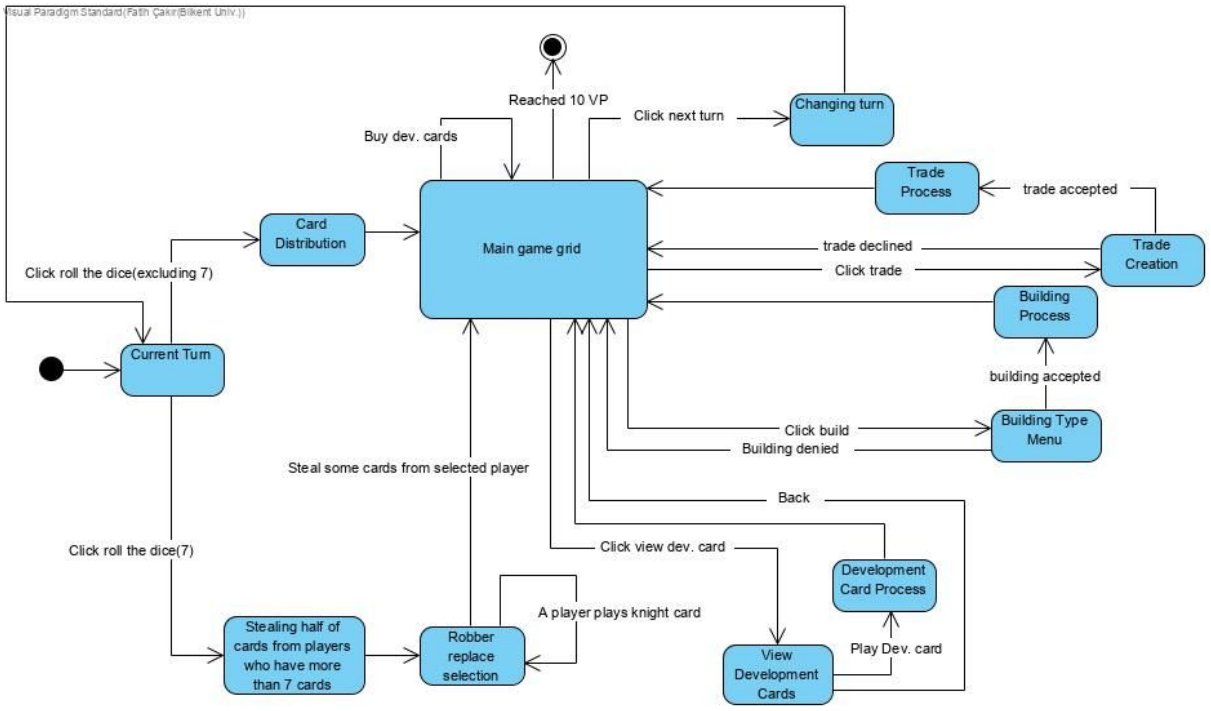

Figure 7: State Diagram of Robber Mode Gameplay

5.2.2 Sequence Diagrams

5.2.2.1 Sequence Diagrams of Gameplay

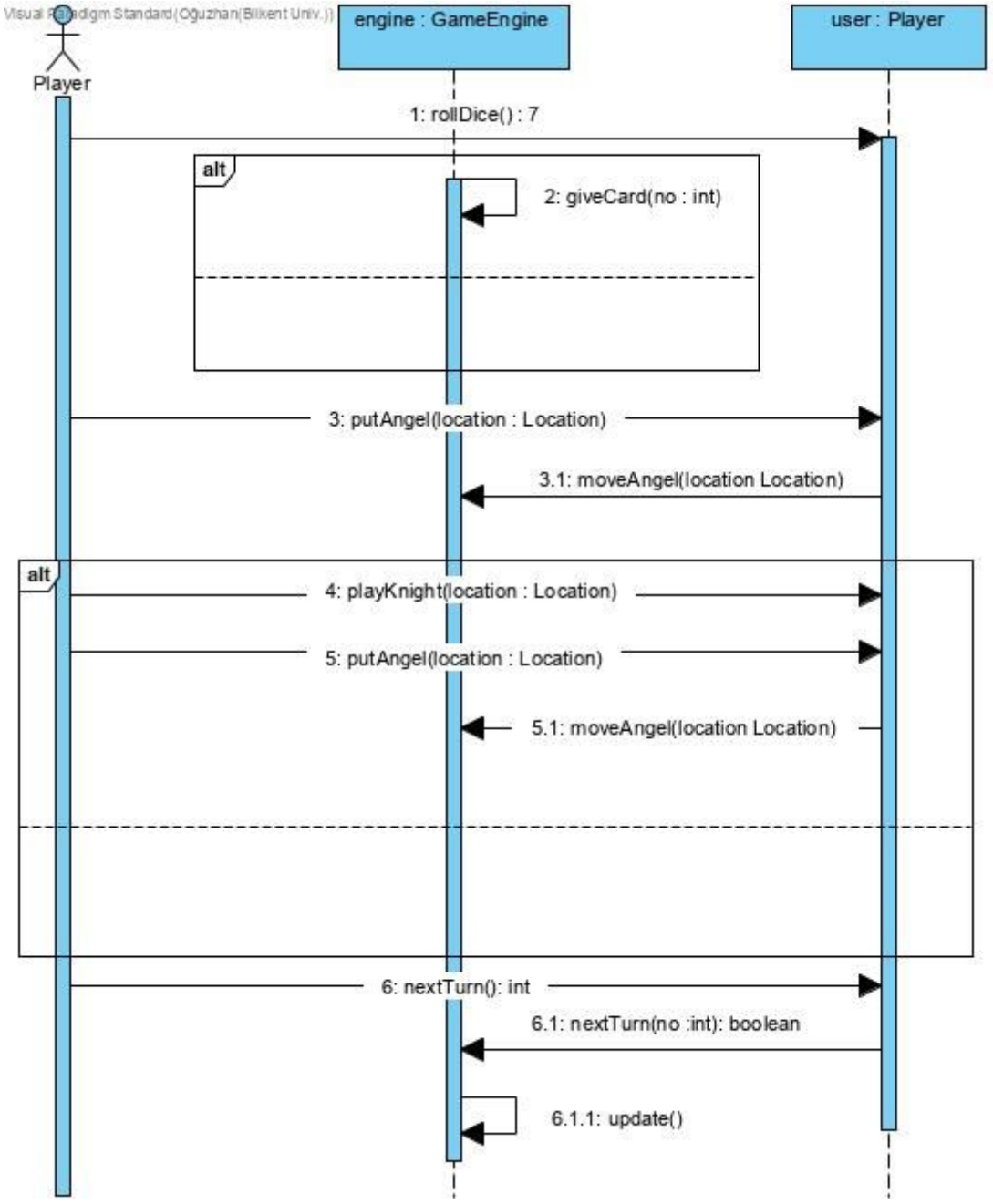

Figure 7: Sequence Diagram of Rolling 7 in Angel Mode

If the total number of the dice is 7, the game engine gives random resource cards to the players who have less than 4 cards to duplicate. Then the player puts the angel to a valid location. If any player who has Knight Card wants to play his/her development card, he/she put the angel to a valid location and the game engine gives him/her a random resource cards. After all move is done, the player press Next Button and the game engine pass the next turn the game.

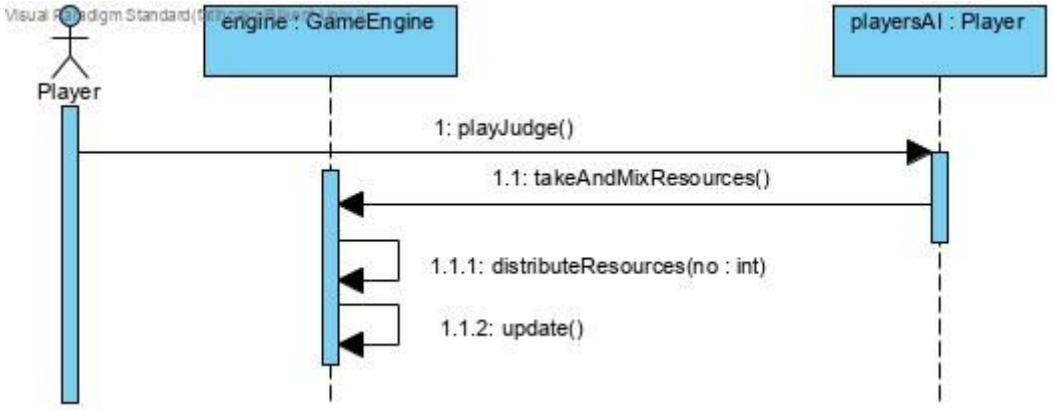

Figure 8: Sequence Diagram of Playing Judge Development Card

If any player plays his/her Judge Card, the game engine takes all cards and mix them. Then the engine distributes to all players again and updates the game.

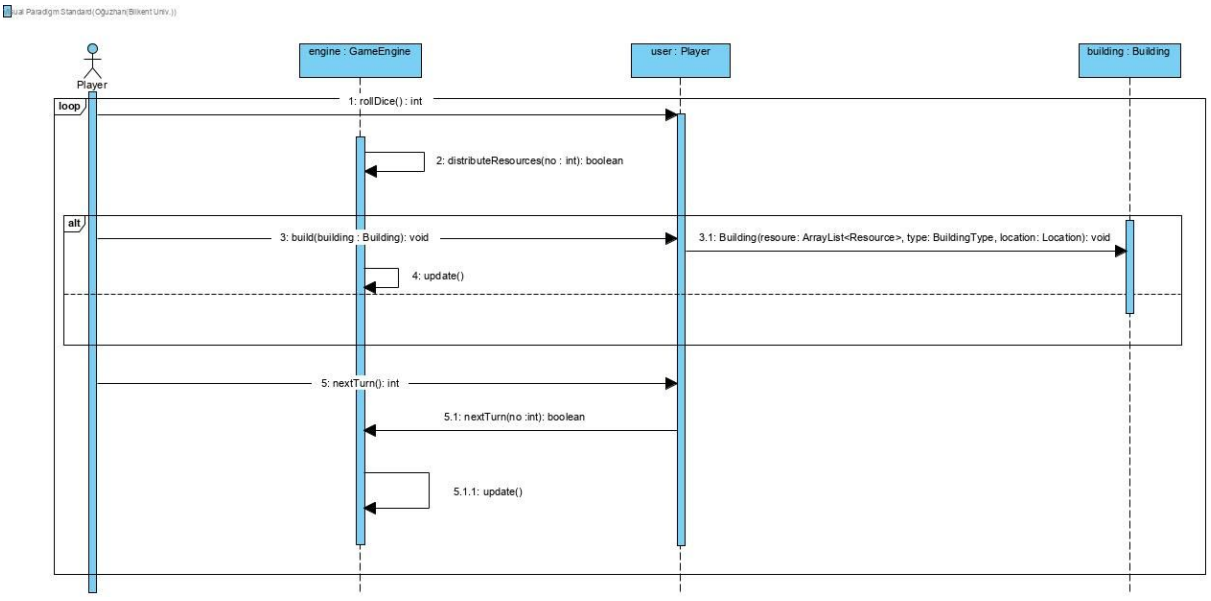

Figure 9: Sequence Diagram of Card Distribution After Rolling

After player's rolling the dice, game engine distributes the resource cards checking each player considering the player should get or not according to the total number of the dice other than 7. The engine updates game and continues. Then if that player wants to build settlements or road, or develop his/her settlements, the player builds these. The game engine updates the game again. If there is not building, player clicks Next Turn button and the engine pass the next player. This loop continues till the victory of one player.

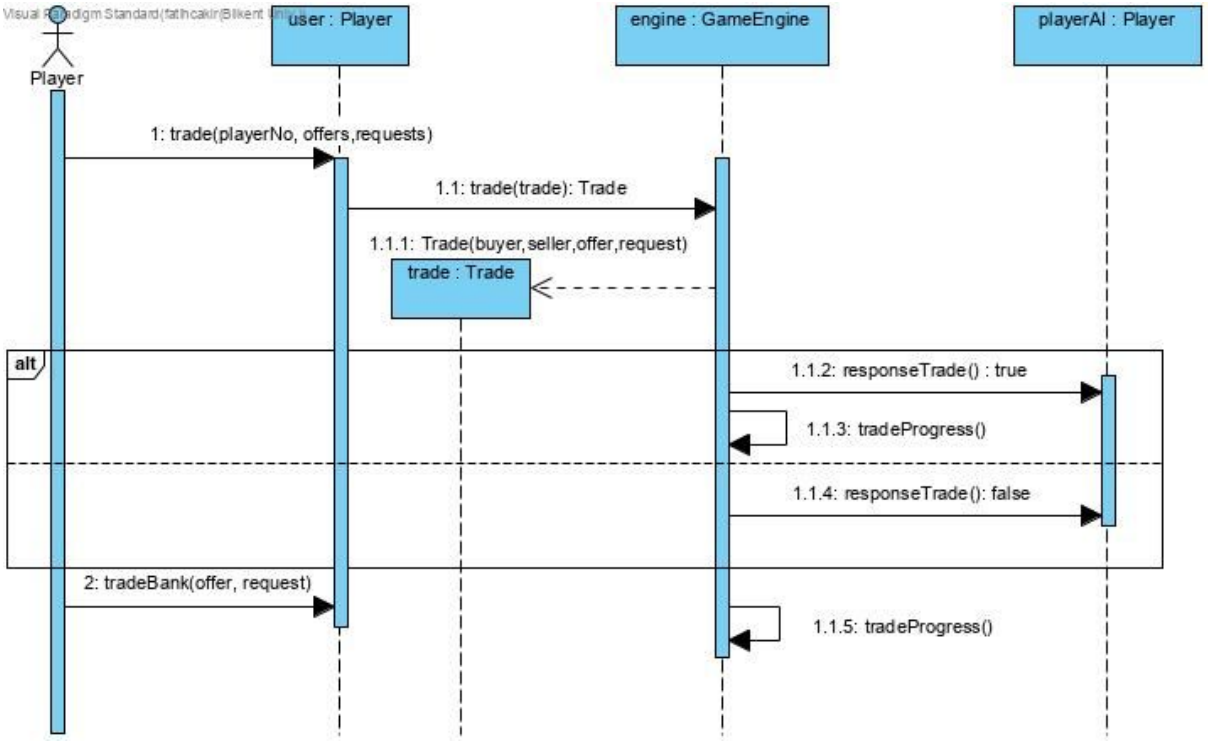

Figure 10: Sequence Diagram of Trade Process

When a player wants to trade his/her resources with other players, the player creates a trade and the game engine sends this trade to other players. If any player accepts the trade, the game engine exchanges the offer and request resource cards. If all player reject the trade, game engine does not exchange.

If a player wants to trade with bank, the player create a trade with bank request and the game engine exchange the cards with its constant rate. After all of the exchanges, the game engine updates the game.

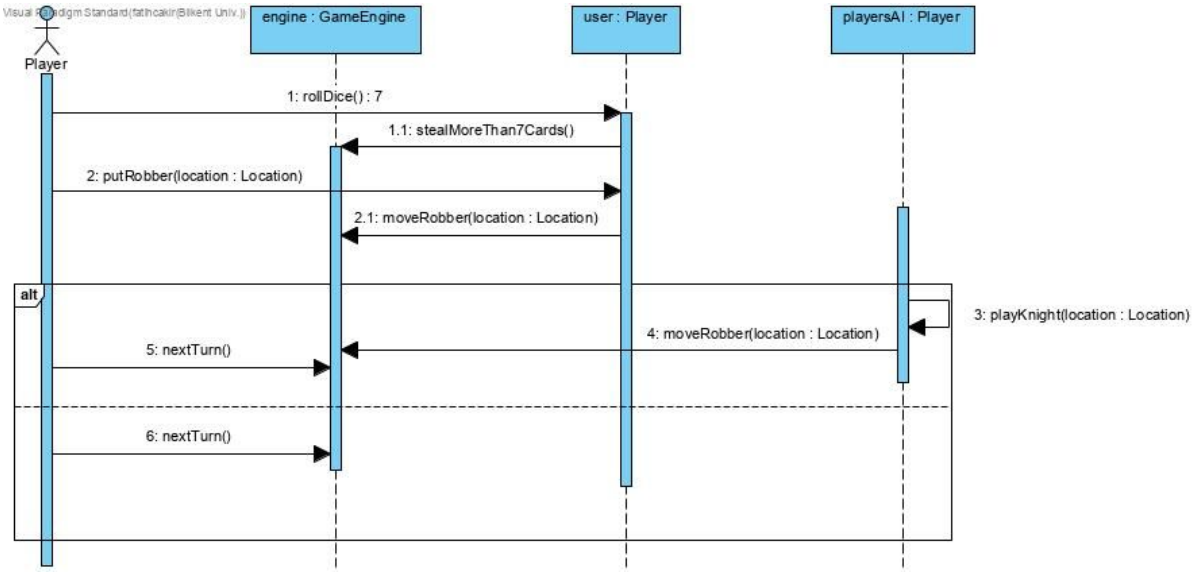

Figure 11: Sequence Diagram of Rolling 7 in Robbery Mode

If the total number of the dice is 7, the game engine steals half of resource cards from players which have more than 7 cards. Then the player puts the robber a valid location. If any player who has Knight Card wants to play his/her development card, he put the robber a valid location. After all move is done, the player press Next Button and the game engine pass the next turn the game.

5.2.2.2 Menu Sequence Diagrams

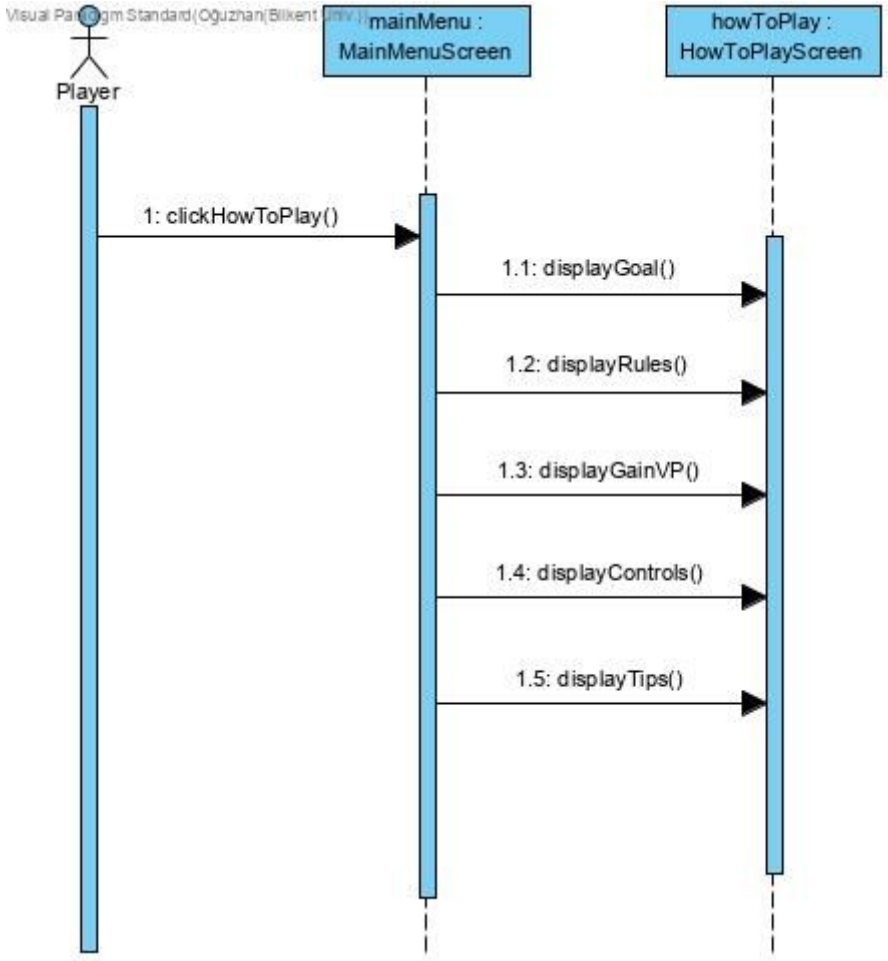

Figure 12: Sequence Diagram of How to Play Screen

When a user click How To Play button from the Main Menu, the how to play screen is displayed. The player clicks any button of the sub topics and these instructions are displayed on the screen.

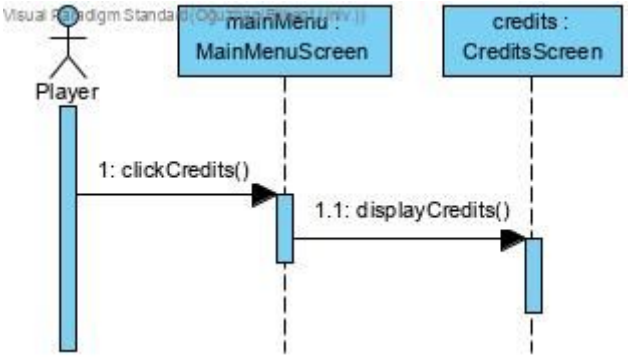

Figure 13: Sequence Diagram of Credits Screen

When the user clicks on Credits button on the Main Menu screen, Credits screen is displayed.

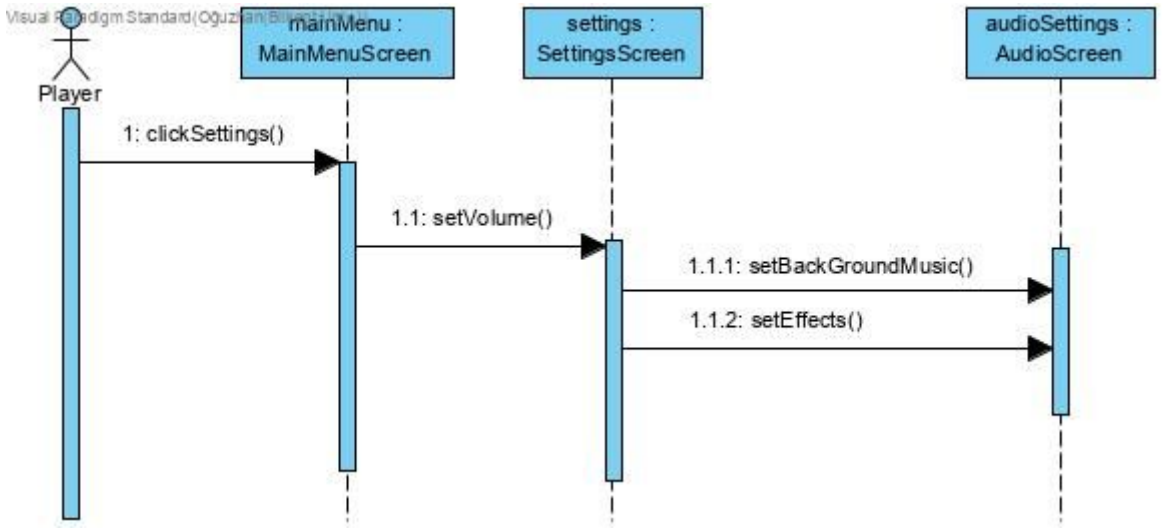

Figure 14: Sequence Diagram of Settings Screen

When the user clicks on Settings button on the Main Menu screen, Settings screen is displayed. The user can set the background music and effects from there.

5.3 Object and Class Model

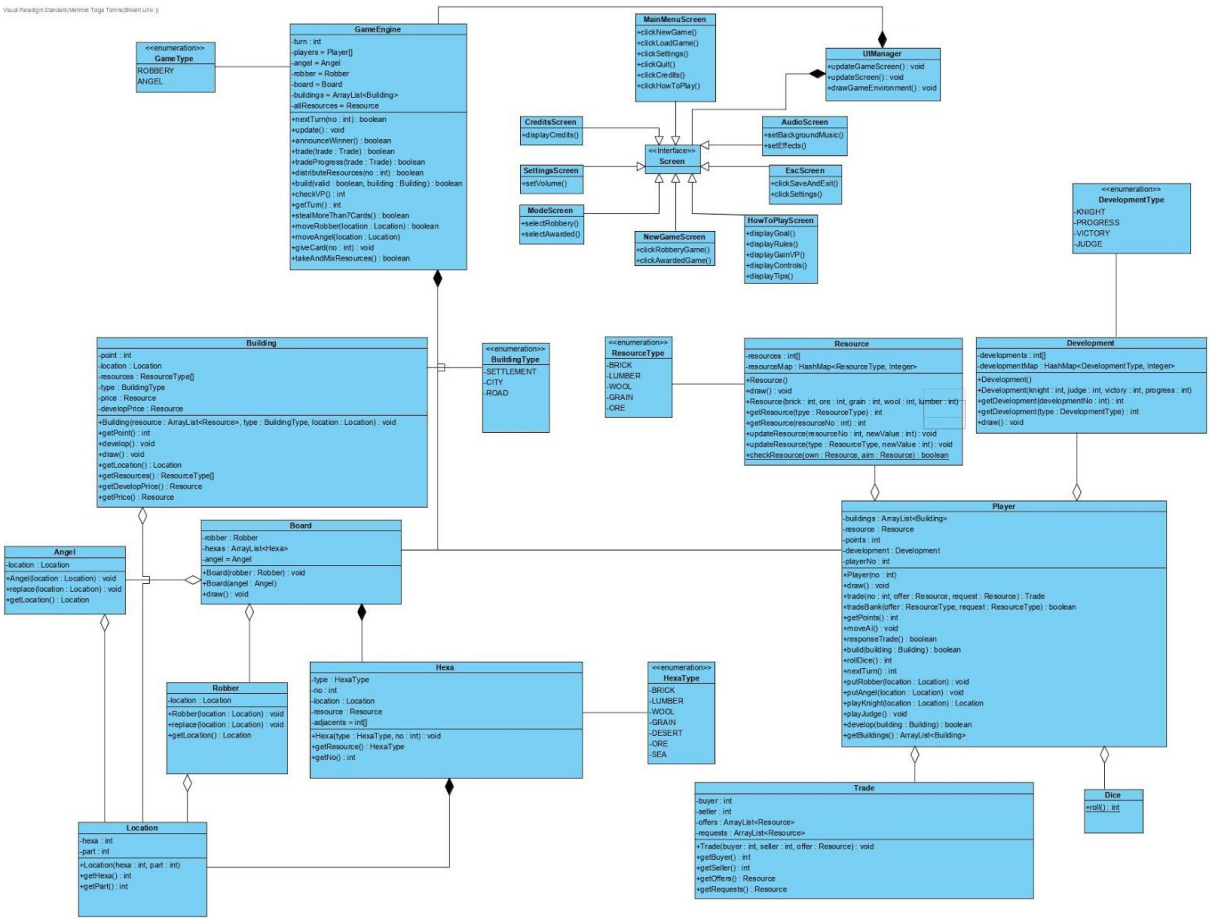

Figure 15: Class Diagram

5.4 User Interface

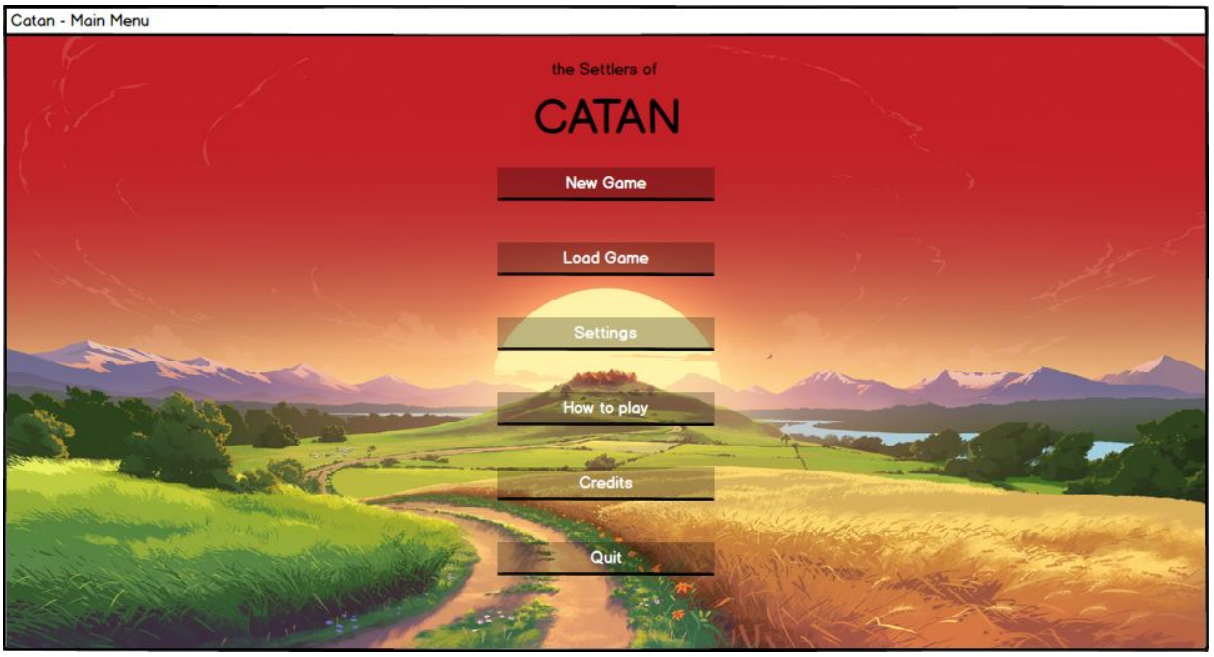

Figure 16: Main Menu

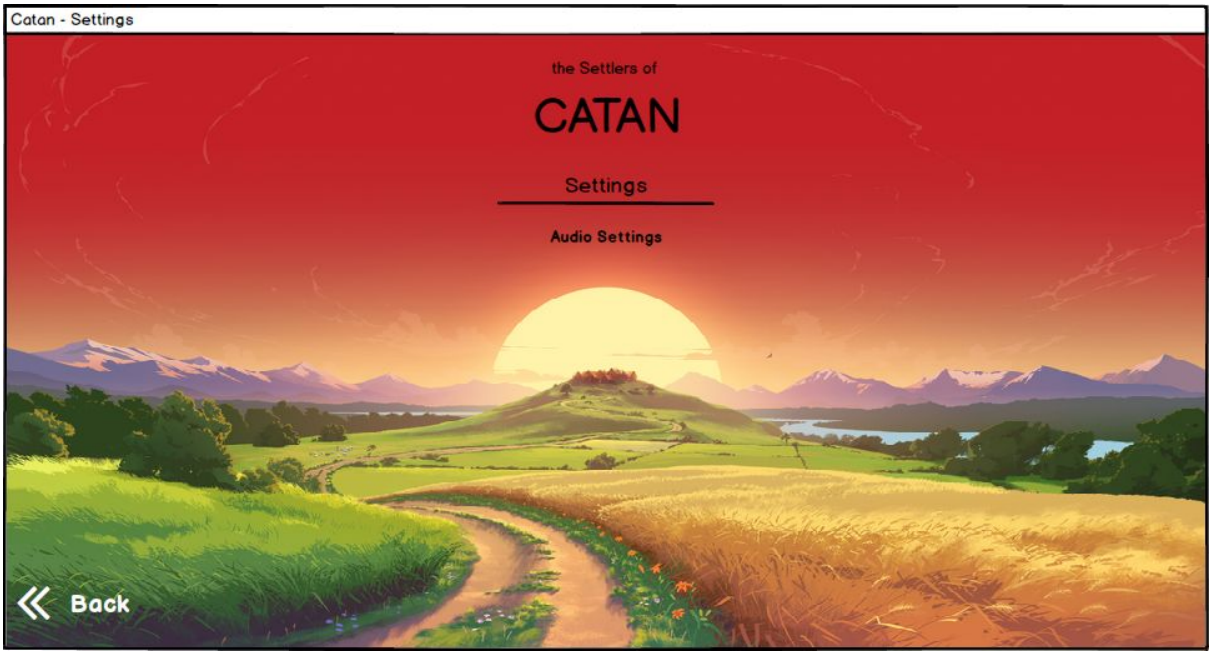

Figure 17: Settings

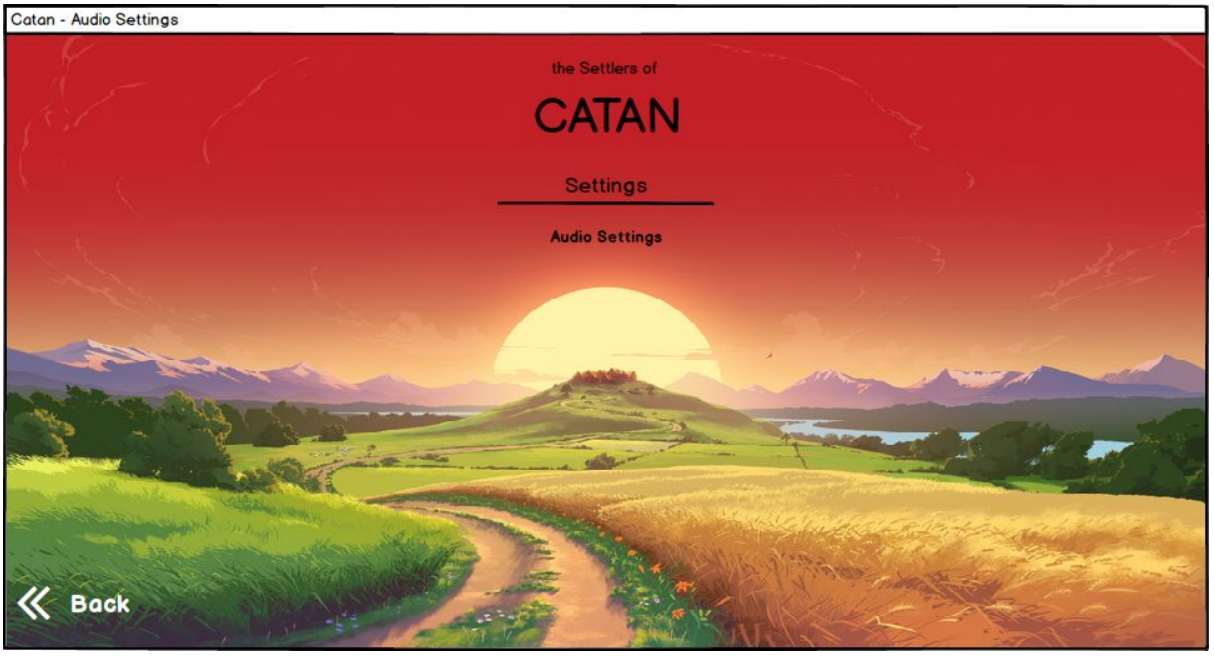

Figure 18 : Audio Settings

Figure 19: How to Play

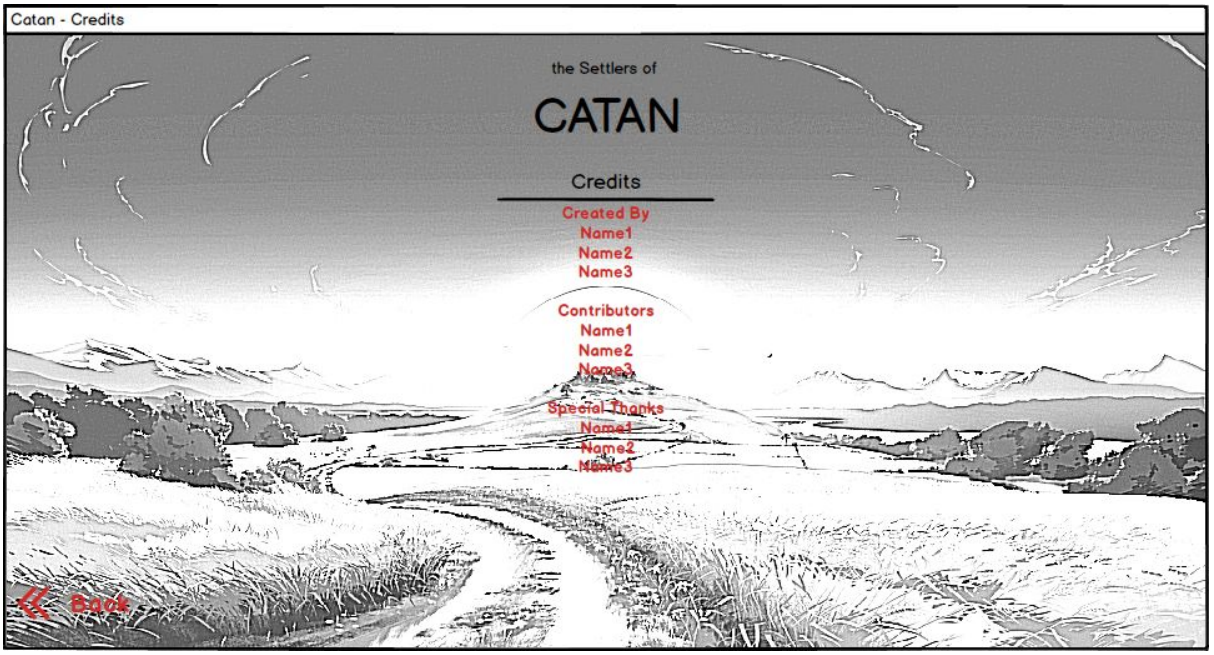

Figure 20 : Credits

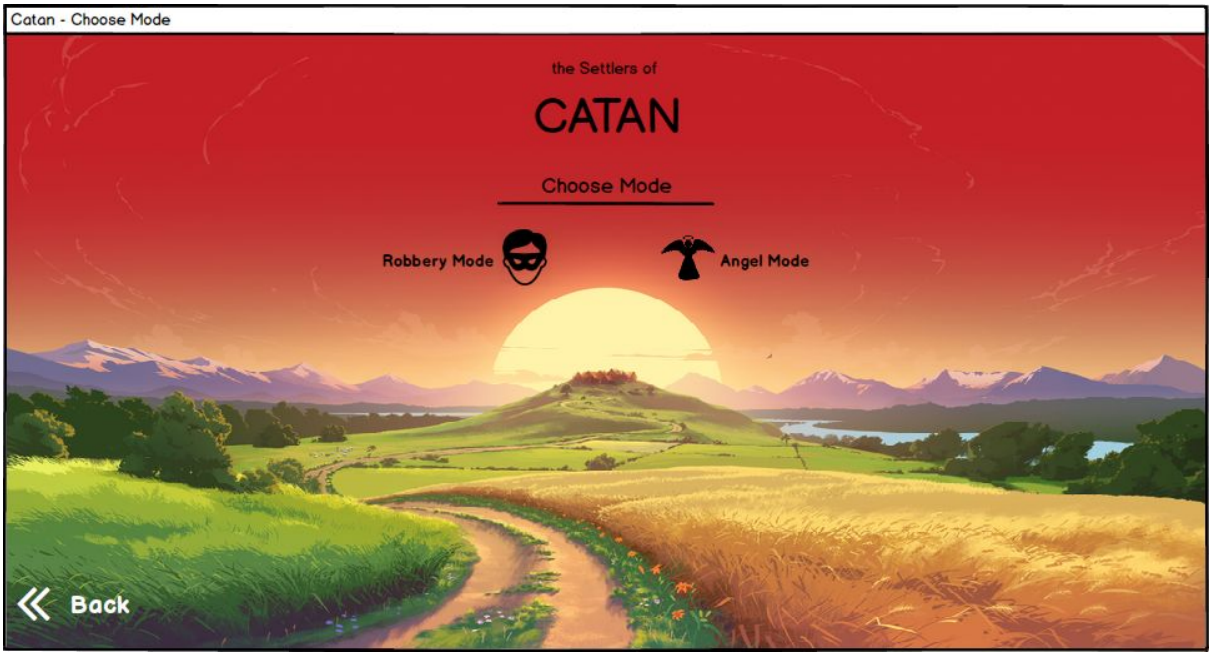

Figure 21: Choose Mode

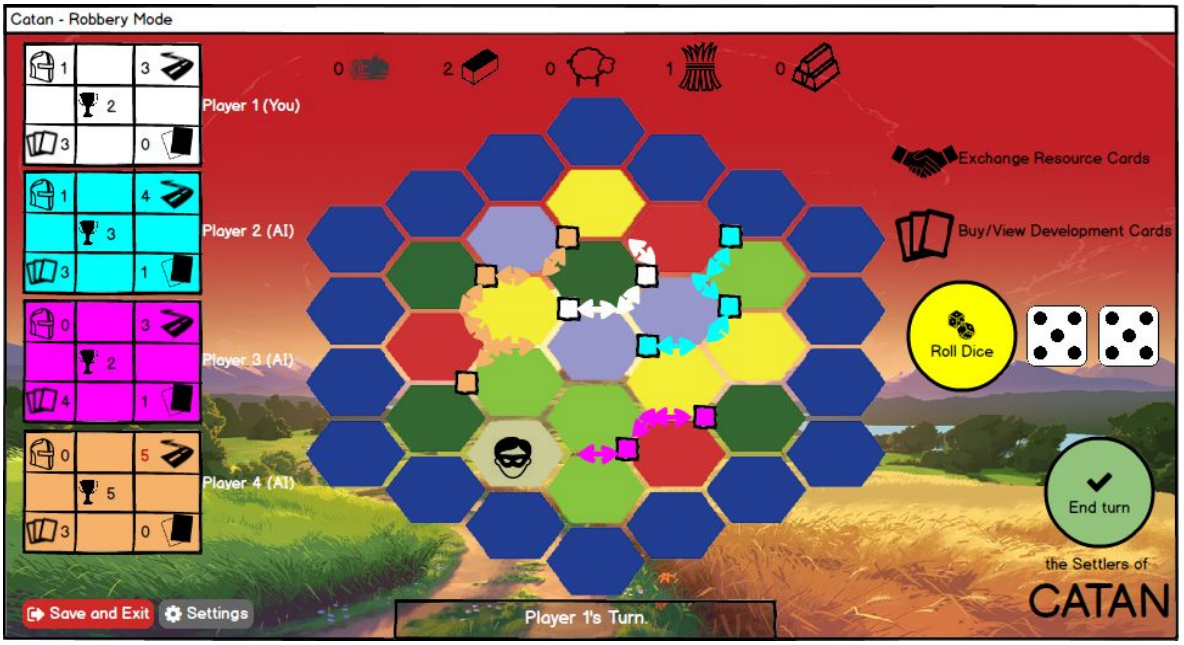

Figure 22: Robbery Mode

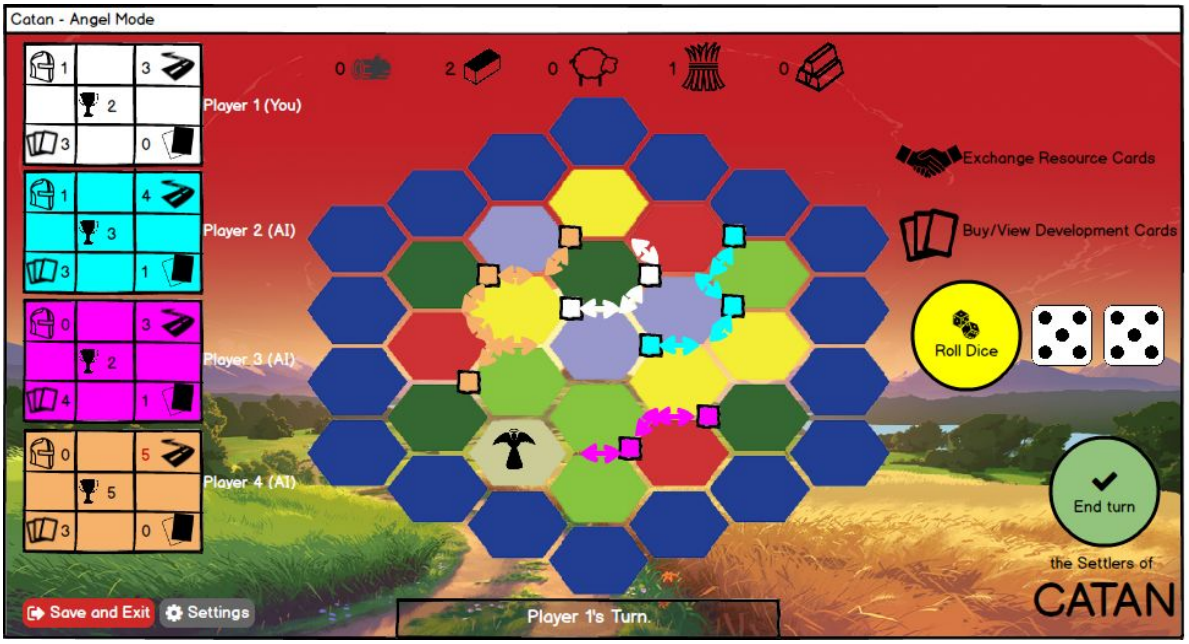

Figure 23: Angel Mode

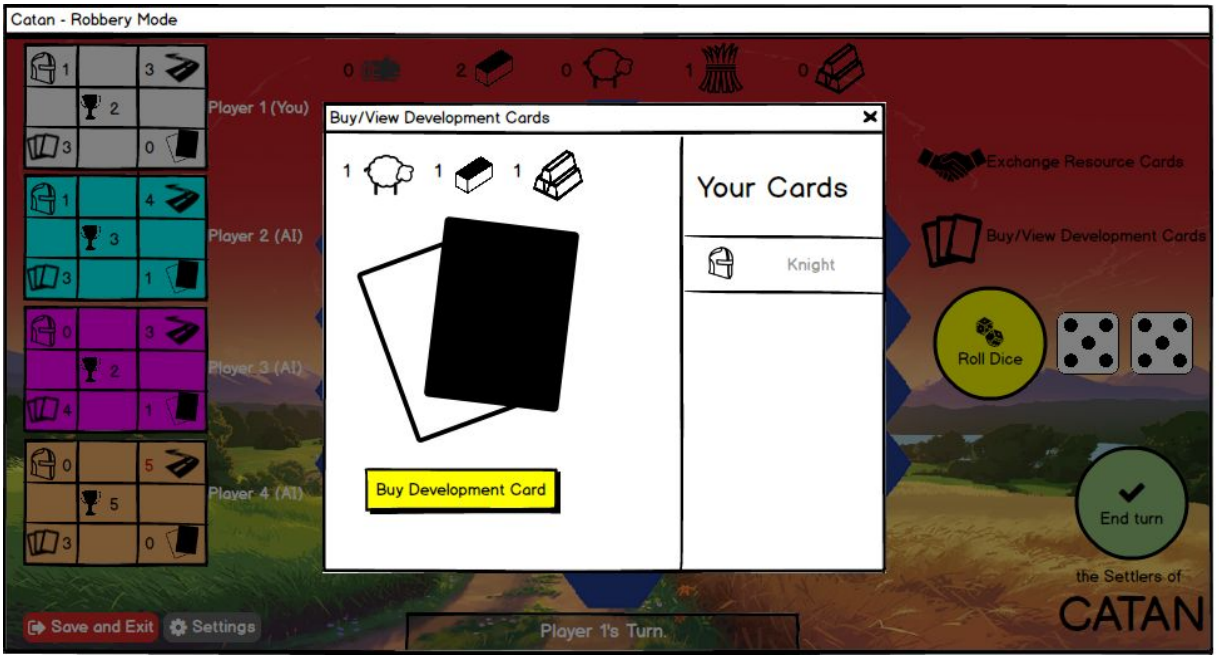

Figure 24: Buying Development Card

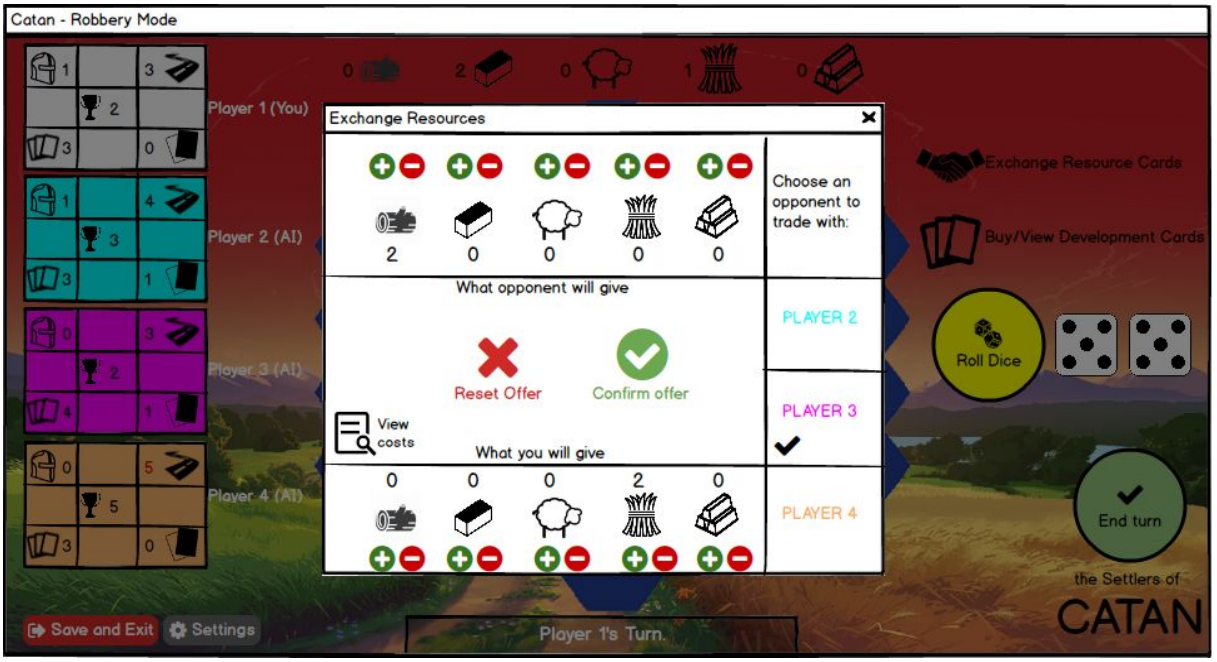

Figure 25: Trade Menu

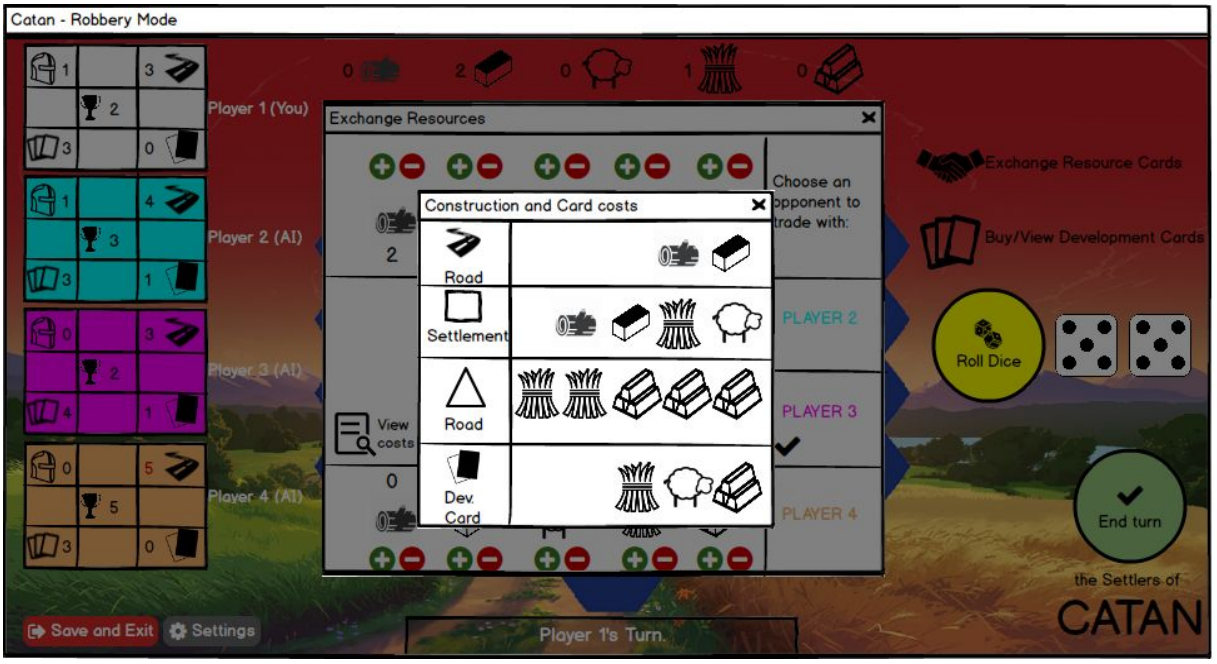

Figure 26 : Costs

Figure 27 : The end of the game

6. Improvement Summary

This report is the second iteration of the Analysis Report for our Catan Game. We fixed our mistakes according to given feedback for the first iteration. Our UI part was almost complete but the other parts had some mistakes. Also in the second iteration, we have tried to fix the errors in the diagrams. Our use case and state diagrams included only main menu actions. We have added the gameplay scenarios to both use case and state diagrams. We have also fixed small problems in old versions. In the sequence diagrams, we tried to correct our mistakes and methods according to the new version of the class diagram.

7. References

1. “The official website forthe world of CATAN,” Catan.com | The official website forthe world of CATAN. [Online]. Available: https://www.catan.com/. [Accessed: 25-Oct-2019].