Bilkent University

Department of Computer Engineering

CS319 Term Project

Monopoly Bilkent Edition

Group 1B

Iteration 2 Analysis Report

Project Group Members:

Melike Fatma Aydoğan - 21704043

Ahmet Cemal Alıcıoğlu - 21801700

Zübeyir Bodur - 21702382

Beril Canbulan - 21602648

Mehmet Çalışkan - 21802356

Instructor: Eray Tüzün

Teaching Assistants: Barış Ardıç, Elgun Jabrayilzade, Emre Sülün

Table of Contents

1. Introduction

In our CS319 OOSE Term Project, we chose to implement the Monopoly game. Moreover, we come with the design of the Monopoly with Bilkent University buildings so that we created the Monopoly Bilkent Edition. During the analysis process, while we thought about the technologies we should add, we tried to enhance the users playing experience.

2. Current System

Monopoly is a board game that players can enjoy to play together in their leisure time. In the game, players try to be rich as much as possible to bankrupt other players and win the game. Furthermore, the Monopoly game has many rules and versions, so that for our system, we choose some rules from the original game and add some rules to increase the fun.

3. Proposed System

3.1. Overview

3.1.1. Game

Our game allows friends to play Monopoly without physical meeting by being online. A friend will choose the Monopoly mode and start the game then will share the game code with other friends.

We improved the Monopoly experience with some improved features like trading and teams.

3.1.2. Map

We designed our Monopoly with Bilkent University buildings. So that, we create another version of the Monopoly game.

3.1.3. Chat

To increase the communication of the players, the system has one or two chat sections. In the team mode, for team communication a private chatting box will be available. Besides that, in each mode a general chat box will be available for all players' communication.

3.2. Functional Requirements

3.2.1. Multiplayer and Online Game

Monopoly Bilkent Edition is a multiplayer game. Players can play this game online with their friends without physical interaction. So that they need the internet and computer during their play.

3.2.2. Opening a game

A player can open a game in three ways. New game can be started. With this option, the system will create a game code and this code will be used by other players. Another way is attending an already created game by using the game code. At last, the player can continue a game that he or she initialized. After that, they will share the code again with other users.

3.2.3. Choosing a game mode

While the player starts a game, he or she needs to choose the game mode whether to be with teams or not and to be with speed dice or not. After the choice process game code will be shared on the screen.

3.2.4. To start the game

Each player should choose their pawn and if they are in team mode they, also, should enter their team number. Once the players have entered the necessary information, dice will be rolled to choose the order of players' turn.

3.2.5. Finishing or Saving the Game

Monopoly finishes when only one player has properties and money when other players are bankrupt. However, since the Monopoly game is a long game we will add save and finish early options. For both options, a player should offer them and other players will accept or decline it. According to the majority, the game will continue or end. For the finish early option, the system will sum up the money and the values of properties and buildings. Winner will be decided according to this summation. Besides that, a player can choose to leave the game in such a situation, the player's properties will go to the bank.

3.2.6. Trading

The system has a section for trading. From this section players can offer trading to other players in two ways. The player can offer to sell an item or buy an item. In both offers, if the other player accepts trading they will exchange items or the other player can offer new value for the properties. Also, it is important that a player only can offer during their own turn.

3.2.7. Speed Die

In this mode, players are not rolling 2 dice, but they are rolling 3 dice where one of them is a speed die. This speed die contains one 1, one 2, one 3, one BUS and two MR. MONOPOLY surfaces.

3.2.8. Teams

By choosing teams mode, players can be separated into teams at the start of the game.

3.2.9. Chat

Players can text in chat boxes to message other players. In the team mode team members can communicate privately with other team members otherwise all messages will be public to all players.

3.2.10. Losing the Game

A player can continue to watch the game after losing the game. They will not be able to play their turn only, the system will skip their turn since they bankrupted.

3.3. Non-functional Requirements

3.3.1. Quality Requirements

Usability : The Monopoly game includes many various actions during the many different cases. Therefore, to decrease the confusion, on the screen possible actions will be listed so that a player can decide what to do during his/her turn. Also, a how to play page will be on the main page. So that, literate people can play our game.

Reliability : Our Monopoly needs the Internet since it is played online. So that we will check each player's connections. Otherwise, the system will give output related to the situation and pause the game for 15 seconds to wait for that player. If the player cannot connect to the game again other players can vote and can continue to the game. Otherwise, other players will be bored. Also, as stated in the usability, since the system shows the options during a turn, it will decrease the possibility of false input and increase the reliability.

Performance : Our system will use computer storage to save games and use KryoNet Server to play the game online. The estimated highest latency will be 200 ms with the efficiency of KryoNet. Also, our game will support 6 players on a server.

Supportability : Since our system is well designed according to the hierarchy and decomposition, necessary changes or additions can change the properties' roles without many changes.

Implementation Constraint : Our project was only restricted to Object Oriented Programming Languages and we chose to code in Java language and use its JavaFX library to implement the GUI.

3.4. System Models

3.4.1. Use-Case Model

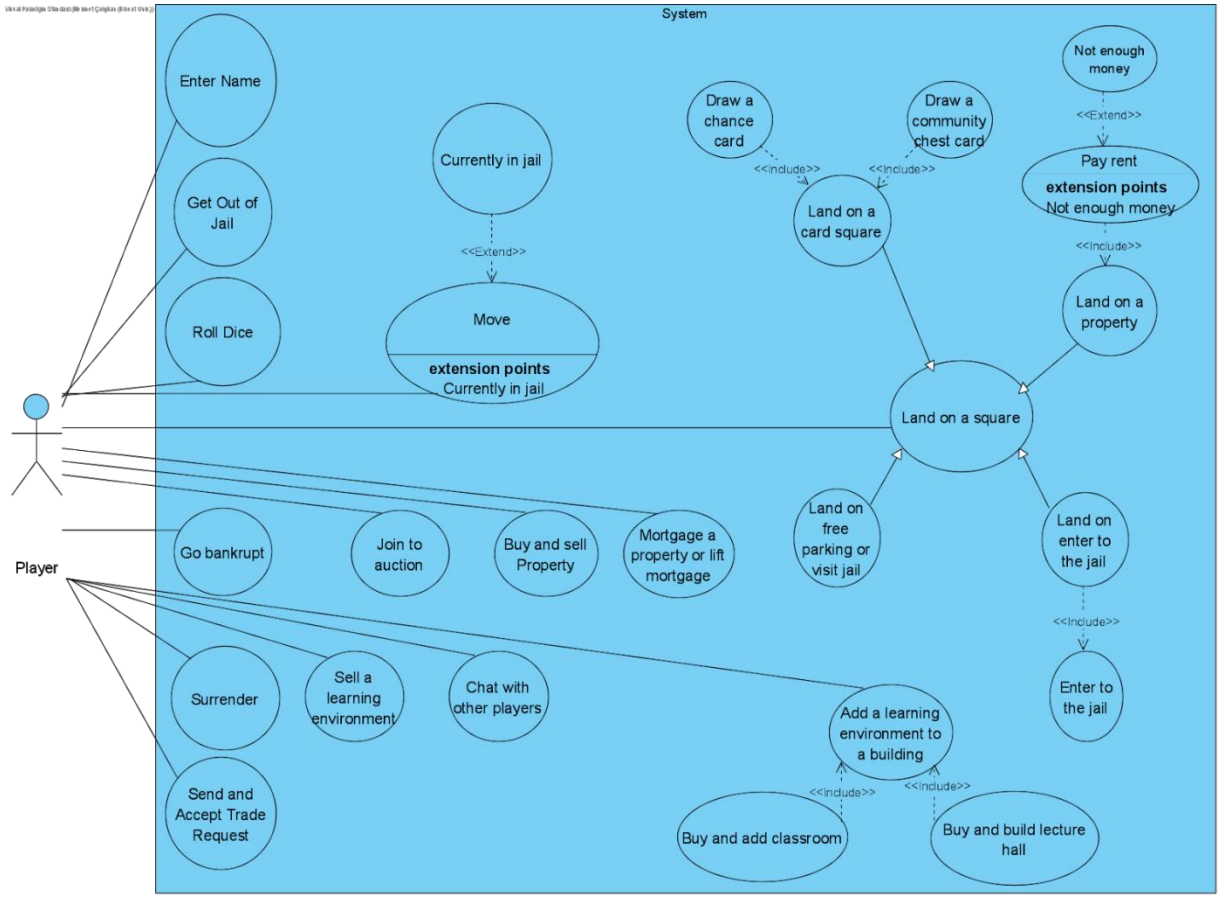


Figure 1 : Use Case Model for Monopoly Bilkent Edition

Use Case #1: Create Game

Primary Actor: Player

Stakeholders and Interests:

- Player wants to create a game
- System creates the game session and places the player

Pre-conditions:

- Player must have entered a username.
- Player must have a proper internet connection.

Entry-conditions:

- Player clicks at the Create New Game button.

Event flow:

- Player clicks at the Create New Game button.
- Player chooses a game mode (Team/ Speed Die).
- Player gives the Game ID to other players for them to join the session.
- After the game session has a minimum 2, maximum 6 players, Player clicks at the “Start the Game” button.
- System starts the game session.

Use Case #2: Join Game

Primary Actor: Player

Stakeholders and Interests:

- Player wants to join a pre-existing game session.
- System places the player to the specified game session.

Pre-conditions:

- Player must have entered a username.
- Player must have a proper internet connection.

Entry-conditions:

- Player clicks at the “Join Game” button.

Event flow:

- Player clicks at the “Join Game” button.
- Player enters the Game ID to the text field.
- If the Game ID is correct, System places the player into the pre-existing game session.

Use Case #3: Turn of a player

Primary Actor: Player

Stakeholders and Interests:

- Player wants to complete a turn.
- System processes the player actions.

Pre-conditions:

- Player must have the control.
- Player must not be in the jail.

Event Flow:

- Player clicks at the “Roll Dice” button.
- System processes the request and displays the result of dice.
- System moves the player to the specified square.
- Player completes the actions according to the square that he/she landed.

- System processes the actions that player has taken.
- Player clicks at the “End Turn” button or time limit reaches.
- System ends the turn and takes over the control to another player.

Alternative Event Flows:

- If player throws double:
    a. If this is the first or second double, System gives the turn to the player again.
    b. If this is the third double, System puts the player into the jail.

Use Case #4: Land on a not owned property square

Primary Actor: Player

Stakeholders and Interests:

- Player lands on a not owned property square.
- System processes the player actions.

Pre-conditions:

- Player must land on a property square.

Event Flow:

- Player lands on a not owned property square.
- System shows a choice panel which has 2 options: Buy or leave it to auction.
- If player clicks at “Buy” button and player has enough money to buy the property, System takes the specified money amount and assigns the property to the player.
- If player clicks at “Leave it to auction” button, System starts an auction process. After ending the auction, System takes the specified money amount and assigns the property to the winning player.

Use Case #5: Land on an owned property square

Primary Actor: Player

Stakeholders and Interests:

- Player lands on an owned property square.
- System processes the player actions.

Pre-conditions:

- Player must land on an owned property square.

Event Flow:

- Player lands on an owned property square.
- If the player has enough money to pay the rent, System transfers the money to the property owner.
- If player doesn't have enough money to pay the rent, System starts an trade process between the player and the property holder.
- After ending the trade process, System processes the requests.

Alternative Event Flows:

- If the player doesn't agree with the property owner in the trade process, System bankrupts the player.

Use Case #6: Build Classroom or Lecture Hall on the building

Primary Actor: Player

Stakeholders and Interests:

- Player wants to build a lecture hall or classroom to the building.
- System adds the learning environments into the building.

Pre-conditions:

- Player must own all buildings in the color group.

Event Flow:

- Player clicks at the “Build on Building” button and specifies which learning environment he/she wants to build.
- System compares the money of the player and cost of the learning environment.
- If the conditions satisfy, System assigns the specified learning environment to the building and takes the money from the player.

3.4.2. Object and Class Model

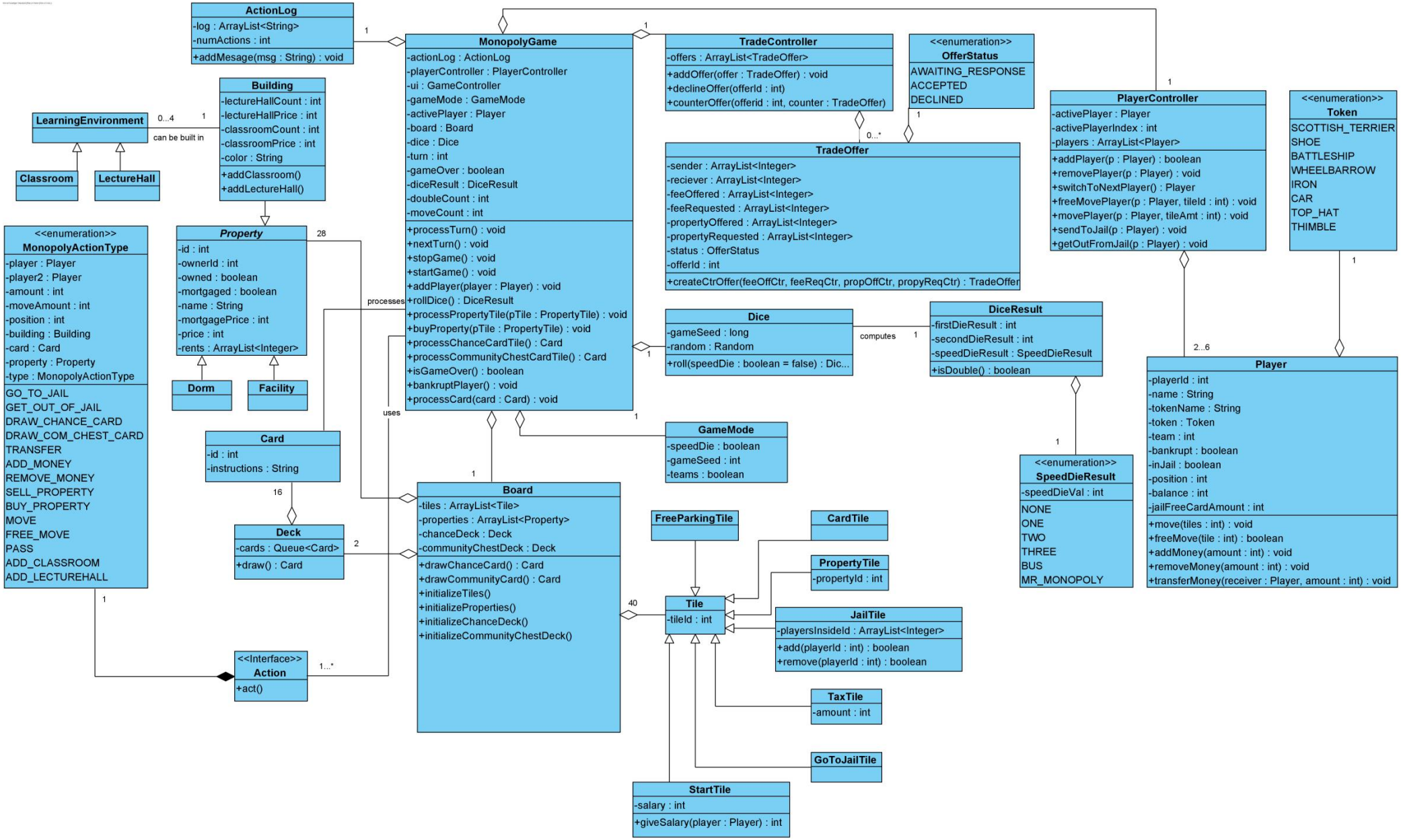


Figure 2: Object and Class Model

3.4.3. Dynamic Models

3.4.3.1. Activity Diagrams

3.4.3.1.1. Creating a lobby and starting the game

In the following activity diagram, there are initially 2-6 players, one being host, the remaining being the guests. Initially, all those players want to play the game Monopoly Bilkent Edition. There, it is described, step by step, how those players create a lobby, join a lobby and start playing the game.

First, all players open the game and decide a username. The host creates a new game. After creating a lobby, the host can change the options for teams, speed die and can lock the lobby - make the lobby private. Meanwhile, guests can enter the lobby whose ID is given to them by the host. If the ID is valid, the lobby is public and not full, guests can enter the lobby. Otherwise, they return to the “Enter Game ID” screen.

The host has to wait for at least one player to arrive - there has to be at least 2 players in the lobby. If the host allows teams in the options panel, all players need to choose a team number so that they can team up. If a player chooses the same number with another player, it means they are in the same team.

After all players decide their team number - if team mode is on - and if there are enough players, the host can press the “Start Game” button and the game will start.

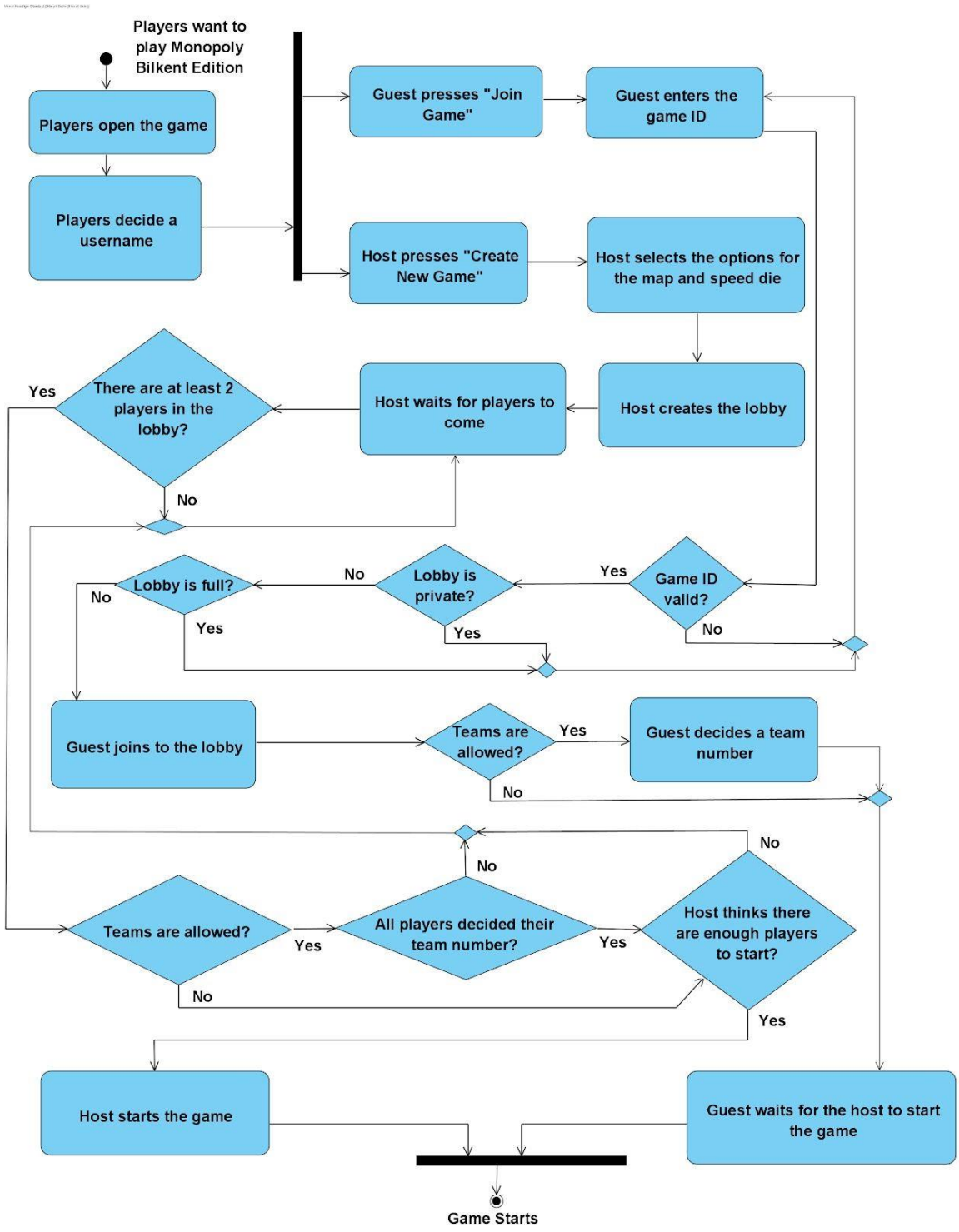


Figure 3 : Activity diagram of creating a lobby and starting a game

3.4.3.1.2. Playing the game

In the following diagram, the win conditions and result of bankrupting in our system are described. Moreover, getting into the jail by rolling three consecutive double rolls and repeating a turn by rolling a double is also described in this diagram. In our system, rolling a double is discluded from rolling a triple; if a player rolls a triple they are not considered to roll a double.

First, players start the game by deciding the order they play by rolling dice simultaneously. The player with the highest dice result starts first.

In the diagram, active player refers to the player who is currently playing. Same term is also used in the rest of the diagrams, even though the term is not an object.

In each turn, the active player rolls their dice. If he/she didn't make his/her third consecutive double roll, he/she moves their token to the place the dice orders them to. Rolling a dice is well explained in the activity diagram *Rolling the Dice and Landing on a Tile*.

If they have made a third consecutive double roll, they immediately go to jail. Otherwise, they move their token and resolve the tile they land. Resolving a tile is also well explained in another activity diagram, called *Resolving a Tile and Ending a Turn*, and *Resolving a Property Tile*.

A player may bankrupt while they are resolving the tile they have landed, e.g. while trying to pay a rent to another player or trying to pay tax to the bank etc. If it happens so, they don't leave the game but they are removed from the turn queue. They remain as a spectator in the lobby and they may leave the lobby. Rules for bankrupting are well explained in *Resolving a Property Tile* and *Bankrupting* diagrams.

Moreover, a player may go to jail by resolving the tile, too. If that happens, they immediately go to jail and leave their turn to the next player. This part is also explained in the state diagram *Jail* and *Resolving a Tile and Ending a Turn*.

Also, the same rule applies for rolling a double in Monopoly Bilkent Edition. If the active player rolls a double, after they end their turn, they still remain as the active player and roll the dice again. Otherwise, the player leaves their turn to the next player after they end their turn.

At one point in the game, players will bankrupt and there will be only one player remaining. If so, that player is the winner and the game is over. If the game mode allows teams, this will occur when there is only one team remaining.

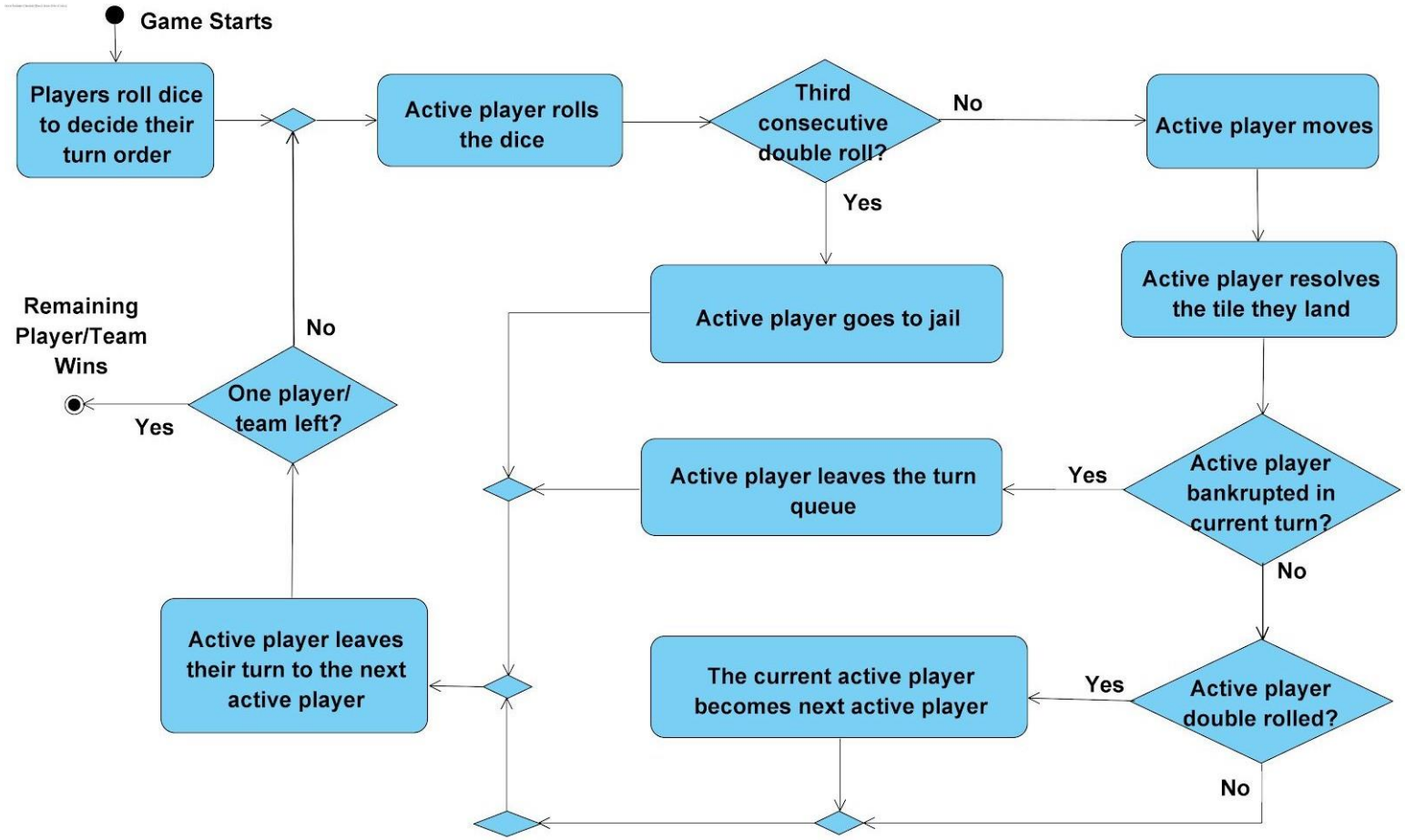


Figure 4 : Activity diagram for a Monopoly Bilkent Edition game

3.4.3.1.3. Rolling the Dice and Landing on a Tile

In the following activity diagram, it is assumed that the player is not in jail and didn't make their third consecutive double roll. There, it is described how the usual dice and speed die feature works.

First, the player rolls the dice. If there is no speed die, they just move their token units equal to the outcome of the dice. Speed die feature works with minor changes compared to the original version of Monopoly™. The player can use the following outcomes of the speed die if he/she were to land on go-to-jail tile; the bus, 1, 2 and 3.

If the speed die surface is a number, they use it like a third die. Apart from that, if they make a triple roll (1-1-1, 2-2-2 or 3-3-3) they can freely move their token to wherever they want (free move), even if the sum of the white dice results in the player going to jail. However, the player doesn't cross the start tile when he/she makes a free move, as if they teleport to the tile.

If the speed die surface is a "Bus", they are given the option to choose the dice result between one die, the other die or both, even if the sum of the white dice results in the player going to jail. After they choose, they continue their turn as usual.

If the speed die surface is a "Mr. Monopoly", the player first resolves the white dice and the place they land. Failure to resolve the tile they land (going to jail or bankrupting) blocks the player from using Mr. Monopoly, as drawn in the diagram. Successfully resolving the tile, the player then does one of the following.

If there is property in the bank, they advance to the next unowned property, so that the outcome of the Mr. Monopoly will always be sold by the bank.

Else, the player advances to the next property that is owned by another player from a different team, so that they will always owe rent to another player.

The player may cross the start tile unless he/she rolls a triple dice.

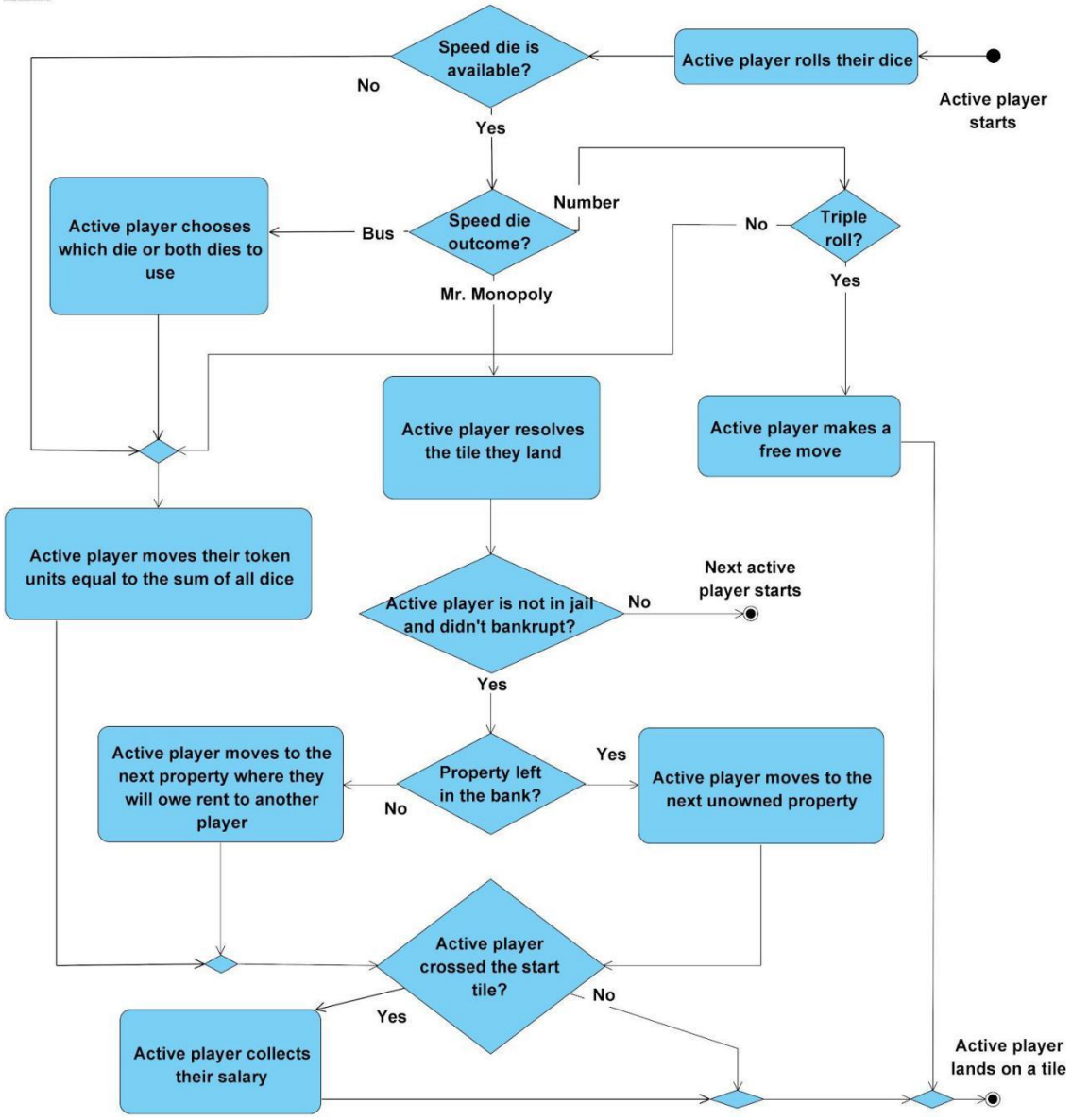


Figure 5 : Player rolls the dice and lands on a tile

3.4.3.1.4. Resolving a Tile and Ending a Turn

In the following diagram, the actions that happen when a player lands on a tile is explained. Moreover, post-actions a player can take before they can end their turn is also explained. Initial condition is that the active player lands on a tile and the final condition is that the next active player starts their turn.

If the active player lands on the Jail Tile (Just Visiting), Start Tile or Free Parking, they don't do anything and may make a post action before ending their turn. Active player doesn't get a salary when he/she lands on the Start Tile as they get their salary while they cross the start tile.

If the player lands on the Go-To-Jail Tile, they immediately go to jail and their turn ends.

If the player lands on Chance/ Community Chest Tile, they draw its card. Then, they resolve the card. The player could go bankrupt or go to jail, as well, while resolving the card. If either of them happens, their turn ends.

If the player lands on a Property Tile (a Building, a Dorm or a Facility), they resolve it. Resolving a property tile is well explained in *Resolving a Property Tile* diagram. A player could go bankrupt while resolving a property tile. If that happens, their turn ends.

If the player lands on a Tax tile, he/she pays the tax to the bank. They could go bankrupt as well, if that happens, their turn ends.

After successfully resolving their tile without going bankrupt or going to jail, a player can do one of the following before ending their turn.

- Send a trade request
- Construct a learning environment (classroom or a lecture hall)
- Sell a learning environment back to the bank
- Mortgage a property
- Lift mortgage from a property

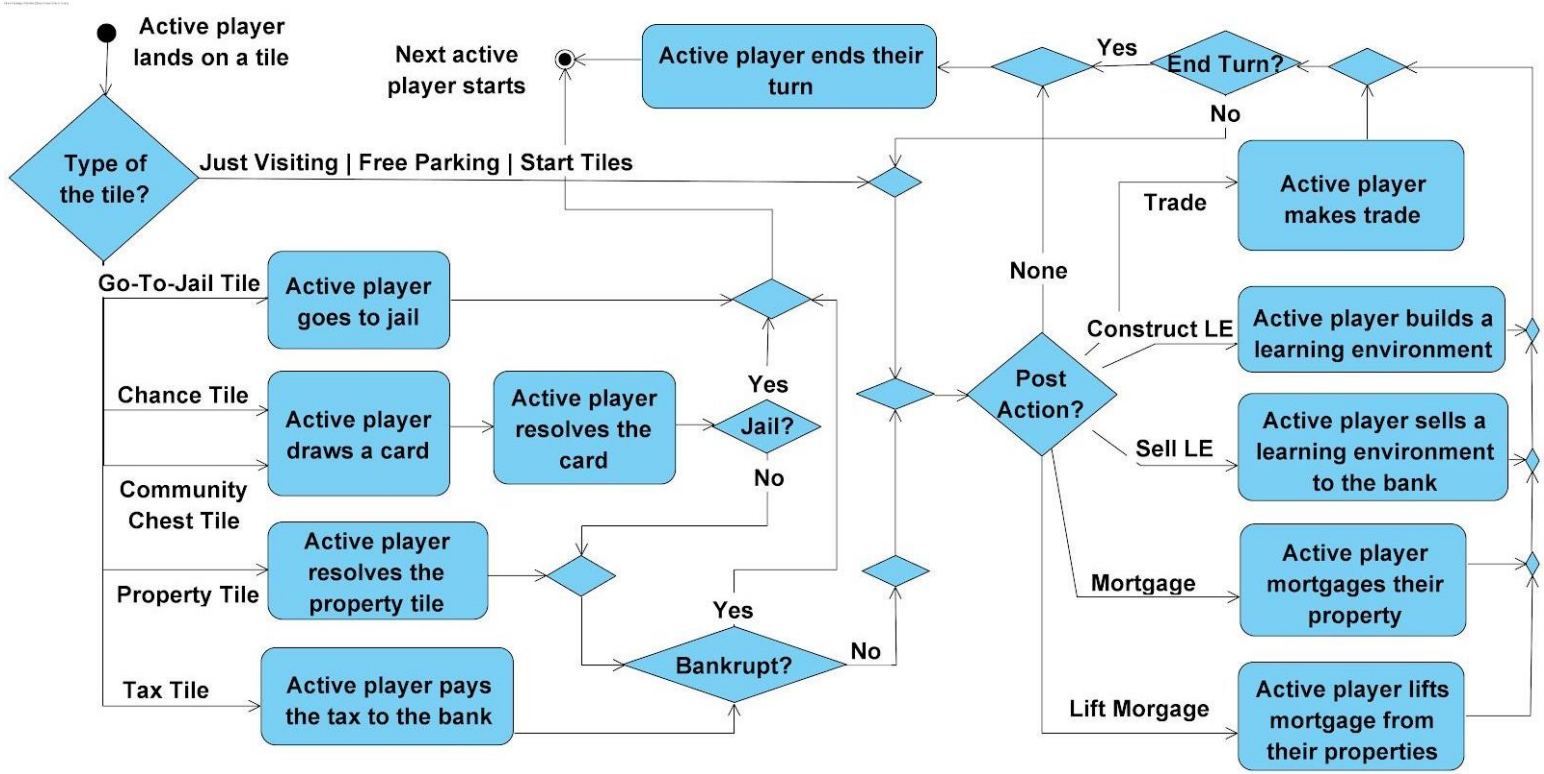


Figure 6 : Player resolves a tile

3.4.3.1.5. Resolving a Property Tile

In this diagram, how a player buys a property and pays a rent to another player is described. Initial condition is that the active player has landed a property tile.

First, the game checks the owner of the property. If there is no owner, the player may buy the property or leave it to auction.

If the player wants to buy it, they need to have sufficient funds for that property. If the player has the sufficient funds, they buy the property.

If the player doesn't have, however, they may mortgage a property, sell a learning environment to the bank or send a trade request to gain more money. The player may repeat those steps until they have sufficient funds and buy the property. The player may end those steps by leaving the property to auction if they want to do so, e.g. they realize that they actually can't buy the property. Then, the bank will start an auction for that property. Auctions are described in the state diagram *Auctions*.

If the owner of the property is the active player (self), another player from the same team or the property is mortgaged, the player does nothing.

If the owner of the property is another player from a different team (every players' team is considered to be different if team mode is disabled), the player owes rent to that player. If the player has sufficient funds for the rent, he/she simply pays the rent to the other player.

If the player doesn't have the sufficient funds, same rules for insufficient funds for buying a property apply. The only difference between two is, if it is for sure that the player won't be able to pay the rent no matter how much he/she mortgages, sells learning environments or trades, they go bankrupt.

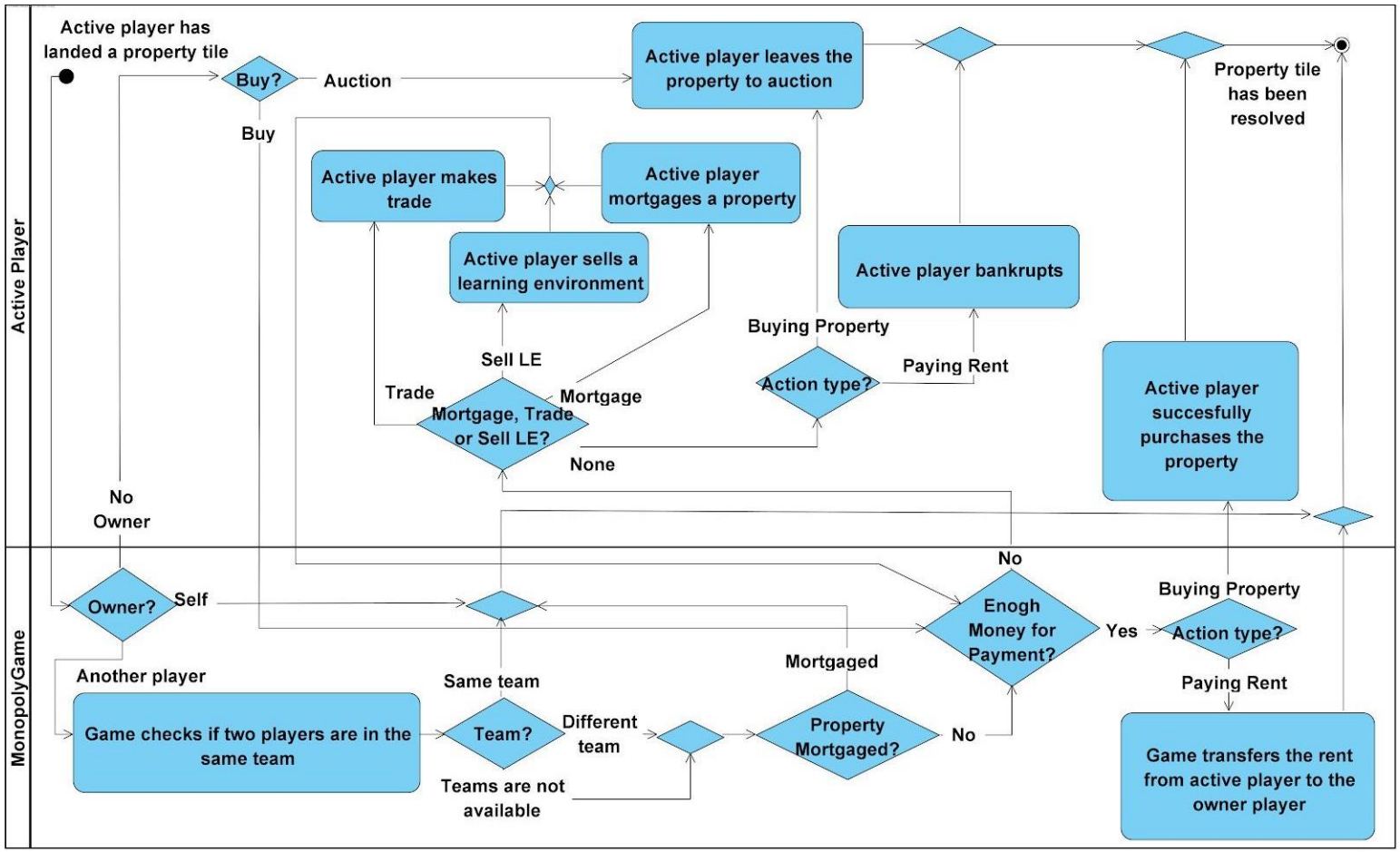


Figure 7 : Resolving a property tile

3.4.3.1.6. Bankrupting

In the diagram below, what happens to player's properties after they go bankrupt is described. If the player went bankrupt by paying a rent to another player, they give their property to that player, most probably mortgaged.

If the player bankrupts by paying a tax to a bank, paying a fee from Chance/ Community Chest cards, they are said to be bankrupt by the bank and the bank starts auction for each property the player had owned.Then, the player leaves the turn queue to be a spectator in the lobby.

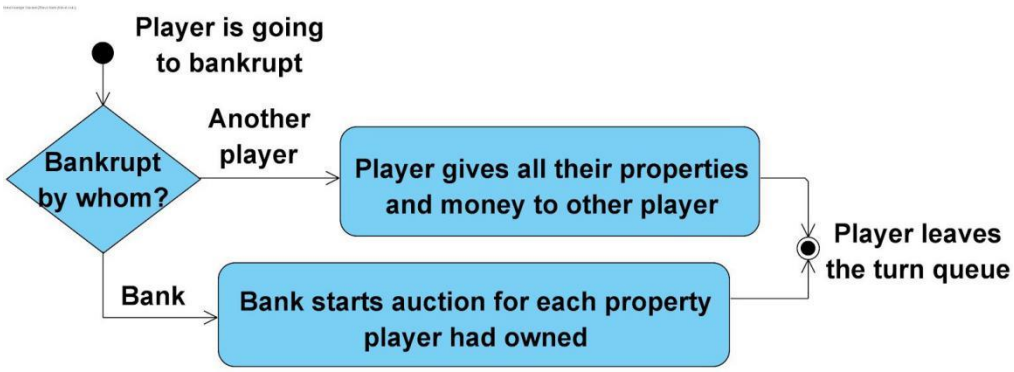


Figure 8 : Player bankrupts

3.4.3.1.7. Trading

In the following diagram, how a sender player's trade request is reviewed by the receiving player and how the receiver player makes the decision are described.

Only the active player of a turn can send a trade request in Monopoly Bilkent Edition. First, the player prepares the trade offer using the Trade panel. They can choose from his/her properties and funds to offer, for the receiver player's properties and funds to request. After choosing those, they can send the trade offer for the receiver player.

The receiver player will be notified and they will be able to see the details of the offer. Then, they can accept or reject the offer or counter-offer the existing offer.

If the receiver accepts the offer, a request is accepted and the game processes the trade.

If the receiver rejects the offer, the sender player is notified that the request is rejected.

If the receiver decides to counter-offer, the sender and receiver switch places. Then the old receiver prepares a trade request for the old sender. This counter-offering can continue until one of the players accepts or rejects.

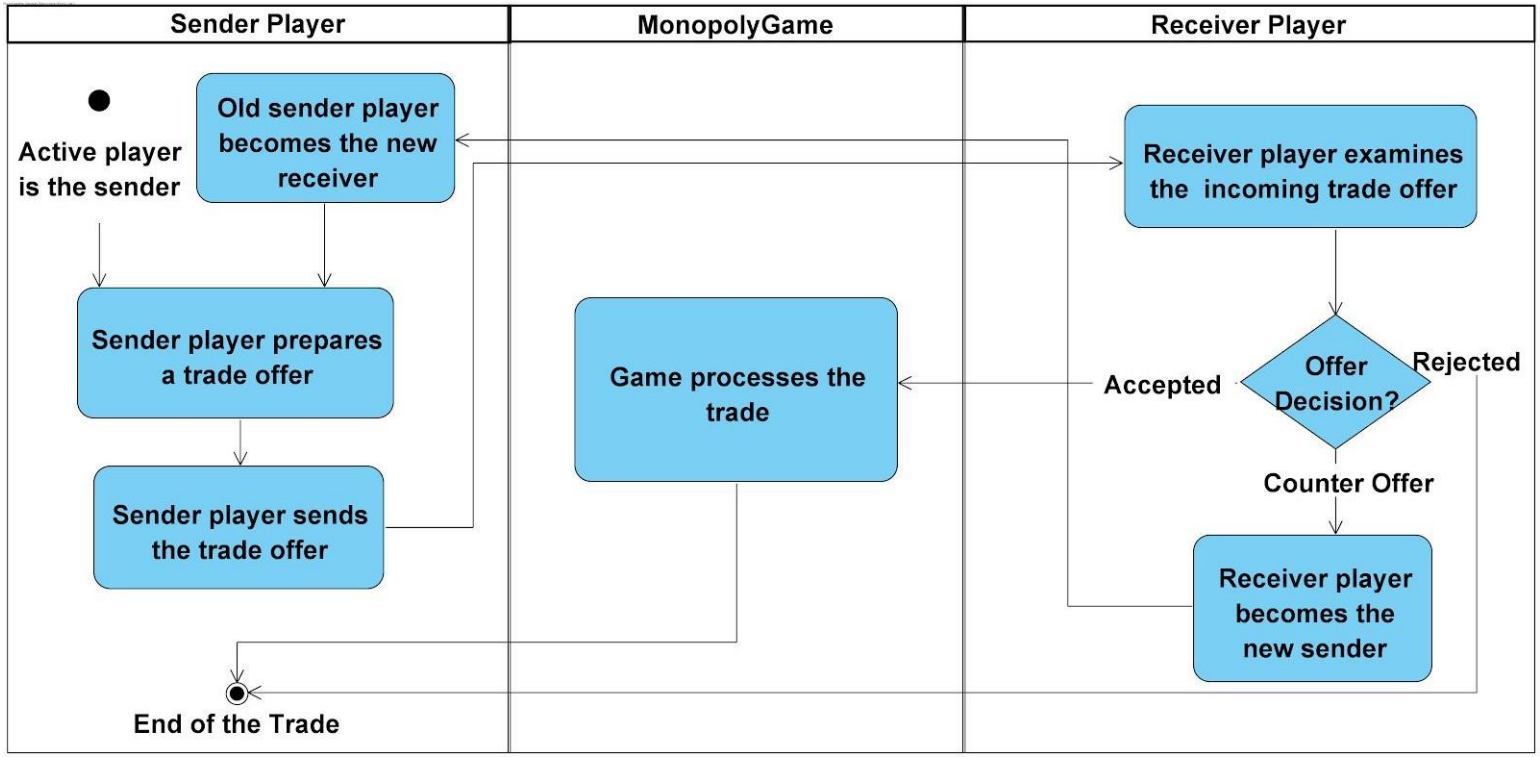


Figure 9 : Trade between two players

3.4.3.2. State Diagrams

3.4.3.2.1. Auction

In the game, if a user lands on a not owned property and does not want to buy there, the property automatically opens to the auction through the other players. States of this action are to get a new offer or waiting 15 seconds to finish the auction. Then the property will be given to the player who makes the best offer.

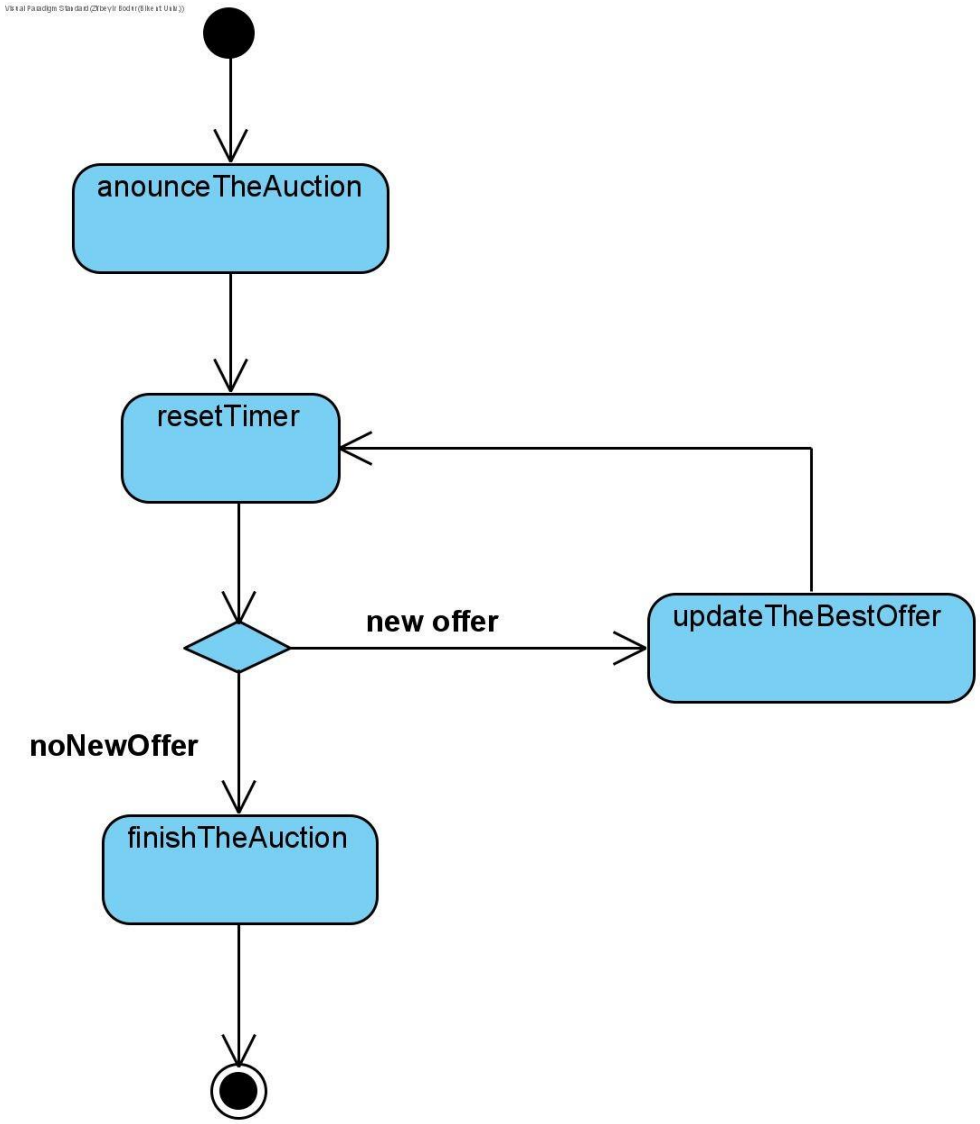


Figure 10 : State diagram for an auction

3.4.3.2.2. Jail

A player can enter a jail in three situations and after entering there there are two states that the player can get out of the jail. These states are paying the penalty or rolling double.

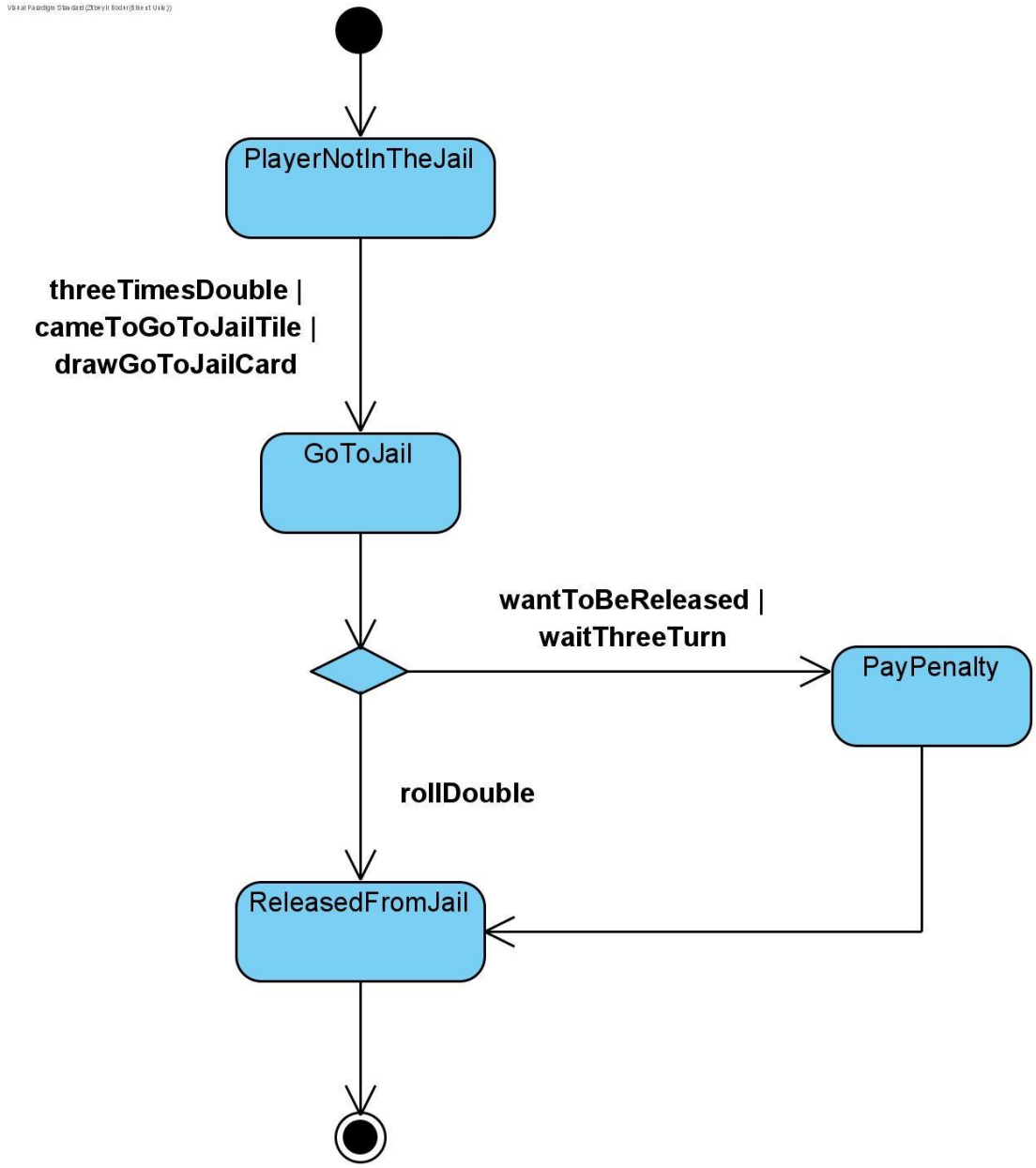


Figure 11 : State of a player in case of Jail

3.4.3.2.3. Mortgage

A property has two mortgage situations to be or not and the player can change its situation with necessary actions.

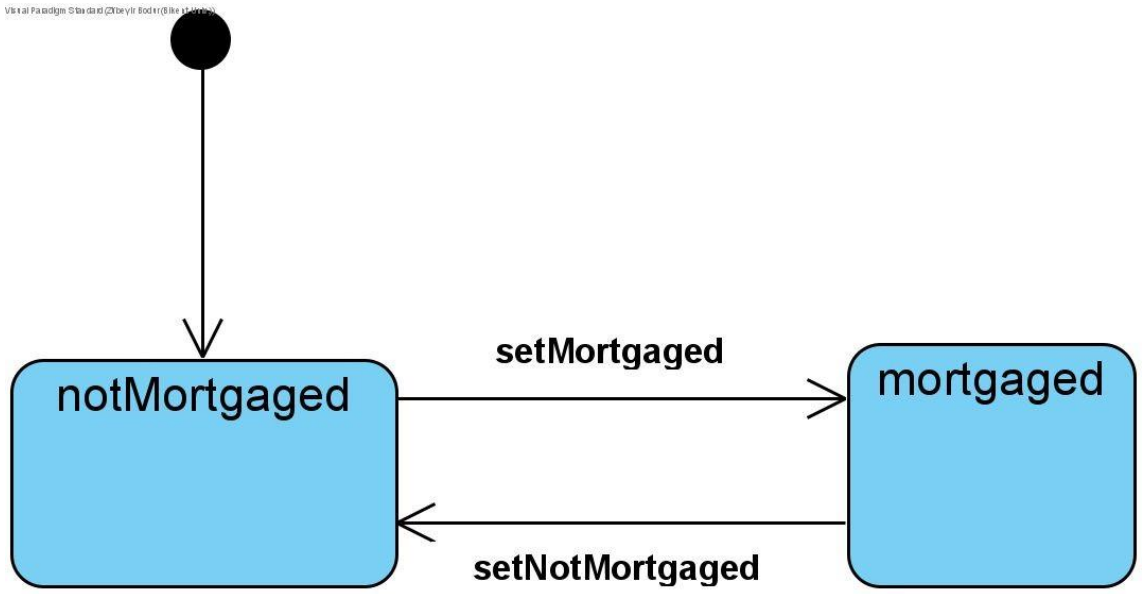


Figure 12 : State of a property with mortgage operations

3.4.3.2.4. Building

In the following diagram, a story of a building is described in states. Those are separated two at first, when a building is in the bank and in a player.

When the building is in the bank, it can be purchased by another player, regardless of how they purchase (by auction or by simply buying).

After it is purchased, the building's color group is initially incomplete. For a color group to be complete, all buildings of the same color group should be collected by the same player. This is shown by the *colorGroupComplete* transition. Moreover, if a building is mortgaged in a color group, that color group is considered to be incomplete. Until all of the mortgage in those building are lifted, it will remain as incomplete. By lifting all mortgages, the building also makes a *colorGroupComplete* transition, again.

If a building's color group is complete, they can start building classrooms. In order for a lecture hall to be built, all the buildings in the same color group should have four classrooms built in it. Unlike Monopoly™, in Monopoly Bilkent Edition, players don't have to build evenly for classrooms, so the differences in number of classrooms are ignored.

As a player constructs a classroom, or a lecture hall, none of the buildings in that color group can be mortgaged or traded. The player needs to first sell all the learning environments the color group contains.

If a player trades his building (*buildingTraded* transition) or loses his building (*buildingLost* transition), by possibly bankrupting by paying rent, to another player, the building will be given to the player as is. In other words, the mortgage state won't change.

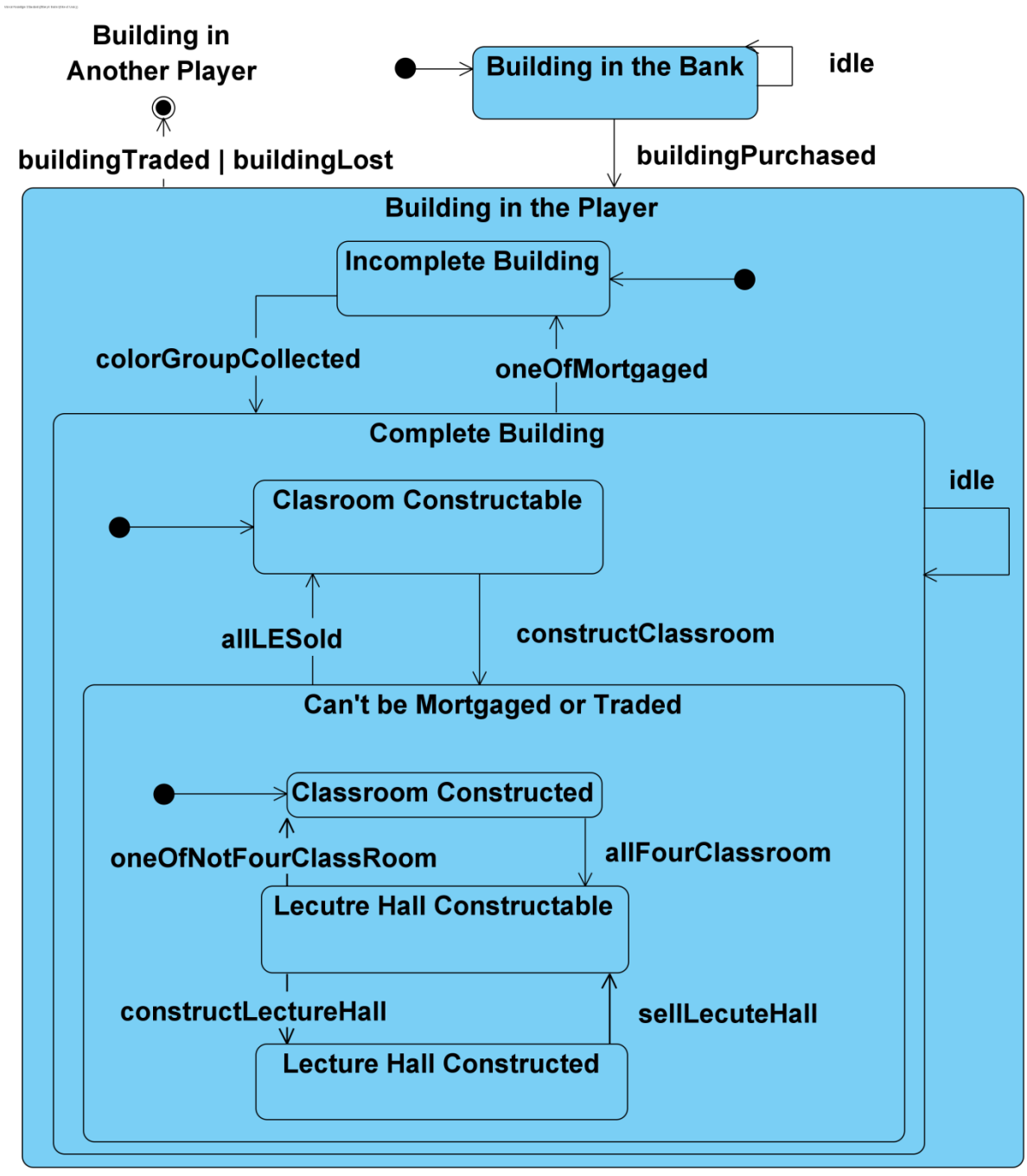


Figure 13 : Nested state diagram of a building

3.4.3.4. Sequence Diagrams

3.4.3.4.1. Drawing a Chance Card

Scenario: Player draws a chance card during the turn

During his/her turn, the user rolls dice and lands on a “Chance Card” tile. The diagram below shows how this turn is processed. First, the MonopolyGame class checks whether the landed tile is ChanceCardTile or not. If it is ChanceCardTile, it calls the processChanceCardTile() method of the Board class. Board class calls the draw() method of the chanceDeck which is an instance of Deck class. chanceDeck returns an instance of Card class and Board returns this card object to the MonopolyGame. MonopolyGame creates a DrawChanceCardAction according to this card and calls the act() method of this action. Finally, MonopolyGame calls the processCard(Card) method and processes the actions written in the card.

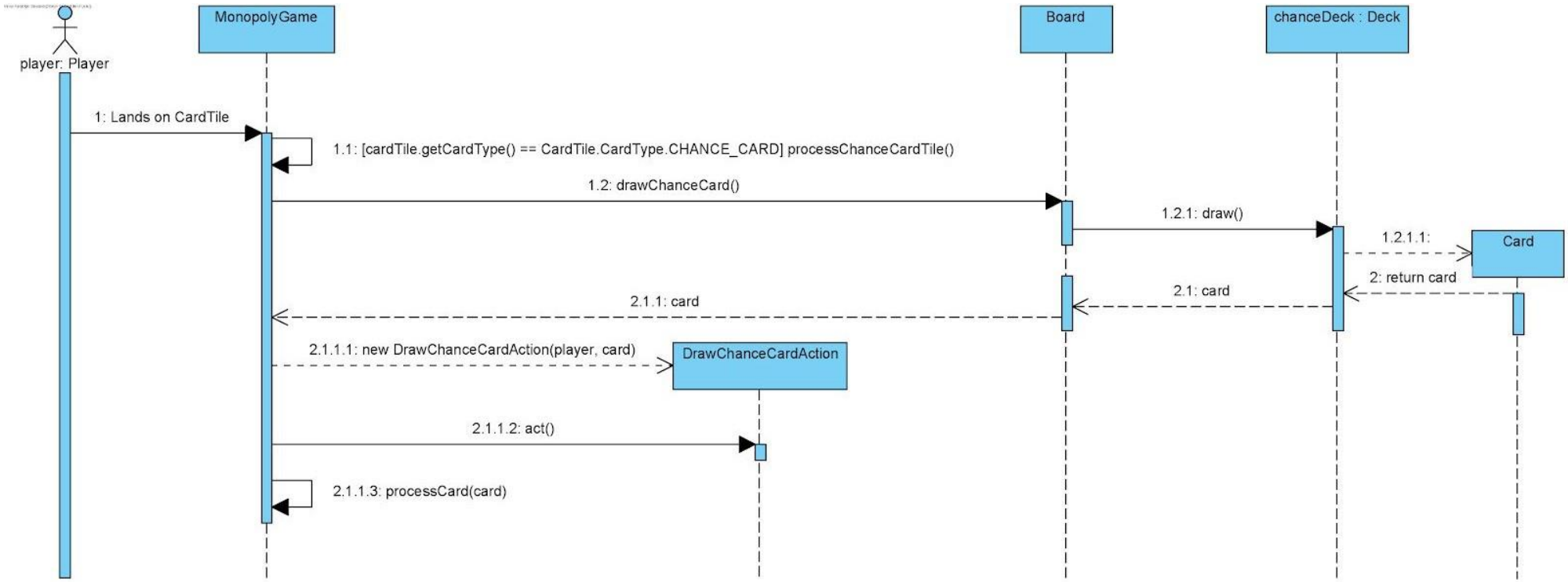


Figure 14: Player draws a chance card

3.4.3.4.2. Paying Rent

Scenario: Player pays the rent of a building which has one classroom.

In this scenario, the player lands on a building tile and the building has one classroom. The system gets the specified rent amount from the Building class and transfers the money to the property owner from the player by creating TransferAction and calling act() method of it.

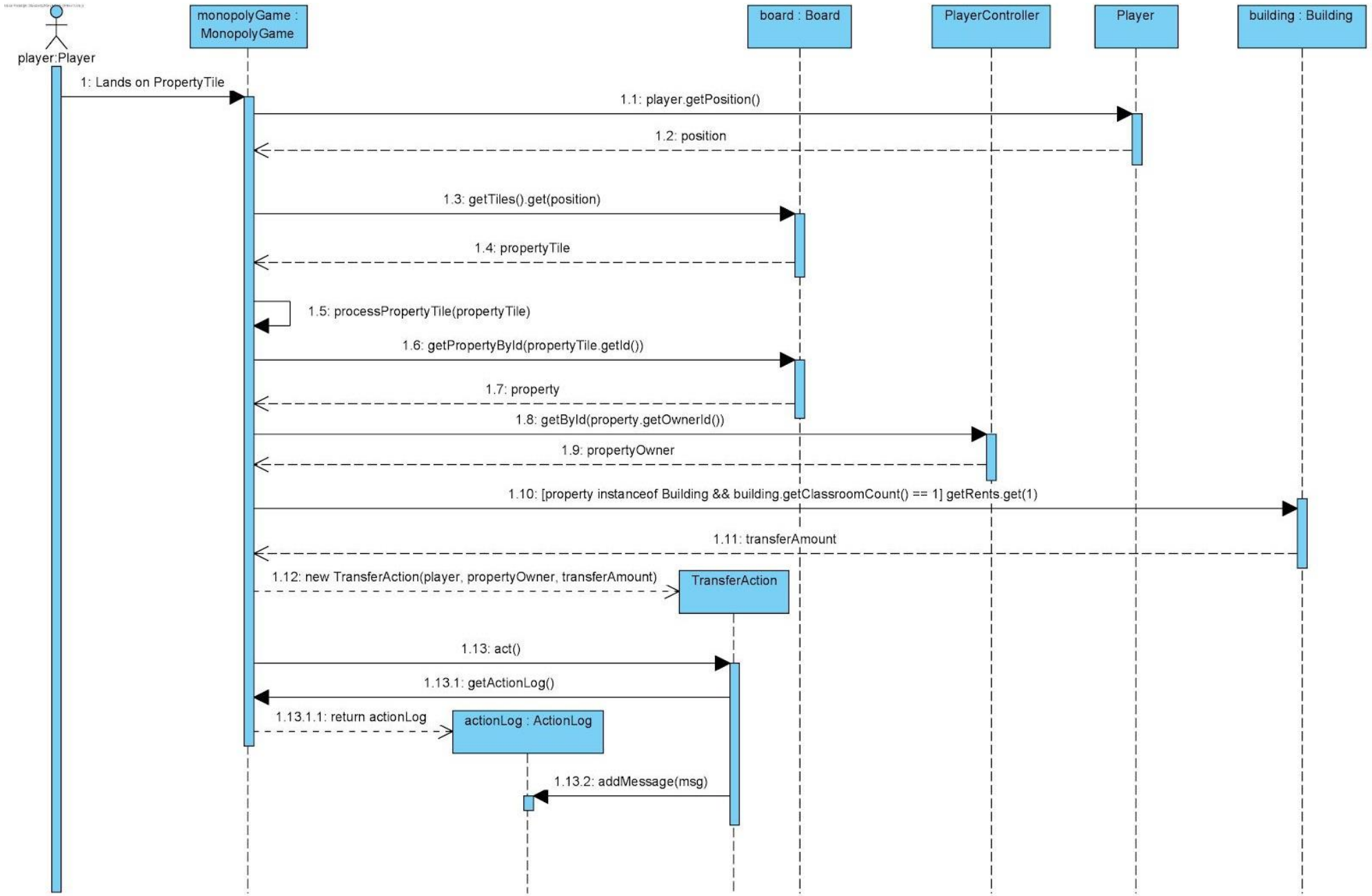


Figure 15: Player pays rent of a building which has one classroom

3.4.3.4.3. Going to the jail

Scenario: Player lands on Go To Jail tile and gets sent to jail by the system.

In this scenario, the player lands on the “Go To Jail” tile and he/she is sent to jail by the system. After the player lands on the tile, a new GoToJailAction class gets created and the player is passed as a parameter to this class. MonopolyGame calls act() method of this action class. GoToJailAction class sets the player’s isInJail attribute true and his/her position to 10 which is the “In Jail/Visit Jail” tile at the board. After these actions, GoToJailAction class gets the ActionLog from the MonopolyGame and enters the message that tells which player went to jail.

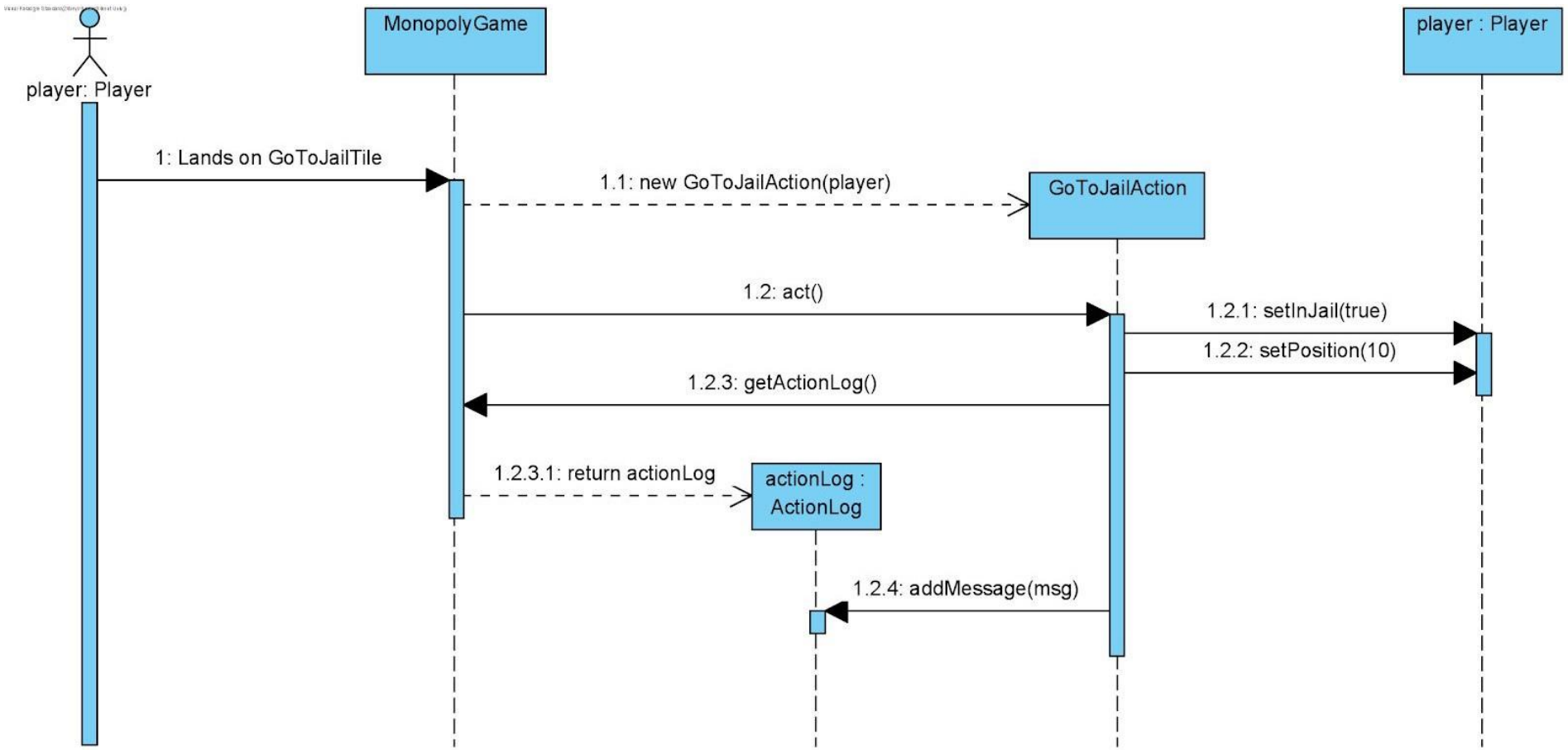


Figure 16 : Player lands on the Go-To-Jail Tile and goes to jail

3.4.3.4.4. Rolling dice and moving

Scenario: Player rolls dice and moves.

In this scenario, the player is not in jail and he rolls the dice. The dice outcome is 3-5 and speed die mode is disabled. Then, he/she moves by the sum of these two, 8.

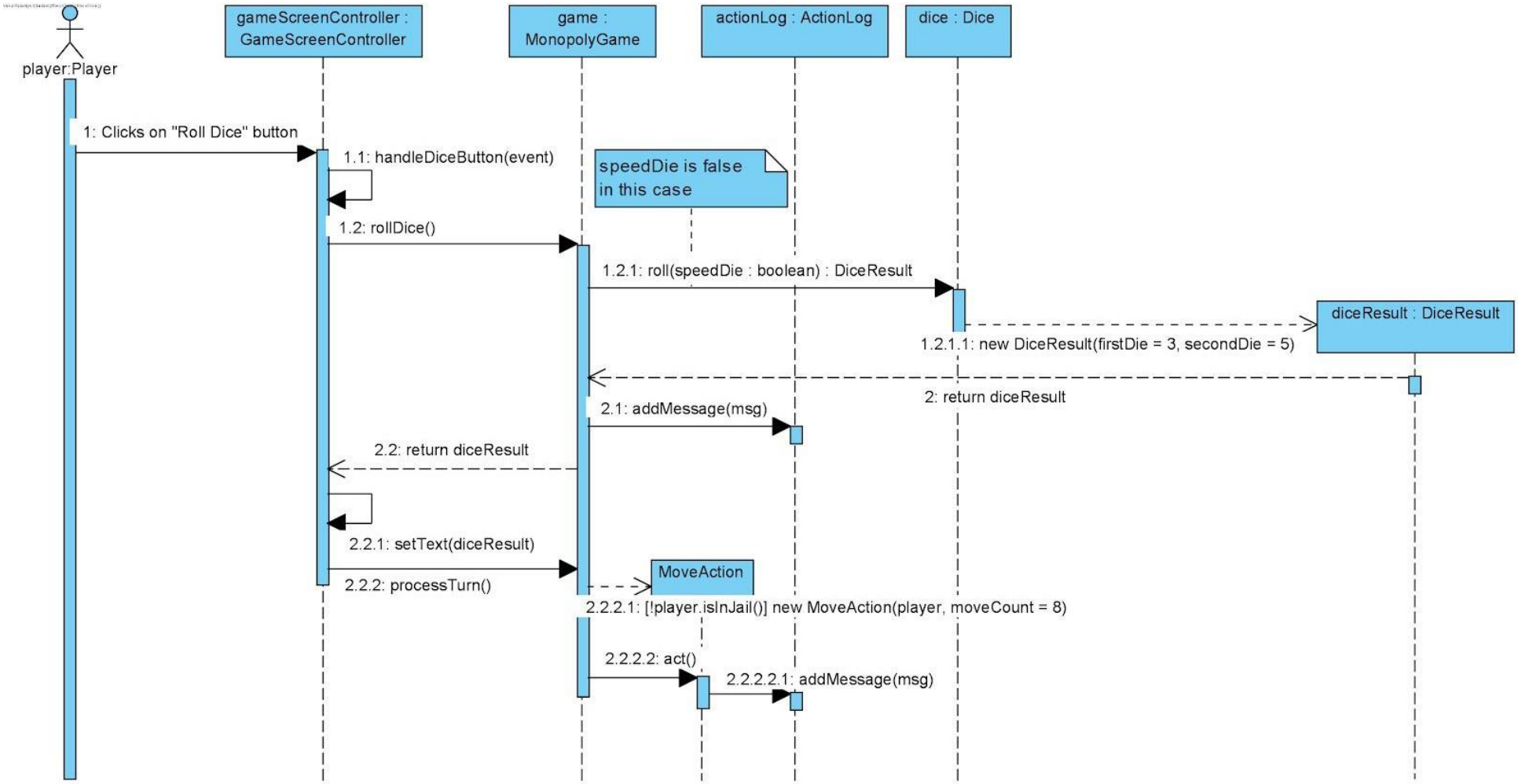


Figure 17 : Player rolls the dice, gets 3-5 and moves by 8 tiles

3.4.4. User Interface

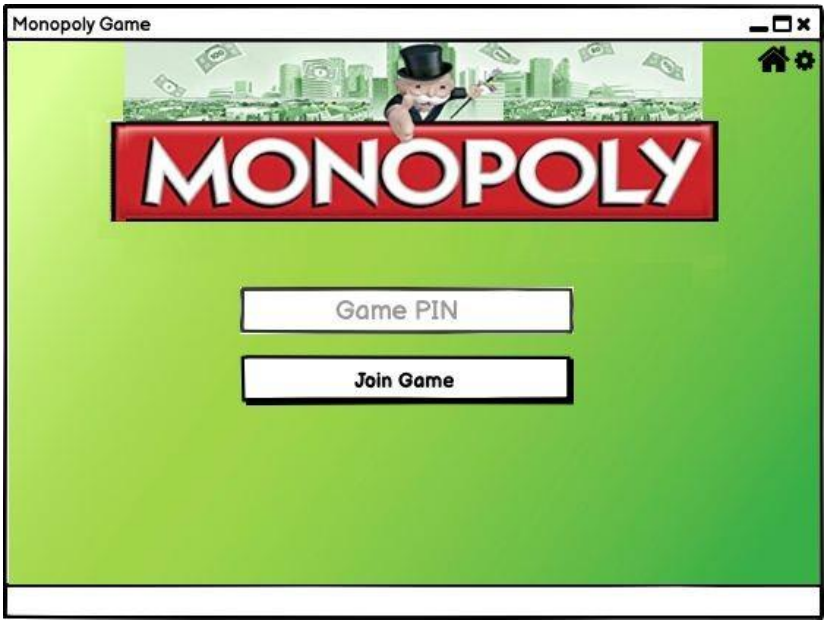
3.4.4.1. Navigation Path

- In the main menu, there are 5 buttons and one text field to write in a username. The first three buttons are not available if no username is specified
    - Whenever someone presses the “Create New Game” button, they go into a game lobby where they are the host.
    - “Join Game” button puts people to the join game screen where they are able to enter a pin to join a lobby as a non-host player.
    - “Load Game” button shows a list of saved games and players are able to host an unfinished saved game by clicking on it.
    - “How to Play” shows the rules of the game and “Options” shows options.
- In the game lobby, people are able to select their tokens and teams. Only the host of the game has access to the “start game” button and can change the speed die and teams options. The lock symbol near the pin is also only accessible by the host and disables other people from joining the lobby by making it private.

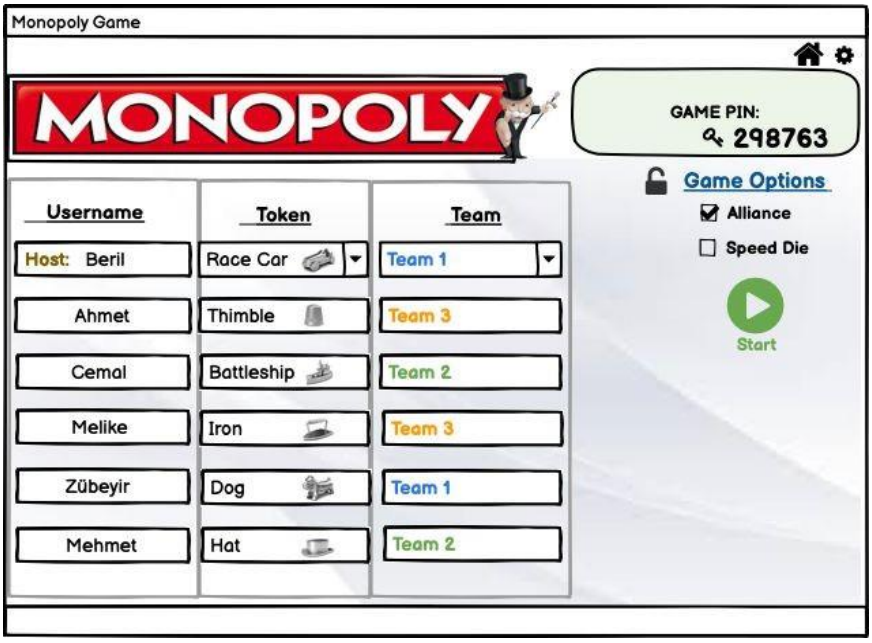
3.4.4.2. Screen Mockups

Main Menu:

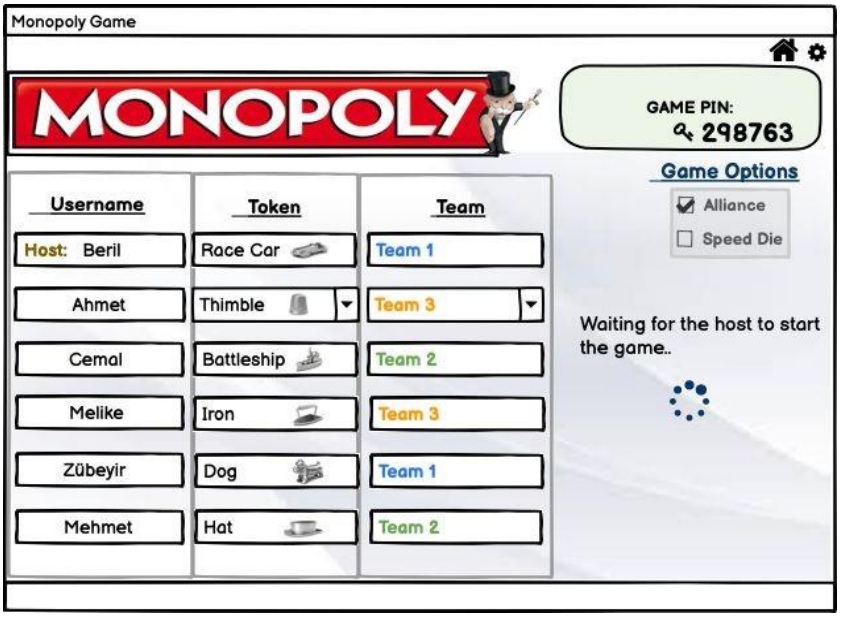
Join Game Screen:



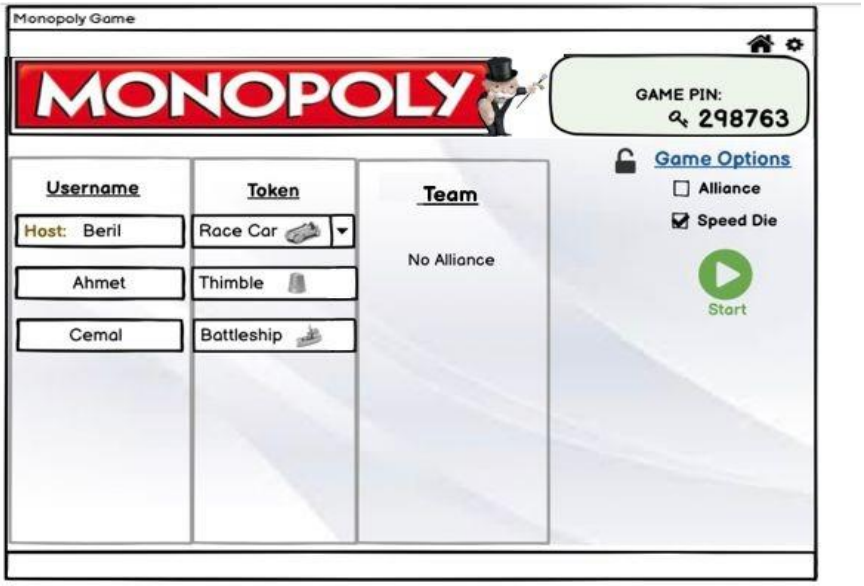
Game Lobby (For the Host):



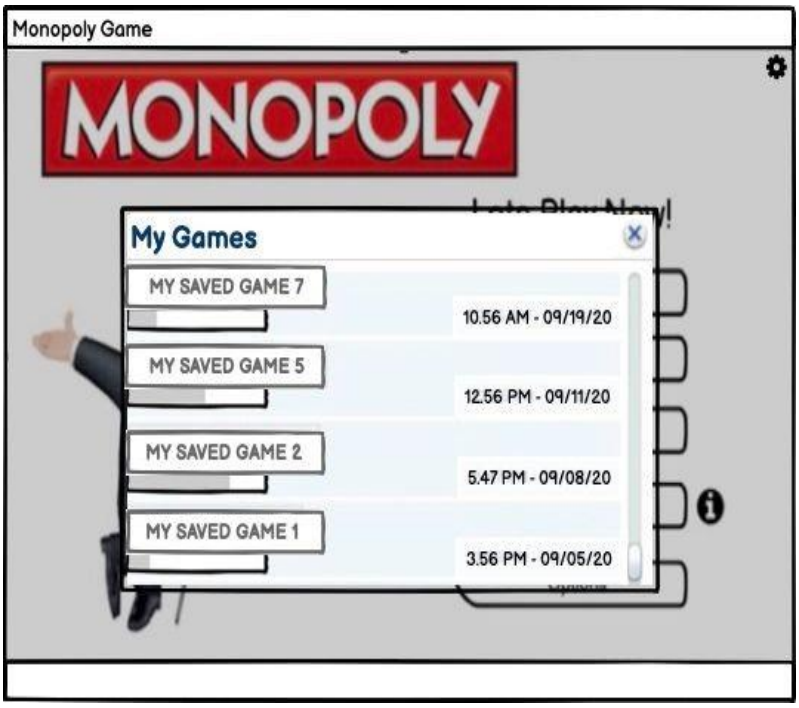
Game Lobby (For the Guests):



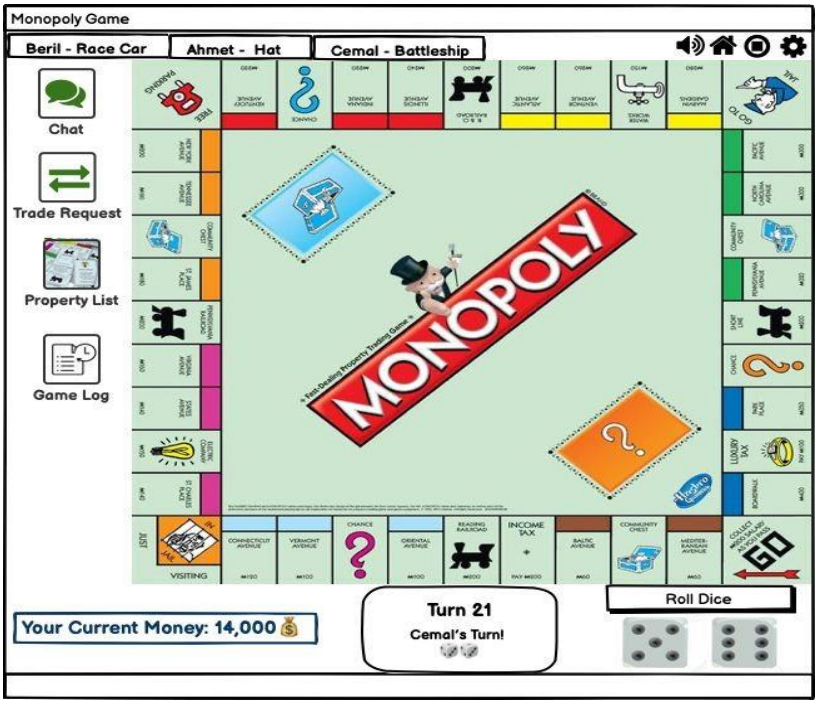
Game Lobby (Teams Disabled):



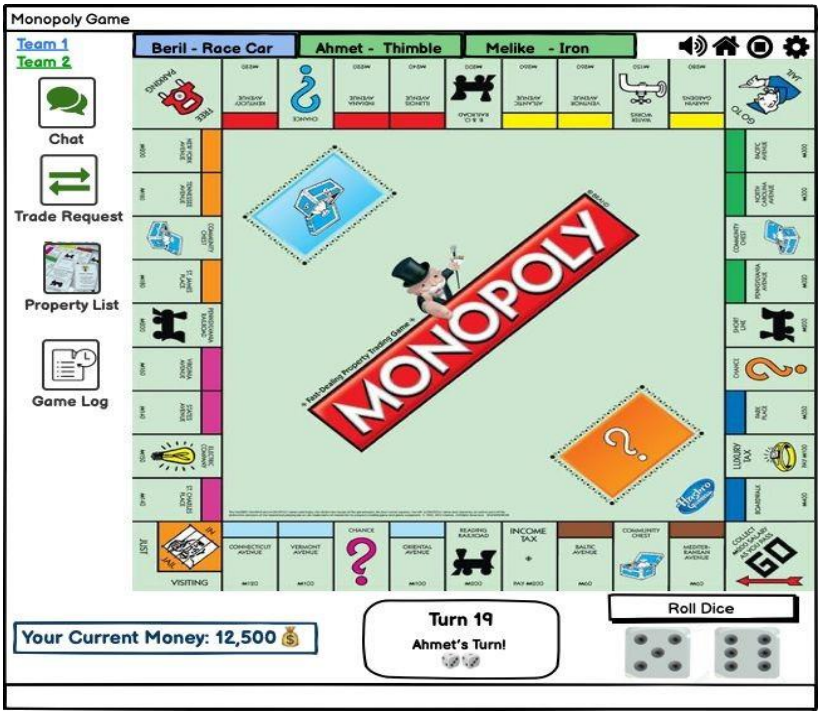
Load Saved Game:



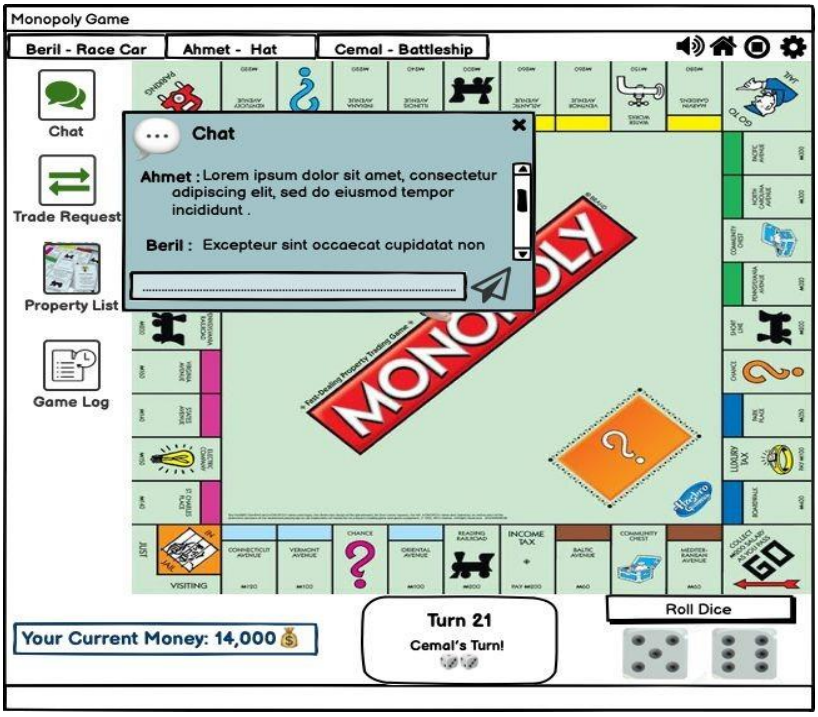
Game Screen:



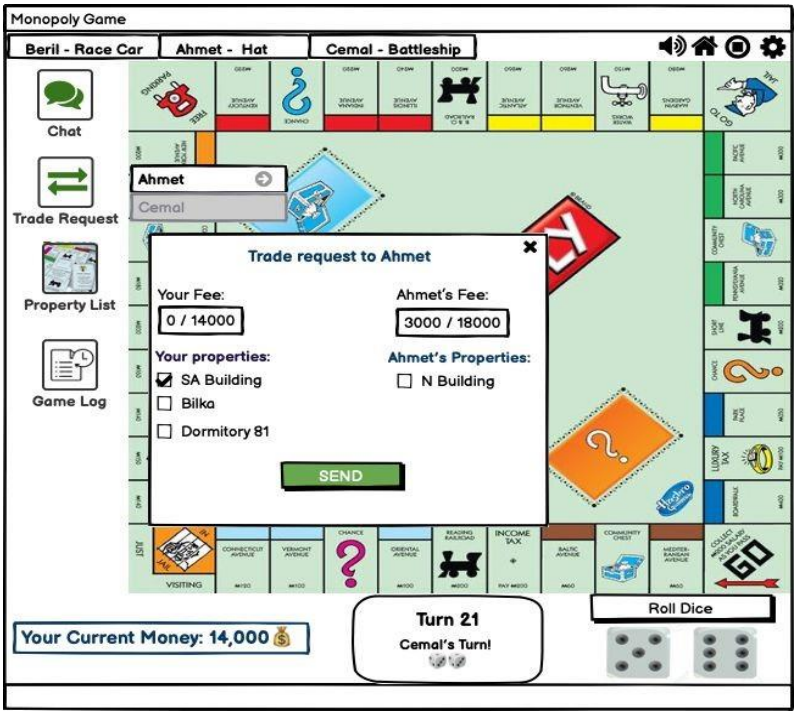
Game Screen (with Teams):



In Game Chat:

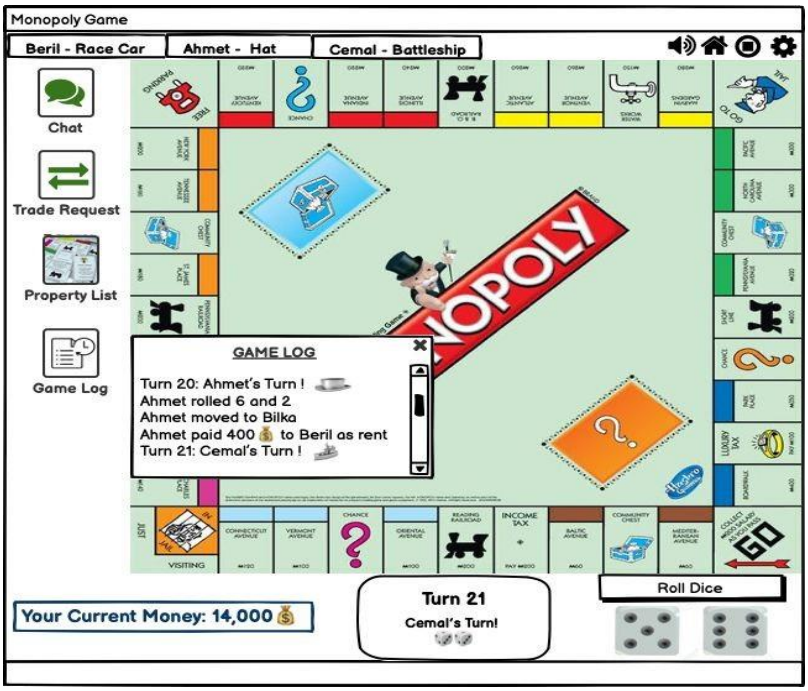


Send Trade Request:



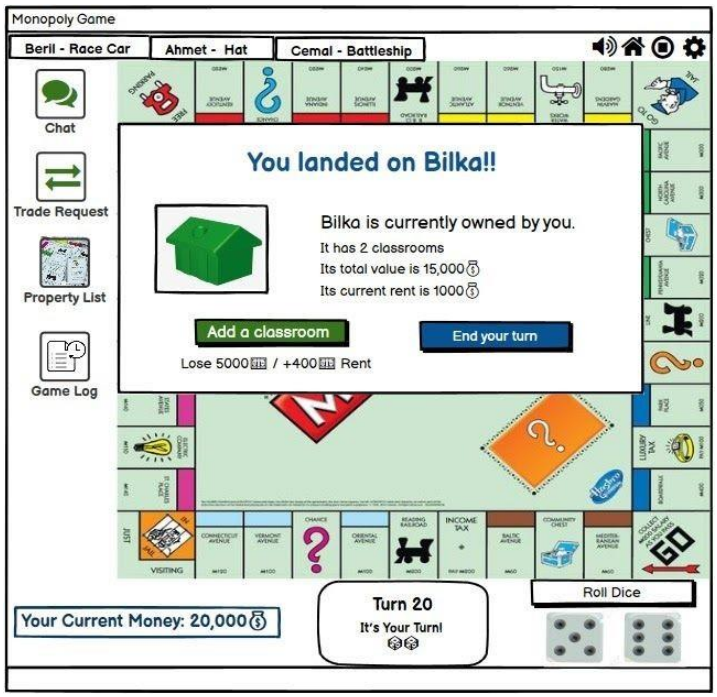
Trade Request Received:

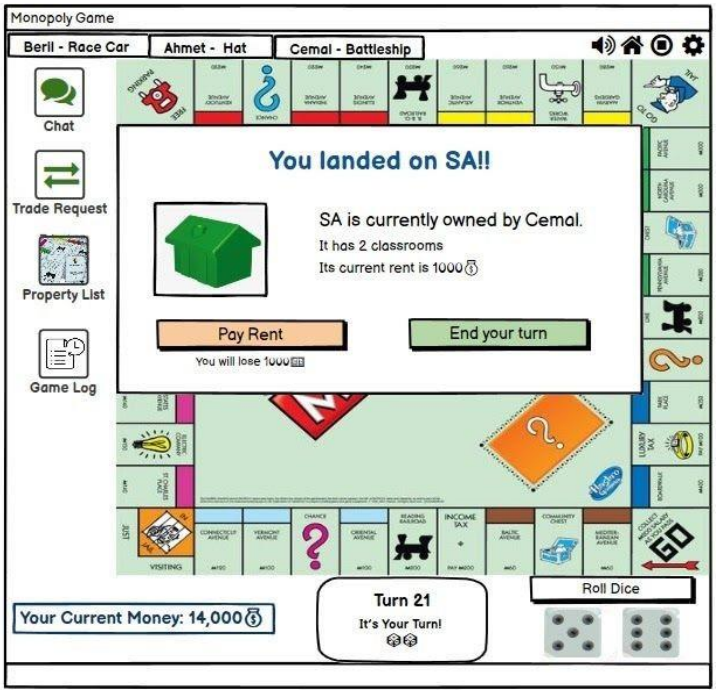
Game Log:



See Property List:

Game Notifications:





4. Improvement Summary

- Edited the dynamic model diagrams according to the feedback.
- Since it causes confusion to add house and hotel to a building, we changed house to the classroom and hotel to the lecture hall.
- Edited the use case diagram according to the feedback, added more separate use cases.
- Edited the functional and non-functional requirements sections according to the feedback.
- Added figure captions below pictures.

5. References

[1] “Speed Die”. Accessed on : Nov. 1, 2020. [Online]. Available: https://boardgamegeek.com/image/456407/monopoly.

[2] “Rules of Monopoly”. Accessed on : Oct 20, 2020 [Online]. Available: https://www.hasbro.com/common/instruct/monins.pdf