

Simplicity Tic-Tac-Toe Requirements Document

Version 1.5 November 11, 2024

**Project Members: Zack, Leif, Joseph, John, Hazal, Gary, Dave, and
Argjend**

Table of Contents

Project Overview	2
Purpose of Requirements Document	2
Project Scope	2
Glossary	3
Tic-Tac-Toe Rules	4
Target User Base.....	4
Requirements	5
Use Cases	7
Use Case 1: Starting a new match.....	7
Use Case 2: Game Flow.....	8
Use Case 3: Selecting a Theme.....	9
Use Case 4: Editing the Custom Theme	10
Use Case 5: Instant Match Replay	12
Use Case 6: Watching a Saved Replay	13
Use Case 7: Deleting a Saved Replay	13
Use Case 8: Saving a Replay	14
Use Case 9: Changing a Replay's Memo	14
Use Case 10: Toggling the Background Music	15
Use Case 11: Resetting the Score	15
Use Case Diagrams	16
Sequence Diagrams.....	17
Class Diagram.....	20

Project Overview

The following documentation details a distributable application for digitally playing Tic-Tac-Toe where the core features are customization options and a match replay system. The game is played on a 3x3 dimensioned grid between two users or a user and simulated opponent. Customization options change the graphical and auditory components of the grid. Matches that previously occurred and were saved can be watched with the replay system. The application will be built to run on Windows 10 and 11.

Purpose of Requirements Document

This document is designed to outline and guide the planning and development of the Tic-Tac-Toe application. Included are the sections project scope, glossary, the target userbase, requirements, use cases, and UML diagrams. The document's goal is to ensure the final product aligns with the project's goals.

Project Scope

Objective 1: Deliver a Tic-Tac-Toe application built to the specifications laid out in the technical sections.

Objective 2: Create a presentation for stakeholders showcasing results of objective 1, the build process, and deliverables.

Resources: A team of 8 computer scientists, 70 days, and a \$0 budget.

Amendment: ~ \$25 for pizza during a meeting, funded by Dave.

Deliverables:

- Requirements Document – October 10th 2024
- Object Document – November 4th 2024
- Application – December 5th 2024
- Installation Guide – December 5th 2024
- User Manual – December 5th 2024
- Presentation for stakeholders – December 5th 2024

Glossary

1. **Background:** This refers to the area under the grid, not the rest of the UI.
2. **Custom Theme:** A theme where the user has chosen their own combination of theme elements.
3. **Grid:** The 3x3-dimensioned gameboard consisting of 4 lines.
4. **Grid Color:** The color of the Tic-Tac-Toe grid lines.
5. **Match:** An instance of a single game of Tic-Tac-Toe. A new instance begins when the grid is generated and not yet filled with symbols. This means a rematch begins a new instance.
6. **Multiplayer:** A match played with no simulated opponent involved.
7. **Point:** A scoring unit, representing a victory in a match.
8. **Records:** The collection of persistent replays saved for replaying.
9. **Replay:** A step-by-step recording of a particular game of Tic-Tac-Toe that can be replayed.
10. **Replaying:** Stepping through a replay move by move to watch a previously played game. The movement may be forward or backwards.
11. **Reset:** The score for each player is set to 0
12. **Score:** The summation of all points earned by one player
13. **Simulated Opponent** - The entity that takes the place of the second player in the singleplayer game mode. It can play the game against the human player at three different difficulties.
14. **Singleplayer:** One user plays the game installed on their machine against a simulated opponent
15. **Sound Effects:** The noise the application plays when a symbol is placed in a valid spot. This is customizable.
16. **Stalemate:** A situation where the grid was filled completely without either player achieving the win condition.
17. **Symbol:** The visual representation of a player's move that appears in the boxes created by the grid. By default, these are X's or O's, but they are a customizable theme element.
18. **Theme:** The background, grid color, player symbols, and sound effects for placing a symbol.
19. **Theme Element:** One of the four customizable theme options defined under theme.
20. **Tic Tac Toe:** Rules defined in the next section.
21. **Player:** A general term referring to a singular user
22. **Players:** A general term referring to either two humans, or a human and a simulated opponent
23. **Win tracking:** The visual display of both player's scores on the GUI displaying the match.

Tic-Tac-Toe Rules

Two players take turns marking spaces on a 3x3 grid. Each turn, a player marks exactly one empty space on the grid. Once a space on the grid has been marked, it can no longer be changed. Each player marks empty spaces with their respective symbols. Before the start of the game, it is decided (either randomly or by the players), who will be assigned to which symbol, of which the defaults are X and O.

At the start of the game, all spaces in the 3x3 grid will be unmarked. The player assigned to X will take their turn first. A player wins and the game ends if they place three of their symbols in a horizontal, vertical, or diagonal row. A stalemate occurs and the game ends if all spaces on the 3x3 grid have been marked and neither player has placed three of their symbols in a horizontal, vertical, or diagonal row.

Target User Base

1. Casual Tic-Tac-Toe Players

Players who are not concerned about the fine points of Tic-Tac-Toe. They may not care or know about the complexity of various strategies. These are people who would play for a very short amount of time and may not have the greatest computer hardware. The main appeal for these players would be customization, as they can create funny themes for fun.

2. Advanced Tic Tac Toe Players

Players who know about the more complex strategies of Tic-Tac-Toe. These players would play for longer periods of time and be invested in their growth and player skill. The main appeal for these players is the replay feature, as it allows them to track their skill over time. The customization features will support player retention for these players. They are also more likely to have higher quality computer hardware.

Requirements

Functional

- Application follows the rules of Tic-Tac-Toe outlined in the rule section
- Application will have a GUI that players will use to interact with the application
- Application will be delivered in a compressed file format
- No component of the application will require the Internet to function
- User can play Tic-Tac-Toe against a simulated opponent, referred to as singleplayer
- The simulated opponent has three distinct difficulties
- User can play Tic-Tac-Toe locally with another person, referred to as multiplayer
- After completion of a Tic-Tac-Toe game, players will have the option to play again
- A score will be kept of each player's wins only in multiplayer
- Stalemate does not count as a point for either player.
- Points will not be persistent upon exit of application
- Players will have the ability to manually reset score count
- Players can replay their matches
- Players can save replays at the end of a match in both single/multiplayer modes
- User can delete saved replays
- Application will be able to play music and sound effects.
- The user can disable music, sound effects, or both
- The user can select a default theme or create a custom theme which visually and auditorily changes aspects of the game
- Themes and records are persistent upon exit of the application.
- Default themes can't be deleted by user
- Tic-Tac-Toe player symbols will be customizable

Nonfunctional

-Performance

- Application will launch within 5 seconds.
- Application will close within 2 seconds.
- Application will take up no more than 2 GB of device storage.

-Compatibility

- System will not interfere with other applications or processes

-Maintainability

- Code should be well-documented and incorporate good coding practices to facilitate future updates and enhancements

-Quality

- Application must be installed without any defects

-Reliability

- Application should have a maximum downtime of 5 minutes for updates
- Critical failure rate should be under 1% occurrence

-Data Integrity

- Records should be accurate and consistent at all times
- Score should be accurate and consistent at all times

-Usability

- Application should be easy for first-time users to interact with

-Portability

- Application will support running on Windows 10 and 11

-Documentation

- Documentation should be thorough for future reference

Use Cases

Use Case 1: Starting a new match

Use Case 1	Starting a new match
Actor	User
Use Case Overview	The user has opened the application and wishes to start a game. The user will make several decisions defining what their match will look like.
Trigger	Starting the application and navigating into a match
Precondition 1	Application is installed
Precondition 2	Application is on the screen launching will default to
Main Success Path	
Description	The user decides to select a multiplayer game, does not change their name, and goes first.
1	Choose “start game” command
2	Select “multiplayer” command
3	Indicate the default names of Player1 and Player2 are fine
4	Choose to not change default of player 1 going first
Termination	The game begins in multiplayer with the names Player1 and Player2 and a default themed grid. Player 1 will make the first move.
Alternative Flow 2	
Description	Describes the command process for choosing to play against a simulated opponent.
2A1	Select “singleplayer” command
2A2	Select one of the three difficulties for the simulated opponent
Termination	The resulting game after finishing the rest of the steps will be against a simulated opponent at the specified difficulty.
Alternative Flow 3A	
Description	Describes the process for changing the default player names in multiplayer
3A1	Enter the desired name for player 1 or leave default
3A2	Enter the desired name for player 2 or leave default
3A3	Indicate the changes are satisfactory to proceed
Termination	The resulting game after finishing the rest of the steps will be a multiplayer game with one or both customized names.
Alternative Flow 3B	
Description	Describes the process for changing the default player names in singleplayer
3B1	Enter the desired name for player 1
3B2	Enter the desired name for the simulated opponent or leave default
3B3	Indicate the changes are satisfactory to proceed

Termination	The resulting game after finishing the rest of the steps will be a singleplayer game with the user and or simulated opponent names customized.
Alternative Flow 4A	
Description	Choose to have player 1 not go first
4A1	Choose for player 1 to go second
Termination	The resulting game will be to the specifications from previous choices with player 2 or the simulated opponent going first.
Alternative Flow 4B	
Description	Randomly choose who goes first
4B1	Select the option for the application to randomly decide who goes first
Termination	The resulting game will be to the specifications from previous choices with a randomly chosen player or simulated opponent going first.

Use Case 2: Game Flow

Use Case 2	Game Flow / Playing a Match
Actor(s)	User(s)
Use Case Overview	The user has specified a match to their preferred settings and will now play a game of Tic-Tac-Toe.
Trigger	User successfully finished use case 1.
Precondition 1	Settings from use case 1 have been set
Main Success Path	
Description	Detailed is the flow of a match without any attempts to place a symbol in an already occupied grid space. Assumes player 1 goes first. (The use case does not change even if player 2 goes first or if player 2 is the simulated opponent)
1	Player 1 places their symbol in an unoccupied grid space
2	Player 2 places their symbol in an unoccupied grid space
3	Repeat steps 1 through 2 until all the grid spaces are filled up or a user achieves the win condition of three of their symbols in a row
Termination	The match concludes, displaying the result.
Alternative Flow 1A / 2A	
Description	User currently taking their turn attempts to place their symbol in an occupied grid space
1A1 / 2A1	User attempts to place their symbol in an occupied grid space
1A2 / 2A2	User is informed in some way that the space is occupied
Termination	The user making their move keeps their turn and is allowed to attempt another move.

Use Case 3: Selecting a Theme

Use Case 3	Selecting a Theme
Actor	User
Use Case Overview	The user selects which theme will be applied to the grid before going through the process of starting a new match.
Trigger	User decides to change theme
Precondition 1	Application has been launched
Precondition 2	Application is on the main menu
Main Success Path	
Description	The user decides to select a theme in the menu to apply to the grid. This needs to be done before going through the process of starting a match.
1	User selects the “choose a theme” command
2	User chooses a theme from one of the preset options or the custom theme
3	Once satisfied with their selected theme the user chooses a command to return to the main menu
Termination	The selected theme is applied to the grid when a match is successfully started.
Alternative Flow 2A	
Description	The user attempts to select the custom theme which has not yet had all the theme elements set.
2A1	The program informs the user in some way that the custom theme is currently unavailable, as it has not been properly filled
Termination	Returns user to the menu where they can select a theme
Alternative Flow 3A	
Description	The user is not happy with the theme they’ve chosen and selects another option.
3A1	User chooses another theme from one of the preset options or the custom theme
3A3	Once satisfied with their selected theme the user chooses a command to return to the main menu
Termination	The selected theme is applied to the grid when a match is successfully started.

Use Case 4: Editing the Custom Theme

Use Case 4	Editing the Custom Theme
Actor	User
Use Case Overview	The user decides to edit the custom theme.
Trigger	User decides to change theme
Precondition 1	Application has been launched
Precondition 2	Application is on the main menu
Main Success Path	
Description	The user navigates into the theme editing menu. The assets they created are all correctly sized and of the correct file types. All four of the theme elements are changed.
1	User selects the “edit custom theme” command
2	User selects the “change background” command
3	From the file dialog that opens, the user chooses a custom asset to load which meets the specifications laid out in the user’s manual
4	User selects the “change grid color” command
5	From the color selector dialog that opens, the user chooses what color they want for the grid lines
6	User selects the “change symbols” command
7	User selects the “change player 1 symbol” command
8	From the file dialog that opens, the user chooses a custom asset to load which meets the specifications laid out in the user’s manual
9	User selects the “change player 2 symbol” command
10	From the file dialog that opens, the user chooses a custom asset to load which meets the specifications laid out in the user’s manual
11	User selects command to return to the editing menu
12	User selects the “change sound effect” command
13	From the file dialog that opens, the user chooses a custom asset to load which meets the specifications laid out in the user’s manual
14	When satisfied with changes, the user chooses a command to return to the main menu
Termination	The newly created custom theme is now available for selection in its specific command before starting a new match.
Alternative Flow 2A	
Description	User chooses to not select the “change background” command at all
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 3A	
Description	The file that the user has chosen is deemed unusable by the application.
3A1	User is informed that the custom asset selected does not meet the specifications laid out in the user manual

Termination	The user is returned to the edit theme menu where they can choose what to do next.
Alternative Flow 4A	
Description	User chooses to not select the “change grid color” command
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 6A	
Description	User chooses to not select the “change symbols” command
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 7A	
Description	User chooses to not select the “change player 1” command
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 8A	
Description	The file that the user has chosen is deemed unusable by the application.
8A1	User is informed that the custom asset selected does not meet the specifications laid out in the user manual
Termination	The user is returned to the edit theme menu where they can choose what to do next.
Alternative Flow 9A	
Description	User chooses to not select the “change player 2” command
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 10A	
Description	The file that the user has chosen is deemed unusable by the application.
10A1	User is informed that the custom asset selected does not meet the specifications laid out in the user manual
Termination	The user is returned to the edit theme menu where they can choose what to do next.
Alternative Flow 12A	
Description	User chooses to not select the “change sound effect” command
Termination	User carries on with the rest of the sequence in the main success path.
Alternative Flow 13A	
Description	The file that the user has chosen is deemed unusable by the application.
13A1	User is informed that the custom asset selected does not meet the specifications laid out in the user manual
Termination	The user is returned to the edit theme menu where they can choose what to do next.

For the writer's sanity, during the file dialog steps an alternate path is not included if the files are assets already used in the game for default themes. This is an intended feature, though it is not explicitly stated in the use case, but rather implicitly implied. This note is to illuminate this.

Use Case 5: Instant Match Replay

Use Case 5	Instant Match Replay
Actor	User
Use Case Overview	After the game ends, but before a rematch is selected, the user decides to watch a replay of the match.
Trigger	User wants to immediately watch a replay of the match
Precondition 1	A match has concluded without proceeding past the results screen
Main Success Path	
Description	The user decides they want to watch an instant replay before choosing what to do next on the results screen.
1	Player selects the "instant replay" command
2	The player can step back and forth through the recorded moves at their leisure
3	When done rewatching, the user selects a command to exit the replay and return to the post match results screen
Termination	

Use Case 6: Watching a Saved Replay

Use Case 6	Watching a Saved Replay
Actor	User
Use Case Overview	The user decides they want to watch a saved replay. This shows the process of doing so from the main menu of the application.
Trigger	User wants to watch a match replay
Precondition 1	At least one saved match replay exists within the game's files
Main Success Path	
Description	The user chooses a replay to watch after navigating into the menu to do so.
1	From the main menu, the user selects the watch replays command
2	User chooses a replay to watch based upon the displayed memo and metadata
3	User toggles back and forth through moves until satisfied
4	The user selects return command to return to the screen where the replays are displayed
Termination	The player returns to the screen displaying the replays, allowing them to either watch another or return to the main menu.

Use Case 7: Deleting a Saved Replay

Use Case 7	Deleting a Saved Replay
Actor	User
Use Case Overview	The user decides they want to delete a saved replay. This shows the process of doing so from the main menu of the application.
Trigger	User wants to delete a match replay
Precondition 1	At least one saved match replay exists within the game's files
Main Success Path	
Description	The user chooses to delete the match replay from inside of the application.
1	From the main menu, the user selects the watch replays command
2	User chooses a replay to delete based upon the displayed memo and metadata
3	User chooses the "delete" command on the replay rather than watch
4	The user selects yes on the prompt making sure they want to
Termination	The player returns to the screen displaying the replays, and the application has moved that specific replay to recycling.

Use Case 8: Saving a Replay

Use Case 8	Saving a Replay
Actor	User
Use Case Overview	The user decides they want to save a replay of the recently finished match. Details the process of doing so.
Trigger	The user wants to save a match replay.
Precondition 1	Be on the results screen after a match
Main Success Path	
Description	The user chooses to save the match replay with a memo.
1	From the results screen, the user chooses the “save replay” command
2	User enters as much or as little as they’d like into the memo prompt
3	User uses the “save” command
Termination	The player returns to the match results screen, now with the save replay button greyed out so copies can’t be made.

Use Case 9: Changing a Replay’s Memo

Use Case 9	Changing a Replay’s Memo
Actor	User
Use Case Overview	The user decides they want to change the replay of a memo, this is the process to do so.
Trigger	User wants to edit a replay’s memo
Precondition 1	At least one saved match replay exists within the game’s files
Main Success Path	
Description	The user chooses to edit the memo of a specific replay.
1	From the main menu, the user selects the watch replays command
2	User chooses a replay to edit the memo of, and selects the “edit memo” command
3	User is given a text prompt with the current text displayed that they can edit as they like
4	The user selects return command to return to the screen where the replays are displayed when they’re done editing.
Termination	The player returns to the screen displaying the replays with the replay’s memo being updated.

Use Case 10: Toggling the Background Music

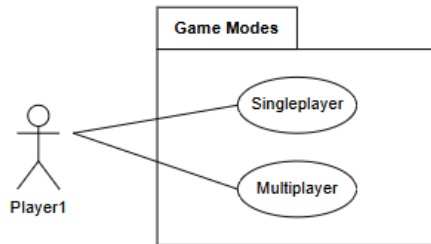
Use Case 10	Toggling the Background Music
Actor(s)	User(s)
Use Case Overview	During a match the user decides they want to toggle the background music on or off.
Trigger	User decides to toggle the background music
Precondition 1	Application has been launched
Precondition 2	Application is currently in a match
Main Success Path	
Description	The user decides they want the state of the background music to be switched.
1	User selects the toggle music button on the match screen.
Termination	The state of the music will be switched to the opposite of its current state. Silence or music will be enjoyed by the player!

Use Case 11: Resetting the Score

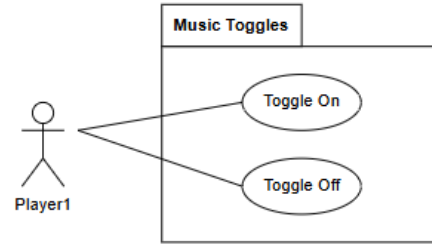
Use Case 11	Resetting the Score
Actor(s)	User(s)
Use Case Overview	After the game ends, but before a rematch is selected, the user decides to reset the score. If a second player is involved this action may or may not be agreed upon.
Trigger	User decides to reset the score
Precondition 1	At least one player's score is above 0
Precondition 2	A match has finished without proceeding to rematch
Main Success Path	
Description	The user decides they want the scores to be reset back to 0.
1	User selects the reset score button before hitting rematch or quitting.
Termination	The scores will be set back to 0 and the application will proceed to execute the user's selected command.

Use Case Diagrams

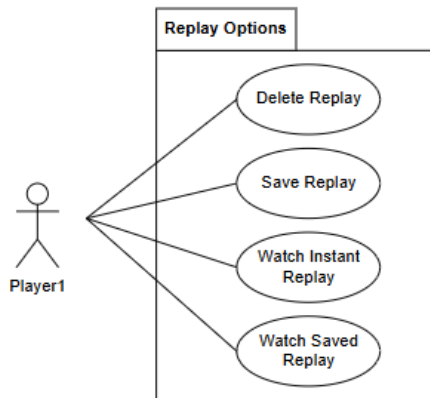
1) Showing Game Modes



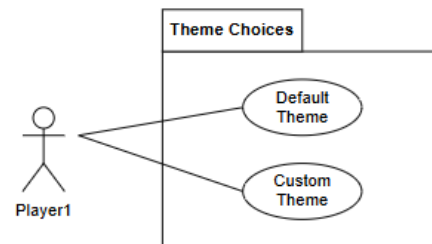
2) Showing Music Toggles



3) Showing Replay Options

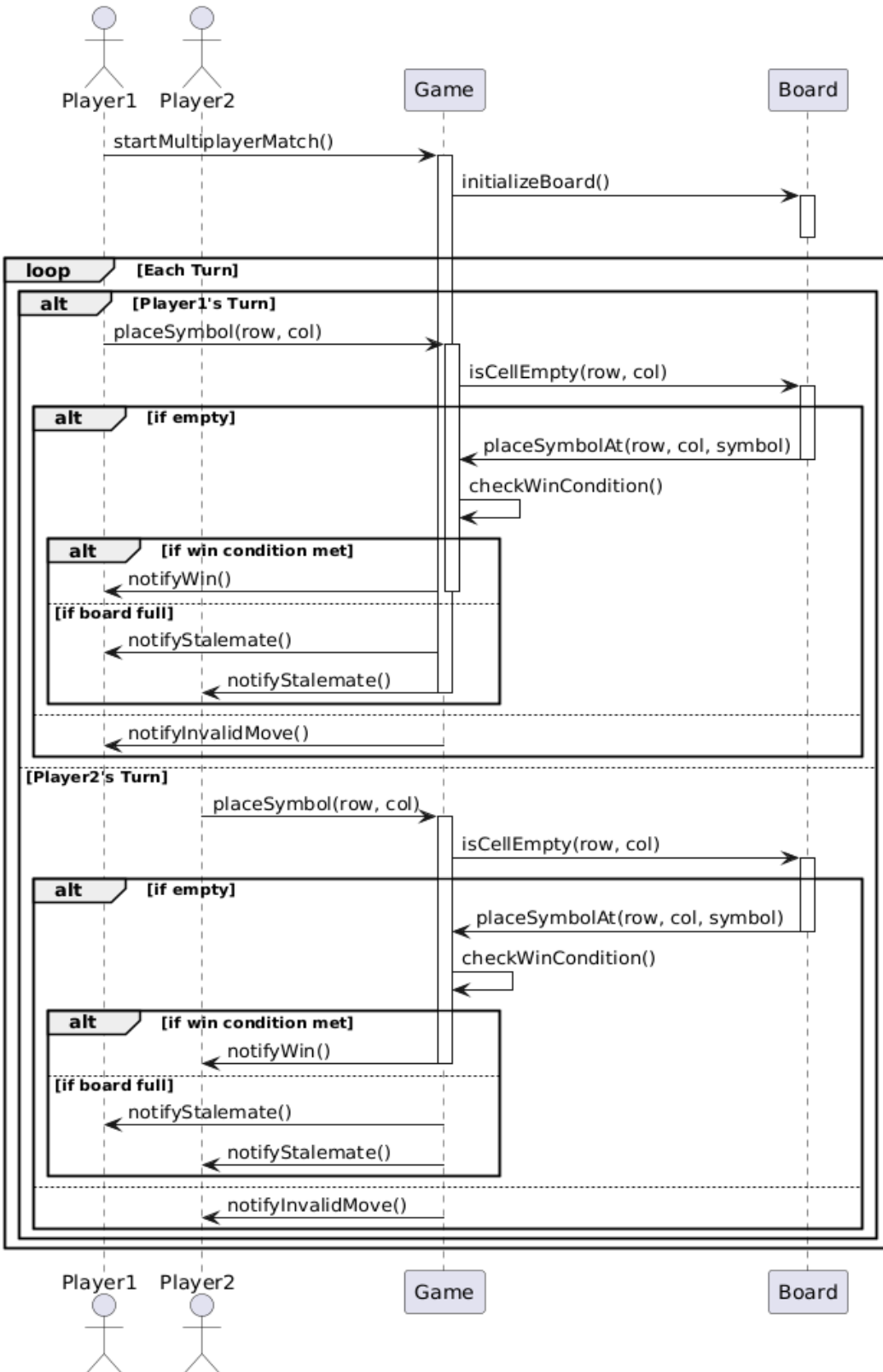


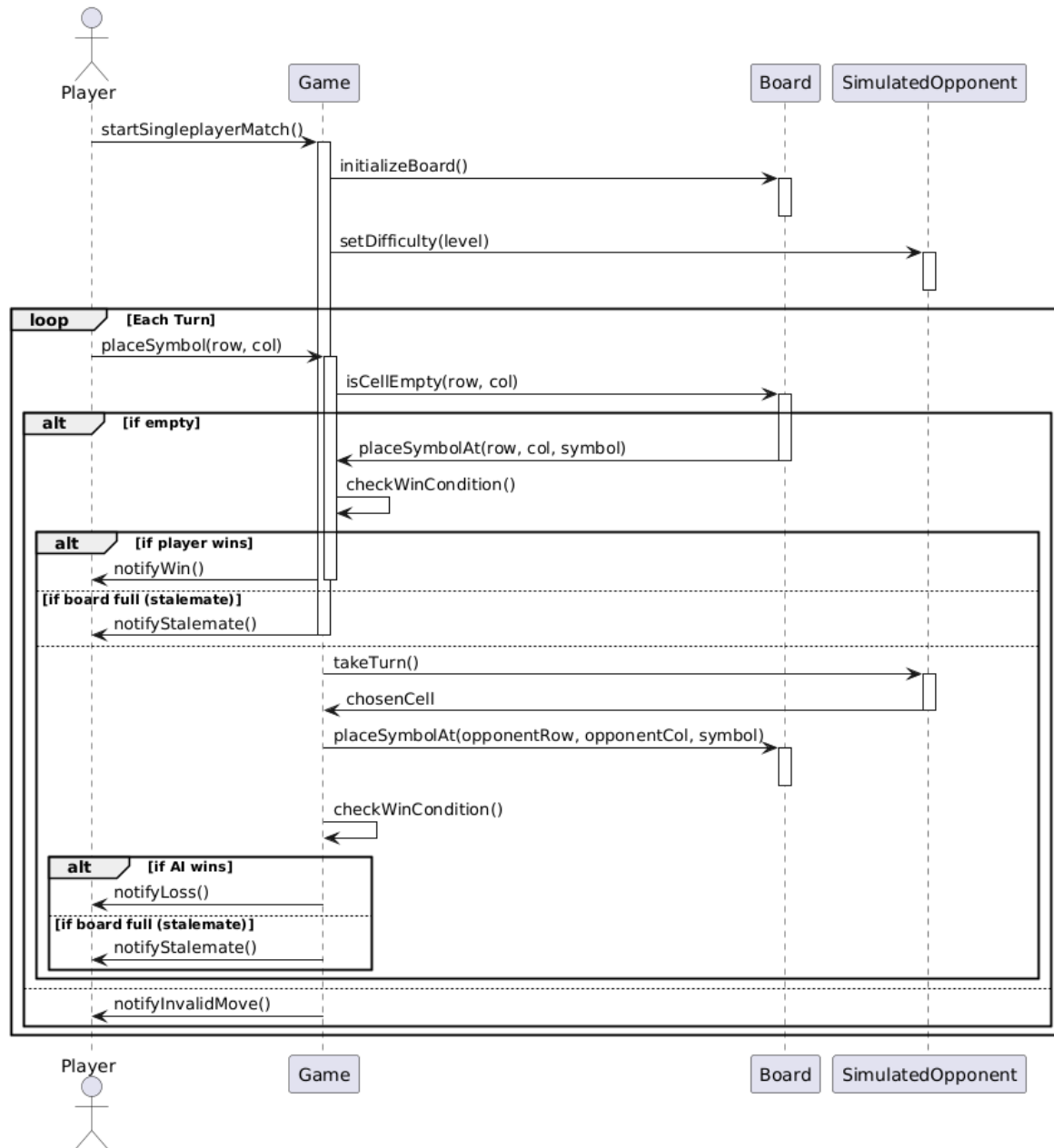
4) Showing Theme Options

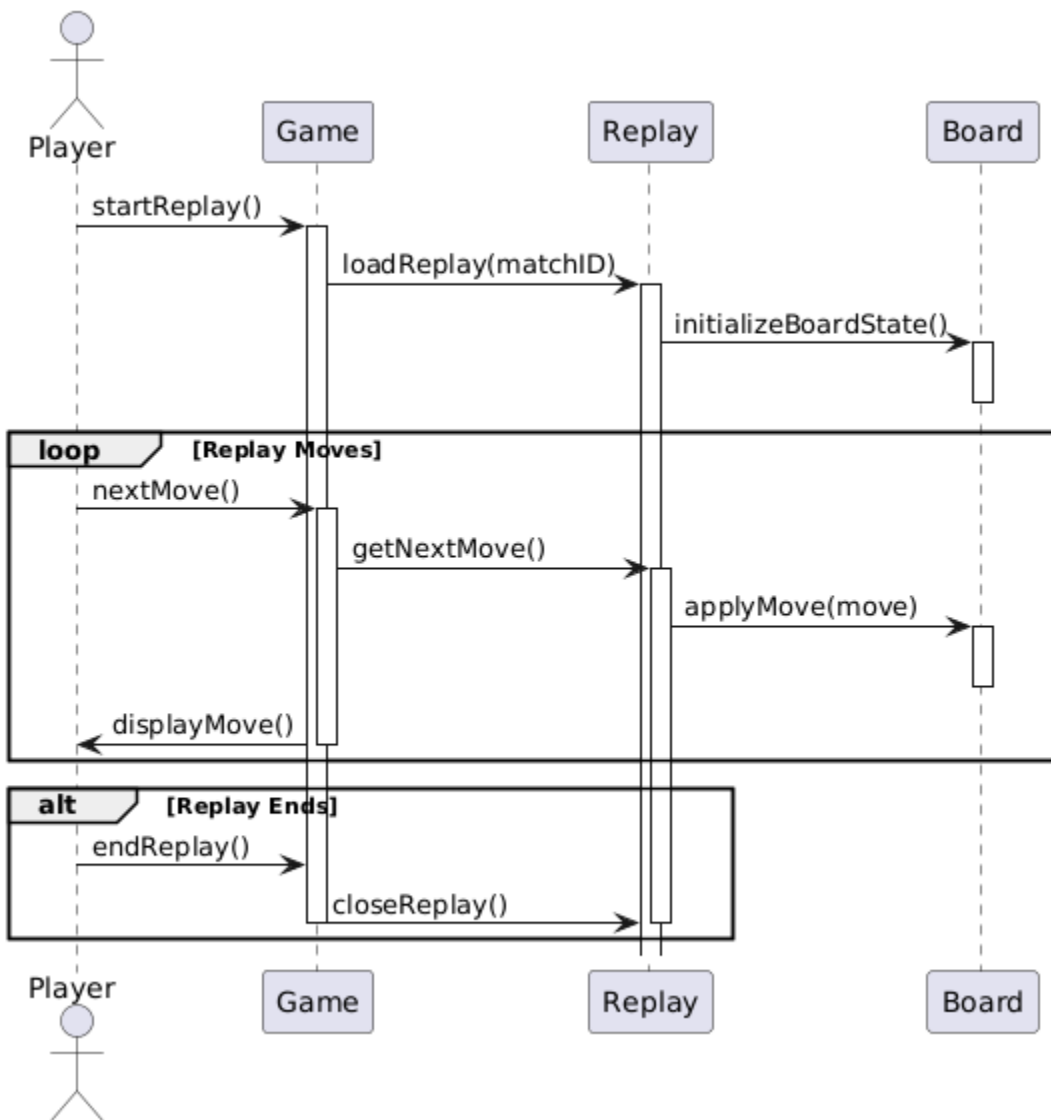


(Outdated, will be replaced by new diagrams for the current use cases)

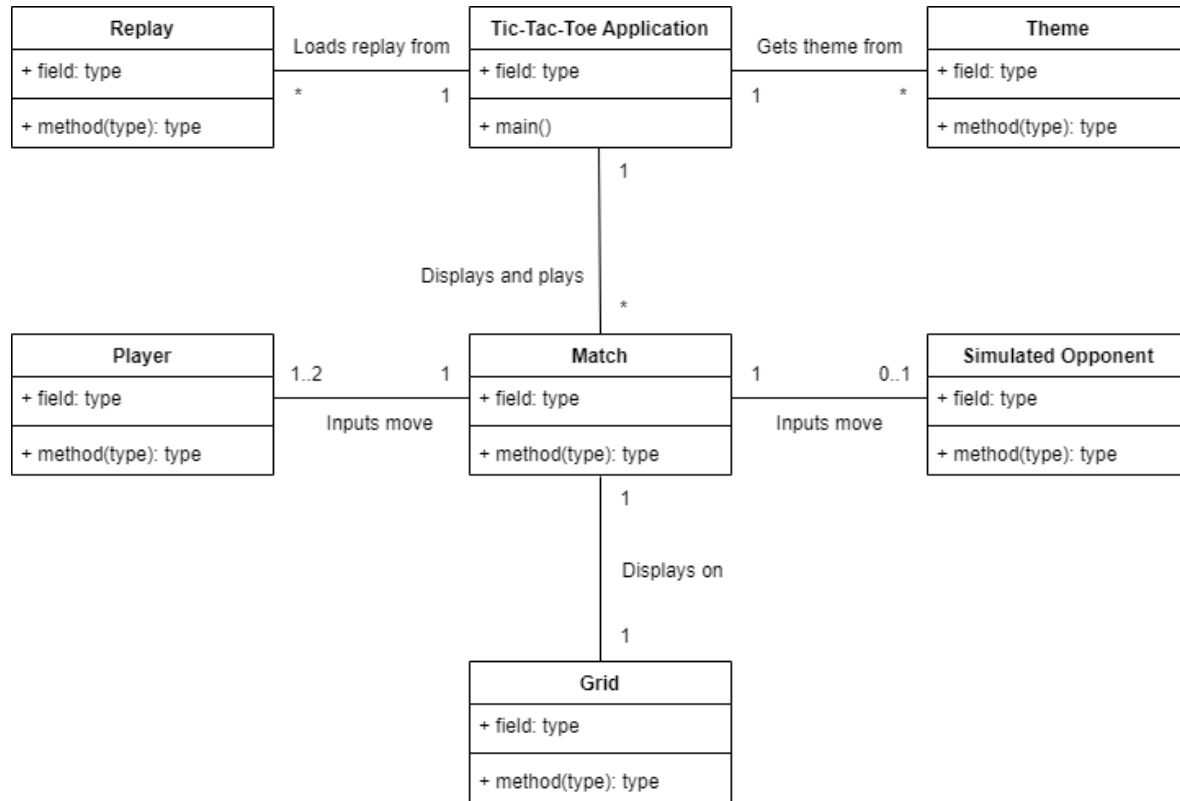
Sequence Diagrams







Class Diagram



(Up to date)