



# Requirements Analysis

## Abstract

Requirement analysis and specification for a Chess game with an 8x8 board

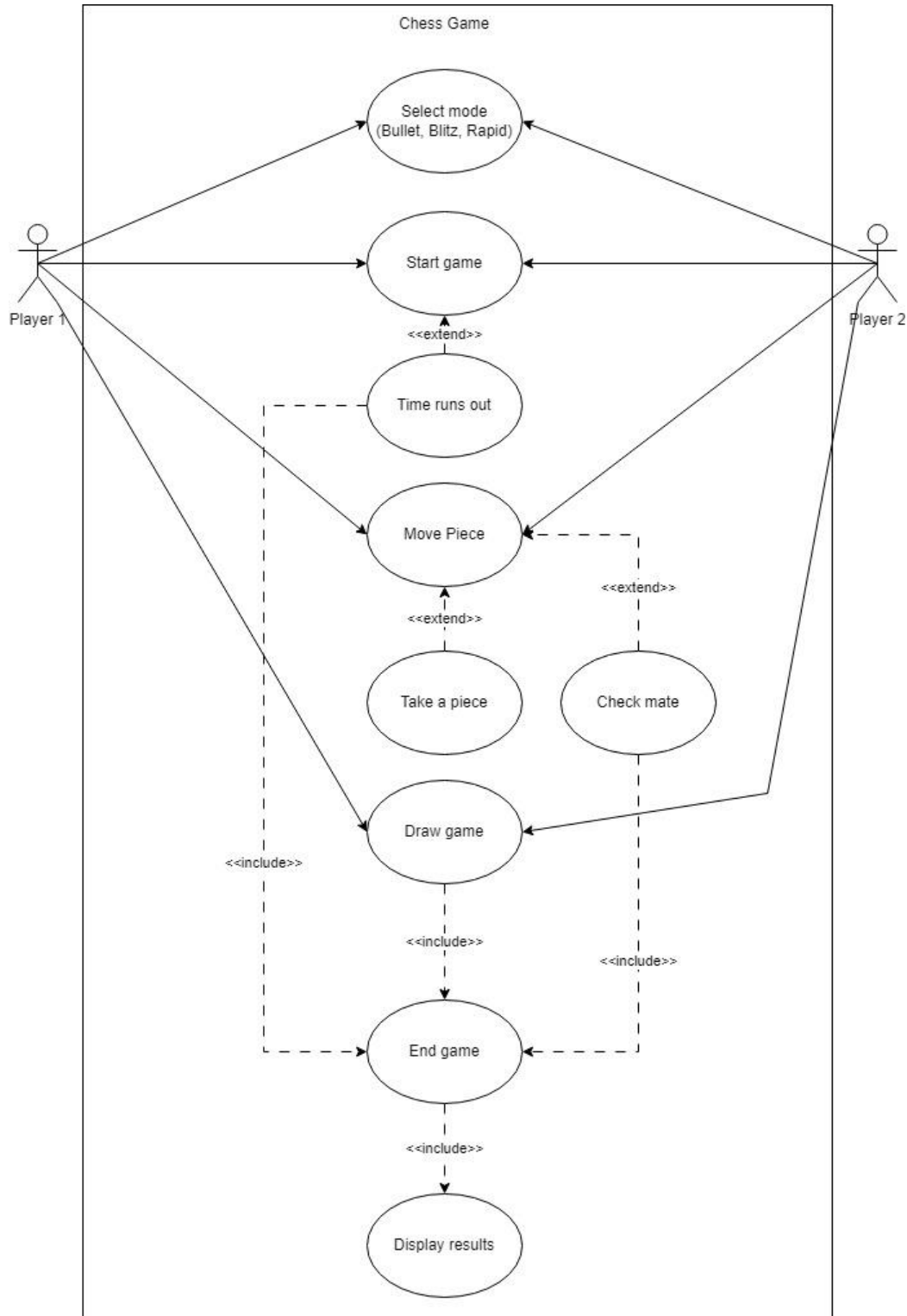
Barandiaran Rivera, Matias

## Exercise 1

1. The project topic chosen was a Chess game with an 8x8 board.

<b>Functional Requirements</b>	<ul style="list-style-type: none"><li>• Players should be able to choose which type of game mode they desire to play (Bullet 1-2 min, Blitz 3-5 min, and Rapid 10-30 min).</li><li>• Players must be able to start the game.</li><li>• Players must be able to move their colored pawns, queen, king, knights, rooks and bishops.</li><li>• Game must end after either black or white side are under checkmate.</li><li>• Player should lose a piece if the opponent takes it.</li><li>• Games must end after time runs out for either of the 2 players.</li><li>• Game must not allow either of the players to make “illegal” moves.</li><li>• The game must display the results at the end.</li><li>• The game must provide a button that allows the parties to draw the game if neither side can win because of repetition.</li></ul>
<b>Non-functional requirements</b>	<p><b>Quality:</b></p> <ul style="list-style-type: none"><li>• Usability: The game should be implemented with a mouse interface; the game should highlight the available tiles for a move to improve user experience.</li><li>• Reliability: The game needs to deal with exceptions to avoid any potential crashes.</li><li>• Performance: The game must run at least 60 frames per second.</li><li>• Supportability: The code should be developed in a structured object-oriented manner to support maintainability in the future.</li></ul> <p><b>Constraints:</b></p> <ul style="list-style-type: none"><li>• Implementation: The game needs to be implemented in Java, using the GUI swing package.</li></ul>

2. Use case diagram for the Chess game. “Include” relationships represent use cases that are always required to be executed within the context of another use case. While “extend” relationships represent an additional step that is not always required. In this case, the actors are player 1 and player 2.



## Exercise 2

<b>Use Case Name:</b> Select Mode (Bullet, Blitz, Rapid)	<b>ID:</b> UC-1	<b>Priority:</b> Medium
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes how a player can select the time mode for the chess game.		
<b>Trigger:</b> Player opens chess jar file and clicks on the appropriate time mode button.		
<b>Preconditions:</b> <ul style="list-style-type: none"><li>• Jar file is functional.</li><li>• Players have downloaded and installed the file on their device.</li></ul>		
<b>Basic Path:</b> <ol style="list-style-type: none"><li>1. The player opens the game's jar file and is presented with the main menu.</li><li>2. The player clicks on the "Select Mode" button.</li><li>3. System displays a list of the available modes (Bullet, Blitz and Rapid).</li><li>4. The user clicks on one of the modes from the list.</li><li>5. The system saves the user selection.</li></ol>		
<b>Alternative path:</b> <p>A5a. If the player changes their mind, they can click on the "Select Mode" button again.</p> <ol style="list-style-type: none"><li>1. The player is presented with the list of the available modes.</li><li>2. The player clicks on a different time mode.</li><li>3. The system updates the player's selection with the new choice.</li></ol>		
<b>Post condition:</b> <ul style="list-style-type: none"><li>• The selected time mode is saved by the system.</li></ul>		
<b>Exceptions:</b> <p>E1a. The jar file wasn't downloaded properly, and the device displays an error message.</p>		

<b>Use Case Name:</b> Start game	<b>ID:</b> UC-2	<b>Priority:</b> High
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes how a player can start the chess game.		
<b>Trigger:</b> Player clicks on the “Start Game” button.		
<b>Preconditions:</b> <ul style="list-style-type: none"> <li>Players have chosen the game mode they want to play in.</li> </ul>		
<b>Basic Path:</b> <ol style="list-style-type: none"> <li>The player clicks on the “Start Game” button.</li> <li>The system verifies that a game mode has been selected.</li> <li>The game starts with the pieces in the initial position.</li> <li>The timers get set up according to the game mode selected.</li> </ol>		
<b>Alternative path:</b> <p>A2a. There is no game mode selected.</p> <ol style="list-style-type: none"> <li>The system displays a message asking to select a game mode first.</li> </ol>		
<b>Post condition:</b> <ul style="list-style-type: none"> <li>The selected time mode is saved by the system.</li> </ul>		
<b>Exceptions:</b> <p>E3a. The game did not start properly, and the system displays an error message.</p>		

<b>Use Case Name:</b> Move Piece	<b>ID:</b> UC-3	<b>Priority:</b> High
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes what happens when a player moves a piece.		
<b>Trigger:</b> A player clicks and drags a piece of their color to a different tile.		
<b>Preconditions:</b> <ul style="list-style-type: none"> <li>Players have started the game and the board is working properly.</li> </ul>		

<ul style="list-style-type: none"> <li>The player has clicked on a piece of their own color.</li> </ul>
<b>Basic Path:</b> <ol style="list-style-type: none"> <li>1. Player clicks on one of their own pieces to move.</li> <li>2. The game highlights the legal tiles for that piece.</li> <li>3. The player drags the piece to a new legal tile.</li> <li>4. The game updates the new coordinates for that piece.</li> <li>5. The display updates the position of the piece.</li> <li>6. The turn is passed to the other player.</li> </ol>
<b>Alternative Path:</b> <p>A4a. The player has captured one of the opponent's pieces.</p> <ol style="list-style-type: none"> <li>1. The piece is removed from the board and saved as a captured piece.</li> </ol>
<b>Post condition:</b> <ul style="list-style-type: none"> <li>The piece has been moved to a new tile.</li> </ul>
<b>Exceptions:</b> <p>E1a. The player tries clicking on a piece that is not from their own color and the system prevents the action from happening.</p> <p>E3a. The player tries to move the piece to a tile with another piece of their color and the system prevents the move from happening.</p> <p>E3b. The player tries to move the piece an illegal tile and the system prevents the move from happening.</p>

<b>Use Case Name:</b> Time runs out	<b>ID:</b> UC-4	<b>Priority:</b> Medium
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes what happens when the time runs out for any of the players.		
<b>Trigger:</b> Timer reaches 0 seconds for any of the two players.		
<b>Preconditions:</b> <ul style="list-style-type: none"> <li>Players have started the game and the first move has been done.</li> </ul>		
<b>Basic Path:</b>		

1. The timer reaches 0 seconds for either of the two players
2. The system saves the player whose timer is not 0 as the winner.
3. The game ends and results are displayed.

**Alternative Path:**

No alternative paths identified.

**Post condition:**

- The game is ended, and the results are displayed.

**Exceptions:**

E1a. The timer does not work correctly.

1. The players might have to restart the game.

<b>Use Case Name:</b> Check Mate	<b>ID:</b> UC-5	<b>Priority:</b> High
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes what happens when a player check mates the opponent.		
<b>Trigger:</b> A player moves their piece and puts the other player's king in checkmate position.		
<b>Preconditions:</b> <ul style="list-style-type: none"><li>• Players have started the game and the board is working properly.</li><li>• The opponent's king has no safe moves to play</li></ul>		
<b>Basic Path:</b> <ol style="list-style-type: none"><li>1. Player makes a move that leaves the opponent's king unprotected.</li><li>2. The system saves the player whose king is in checkmate as the loser.</li><li>3. The game ends and results are displayed.</li></ol>		
<b>Alternative Path:</b> <p>No alternative paths identified.</p>		
<b>Post condition:</b> <ul style="list-style-type: none"><li>• The game is ended, and results are displayed.</li></ul>		

**Exceptions:**

E1a. If the player who places the other player's King in checkmate is not aware of it, the system will inform them and prompt them to make another move.

<b>Use Case Name:</b> Draw game	<b>ID:</b> UC-6	<b>Priority:</b> High
<b>Actor:</b> Player 1 and Player 2		
<b>Description:</b> The use case describes what happens when both players agree to draw.		
<b>Trigger:</b> Both players click on the draw button.		
<b>Preconditions:</b> <ul style="list-style-type: none"><li>• Players have started the game and the board is working properly.</li></ul>		
<b>Basic Path:</b> <ol style="list-style-type: none"><li>1. Both players click on their draw button.</li><li>2. The system saves that both players agreed to draw.</li><li>3. Game ends and results are displayed.</li></ol>		
<b>Alternative Path:</b> <p>A4a. One of the players didn't press the button.</p> <ol style="list-style-type: none"><li>1. The player makes another move, and the game continues.</li></ol>		
<b>Post condition:</b> <ul style="list-style-type: none"><li>• The game ends in a draw and results are displayed.</li></ul>		
<b>Exceptions:</b> <p>E1a. The system doesn't take input from one or either of the buttons.</p> <ol style="list-style-type: none"><li>1. Players must press the button again.</li></ol>		

<b>Use Case Name:</b> End game	<b>ID:</b> UC-7	<b>Priority:</b> High
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes what happens when the game comes to an end.		



<b>Trigger:</b> An event has occurred that leads the game to end.
<b>Preconditions:</b> <ul style="list-style-type: none"> <li>• The game has already started.</li> <li>• Either UC-4, UC-5 or UC-6 has occurred.</li> </ul>
<b>Basic Path:</b> <ol style="list-style-type: none"> <li>1. The game has come to an end</li> <li>2. The results of the game are displayed on the screen, including the winner or if it was a draw.</li> <li>3. Two buttons appear on the screen, one with the function of restarting the game and another with the function of going back to the main menu.</li> </ol>
<b>Alternative Path:</b> <p>A3a. The player clicks on the exit button.</p> <ol style="list-style-type: none"> <li>1. The java process ends and the frame closes.</li> </ol>
<b>Post condition:</b> <ul style="list-style-type: none"> <li>• The game has ended, and the result has been displayed.</li> <li>• The player has the option of restarting the game or going to the main menu.</li> </ul>
<b>Exceptions:</b> <p>E3a. There is an error with one of the buttons.</p> <ol style="list-style-type: none"> <li>1. The player may need to exit the application.</li> </ol>

<b>Use Case Name:</b> Display the results	<b>ID:</b> UC-8	<b>Priority:</b> High
<b>Actor:</b> Player 1 or Player 2		
<b>Description:</b> The use case describes what happens when the game comes to an end and the results are being displayed.		
<b>Trigger:</b> The game has ended.		
<b>Preconditions:</b> <ul style="list-style-type: none"> <li>• The game has ended.</li> <li>• The game board and pieces are still visible on the screen.</li> </ul>		

**Basic Path:**

1. The results of the game are displayed on the screen, including the winner or if it was a draw. Information on the result based on the rules of chess is also displayed. The number of pieces each player has captured and the status of the king.
3. Two buttons appear on the screen, one with the function of restarting the game and another with the function of going back to the main menu.

**Alternative Path:**

A3a. The player clicks on the exit button.

1. The java process ends and the frame closes.

**Post condition:**

- The results of the game have been displayed on the screen.
- The player has the option of restarting the game or going to the main menu.

**Exceptions:**

E3a. There is an error with one of the buttons.

1. The player may need to exit the application.

Activity diagram on the use case of “Move Piece”.

