**Team:** Eric Fossas
Ryan O'Connell
Chase Springer
Kevin Rau
Ryan Riley

**Title:** Chess Game

**Project Summary:** A simple chess game. Each piece will be an object with the appropriate attributes and methods. The players should be able to play a chess game using the objects which will interact with each other.

# Project Requirements

**Business Requirements:**
None needed for a chess game.

**User Requirements:**

| ID | Requirement | Topic Area | User | Priority |
|---|---|---|---|---|
| UR-01 | As a player I need to be able to interact with the main menu. | Main Menu | Player | Moderate |
| UR-02 | As a player I need to be able to move chess pieces. | Gameplay | Player | High |
| UR-03 | As a player I need to be able to rage quit | Gameplay | Player | Moderate |
| UR-04 | As a player I need to be able to select a piece. | Gameplay | Player | High |
| UR-05 | As a players I need to be able to make a user name. | Profiles | Player | Moderate |
| UR-06 | As a player I need to be able to log in to my profile. | Profiles | Player | Moderate |
| UR-07 | As a player I need to be able to see the leaderboard. | Profiles | Player | Moderate |
| UR-08 | As a player I need to be able to log out of my profile. | Profiles | Player | Moderate |
| UR-09 | As a player I need to be able to restart the game. | Gameplay | Player | High |
| UR-10 | As a player I need to be able to start a new game | Main menu | Player | High |

**Functional Requirements:**

| ID | Requirement | Topic Area | User | Priority |
|---|---|---|---|---|
| FR-01 | Game needs to keep track of who's turn it is. | Gameplay | Player | High |
| FR-02 | Game needs to respond to win conditions. | Gameplay | Player | High |
| FR-03 | Game needs to respond to stalemate conditions. | Gameplay | Player | High |
| FR-04 | Game needs to track and display eliminated pieces. | Gameplay | Player | High |
| FR-05 | Game needs to remember where each piece is located on the game board. | Gameplay | Player | High |
| FR-06 | Game needs to display who's turn it is. | Gameplay | Player | High |
| FR-07 | Game needs to replace a pawn with another killed piece when reaching the other side of the board. | Gameplay | System | High |

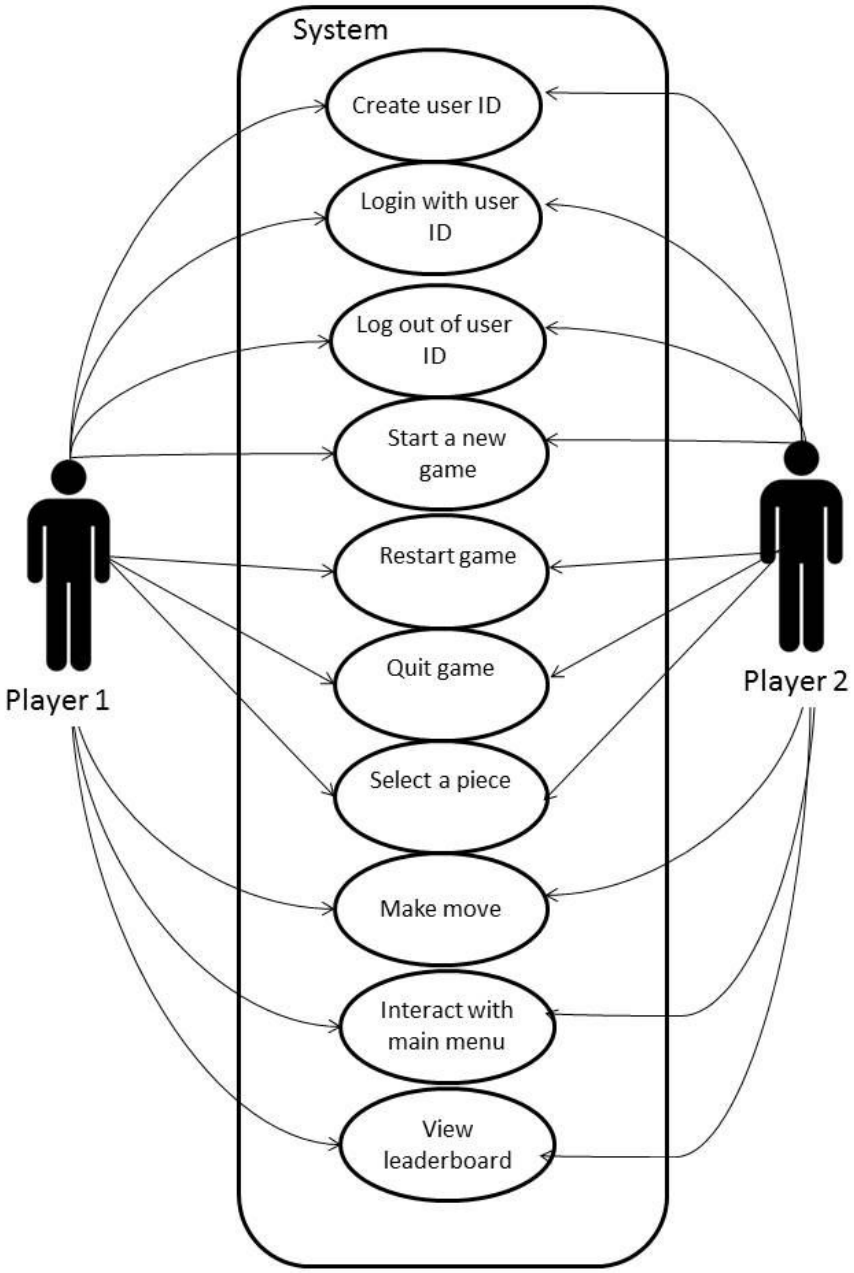
**Non-Functional Requirements:**

| ID | Requirement | Topic Area | User | Priority |
|---|---|---|---|---|
| NFR-01 | Game needs to have an uptime of 99.9% | Gameplay | System | High |
| NFR-02 | Game needs to respond in less than 2 seconds. | Performance | System | High |
| NFR-03 | Full application should not require user to back out and read a help document. | Game | System | Moderate |
| NFR-04 | A player's selected move should appear on the board in under 5 seconds. | Performance | System | Moderate |

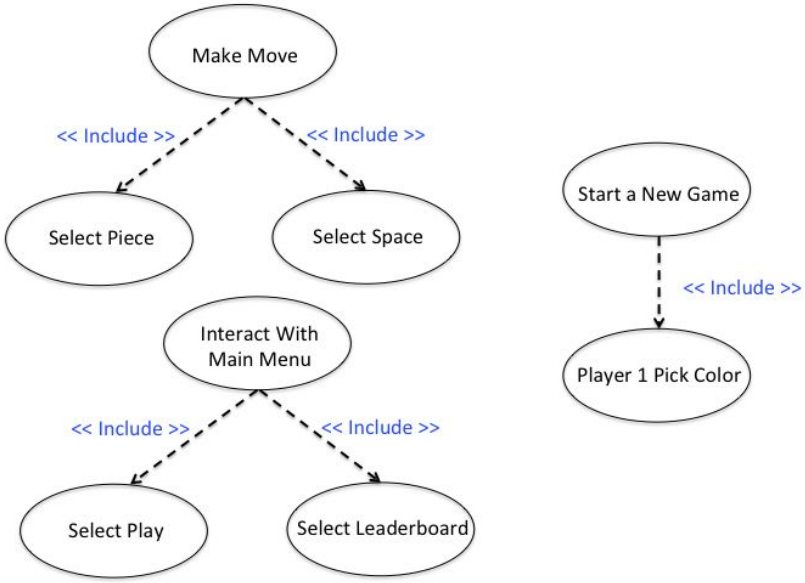
# Use Cases

**Actors**: Player 1, Player 2

**Use Case Overview:**



**Sub-diagrams:**



**Use Case Documents:**

| | |
|---|---|
| Use Case ID: | UC-01 |
| Use Case Name: | Interact with main menu |
| Description: | Users should be able to view and interact with the main menu |

| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | App has been loaded |
| Post-Conditions: | Player is sent to the desired place based on which button was clicked in the main menu |
| Frequency of Use: | Every time the app is loaded |
| Flow of Events: | **Actor Action** / **System Response**<br>User is in the main menu<br>User clicks on desired button — User is prompted by the main menu depending on which button was selected. |
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-02 |
| Use Case Name: | Move a chess piece. |
| Description: | Selected piece is able to move in scope of move set and available moves on board. |

| | |
|---|---|
| Actors | Player1, Player2 |
| Pre-Conditions : | Chess piece is selected in current position. |
| Post-Conditions: | Chess piece location is updated with updated available moves. |
| Frequency of Use: | Every turn in game. |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>User has already selected a chess piece.</td><td></td></tr><tr><td>User moves piece to desired spot on board.</td><td>Piece is updated on the board.</td></tr></table> |
| Variations: | No moves are available. |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-03 |
| Use Case Name: | Quit a game |
| Description: | User wants to end the current game and return to the main menu |

| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | A game is in progress |
| Post-Conditions: | The players return to the main menu |
| Frequency of Use: | Every time a user wishes to end the current game |
| Flow of Events: | |

| Actor Action | System Response |
|---|---|
| User clicks menu button | Menu options are displayed |
| User clicks on quit option | “Are you sure?” displayed |
| User selects yes | Game returns to main menu |

| | |
|---|---|
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-04 |
| Use Case Name: | Select a chess piece. |
| Description: | User selects a chess piece and is given options of available move sets and move limitations with respect to the board. |

| | |
|---|---|
| Actors | Player1, Player2 |
| Pre-Conditions : | Chess piece are unselected. |
| Post-Conditions: | Chess piece is selected with available moves shown. |
| Frequency of Use: | Every turn in game. |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>User is in a game</td><td></td></tr><tr><td>User clicks on a piece</td><td>Piece becomes highlighted as well as valid move destinations</td></tr></table> |
| Variations: | N/A (would be no pieces, therefore game has been completed) |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-05 |
| Use Case Name: | User profile creation |
| Description: | A user will create a new profile that can be used to log into games |

| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | User has not logged into a profile |
| Post-Conditions: | User will be logged in under a new profile |
| Frequency of Use: | Every time a user wishes to create a new profile or there is a new user |
| Flow of Events: | |

| Actor Action | System Response |
|---|---|
| User is in menu and not currently logged into a profile | |
| User clicks on “create profile” button | User is prompted to input a name |
| User inputs name and clicks ok button | User is returned to main menu and signed in under new name |

| | |
|---|---|
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-06 |
| Use Case Name: | User logon |
| Description: | User will logon into a valid profile to enter the game menu. |

| | |
|---|---|
| Actors | Player 1, Player2 |
| Pre-Conditions : | User is not logged into a valid profile |
| Post-Conditions: | User logs in and UI is updated with the main menu interface. |
| Frequency of Use: | Every time a user wants to play the game. |
| Flow of Events: | |

| Actor Action | System Response |
|---|---|
| User is in menu and not currently logged into a profile | |
| User selects a profile. | User is signed into selected profile |

| | |
|---|---|
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-07 |
| Use Case Name: | Check the leaderboard |
| Description: | A user wants to see player statistics |

| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | Player is logged into the game and a game is not in progress |
| Post-Conditions: | Player is returned to main menu |
| Frequency of Use: | Every time a user wants to see user statistics |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>User is in main menu</td><td></td></tr><tr><td>User clicks “display leaderboard” button</td><td>User is taken to a board displaying all users and stats</td></tr></table> |
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-08 |
| Use Case Name: | Save Game |
| Description: | User will load their last saved game. |

| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | User is logged in. A game has been started. |
| Post-Conditions: | User is must log back in. |
| Frequency of Use: | Can happen at any time during a game. |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>User is logged in.</td><td></td></tr><tr><td>User clicks save button.</td><td>Get the current board array</td></tr><tr><td></td><td>Save the board array to the user's save file.</td></tr></table> |
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

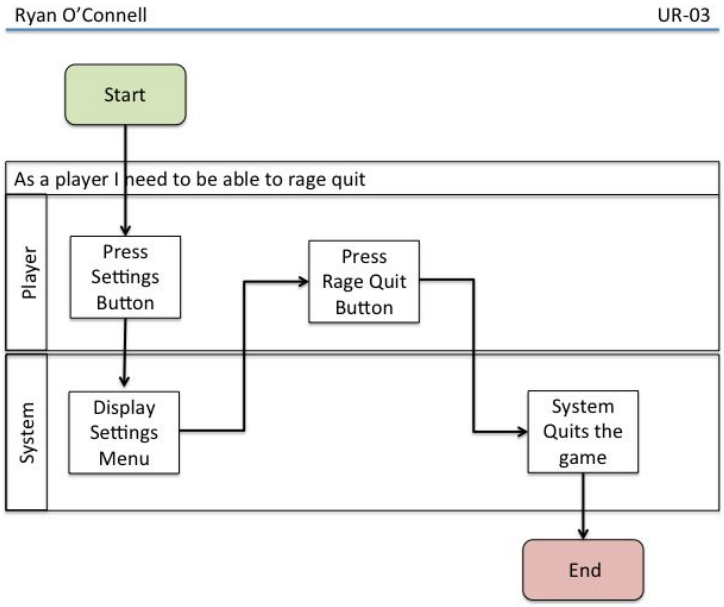
| | |
|---|---|
| Use Case ID: | UC-09 |
| Use Case Name: | Restart a game |
| Description: | User wants to start the current game over |

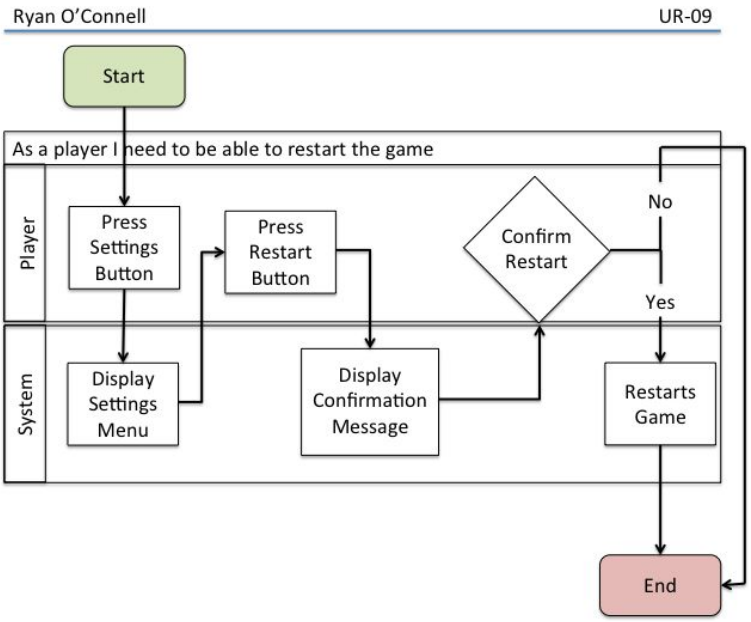
| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | A game is in progress |
| Post-Conditions: | A new game is started with the same players |
| Frequency of Use: | Every time a user wants to restart the game |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>User is in the middle of a game</td><td></td></tr><tr><td>User clicks restart game button</td><td>User is prompted “are you sure?”</td></tr><tr><td>User clicks “yes”</td><td>A new game is created with same users logged in</td></tr></table> |
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

| | |
|---|---|
| Use Case ID: | UC-10 |
| Use Case Name: | Start a new game |
| Description: | Players will be able start a new chess game |

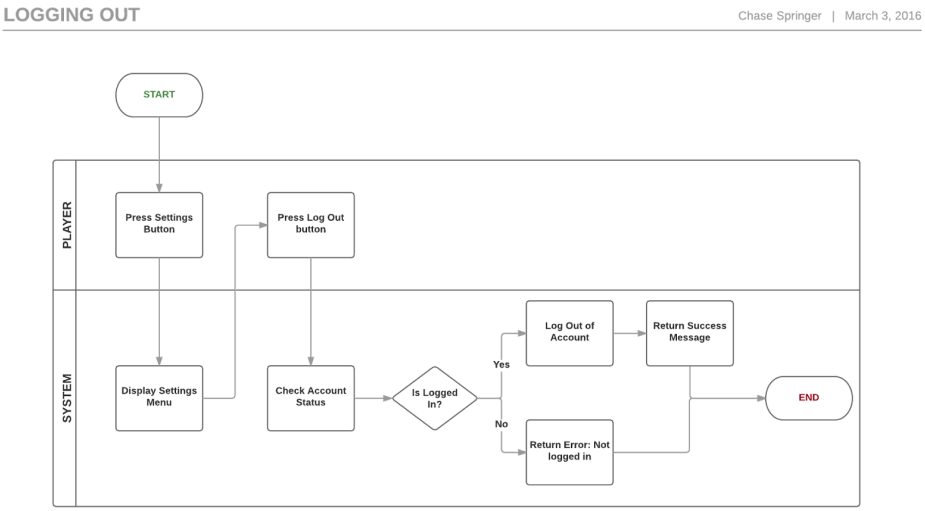
| | |
|---|---|
| Actors | Player 1 or Player 2 |
| Pre-Conditions : | A game is not currently in progress |
| Post-Conditions: | A game will be set up |
| Frequency of Use: | Every time a user wishes to start a game |
| Flow of Events: | <table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>Two users are logged into profiles</td><td></td></tr><tr><td>User selects start new game</td><td>A new board is set up with the two users as players</td></tr></table> |
| Variations: | N/A |
| Notes and Issues: | N/A |
| Developer Notes: | N/A |

# Activity Diagrams:

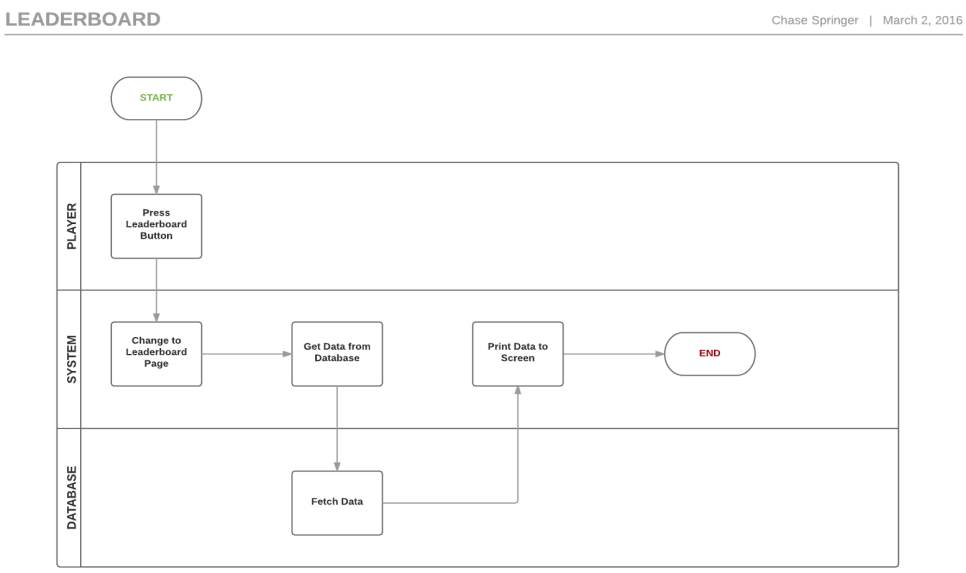


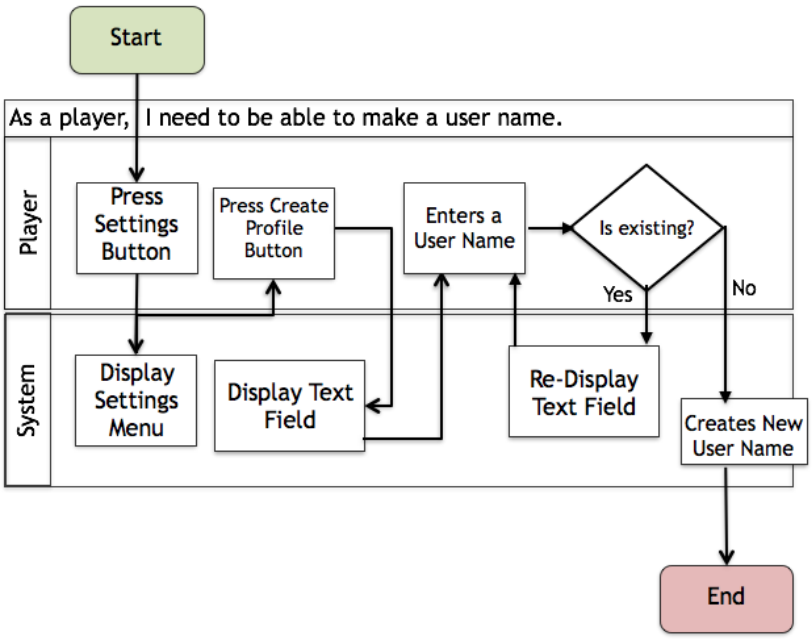
**UR-08** Chase Springer



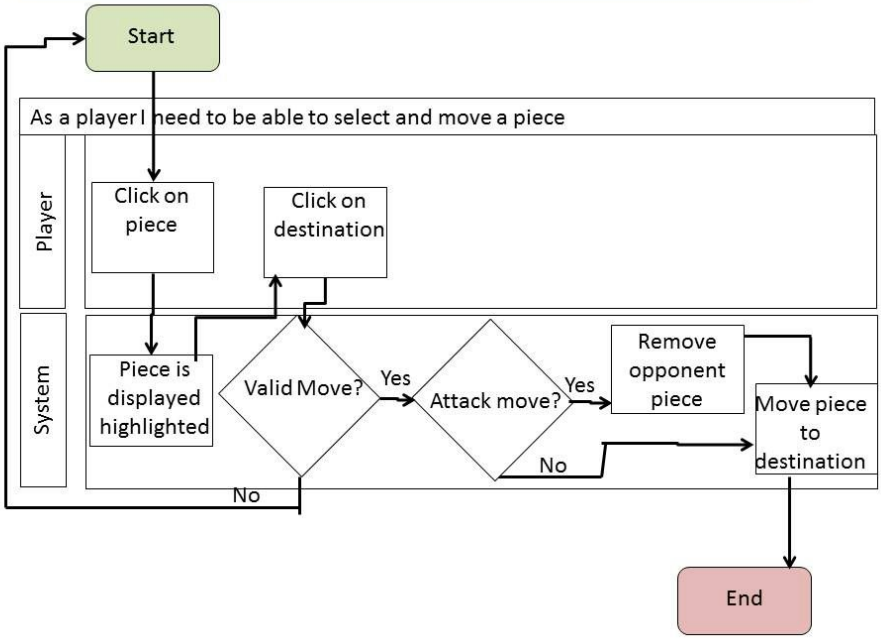

**UR-07**



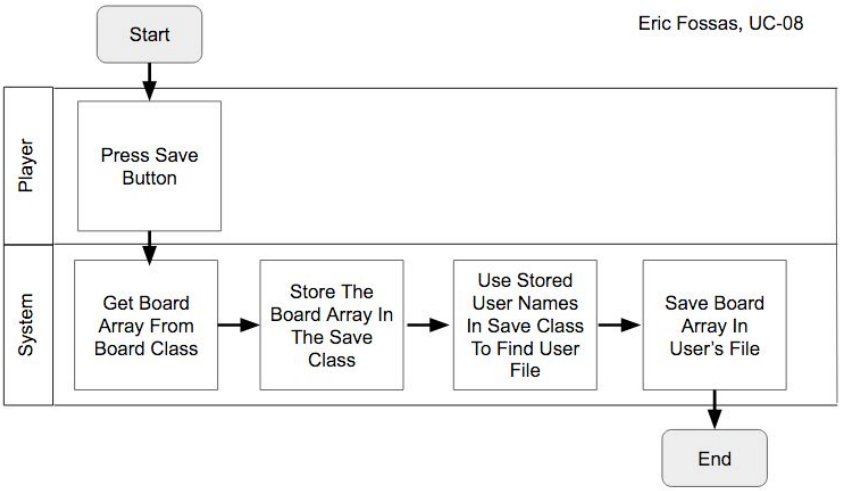
Kevin Rau UC-05




Ryan Riley UR-04, UR-02



As a player, I need to save data

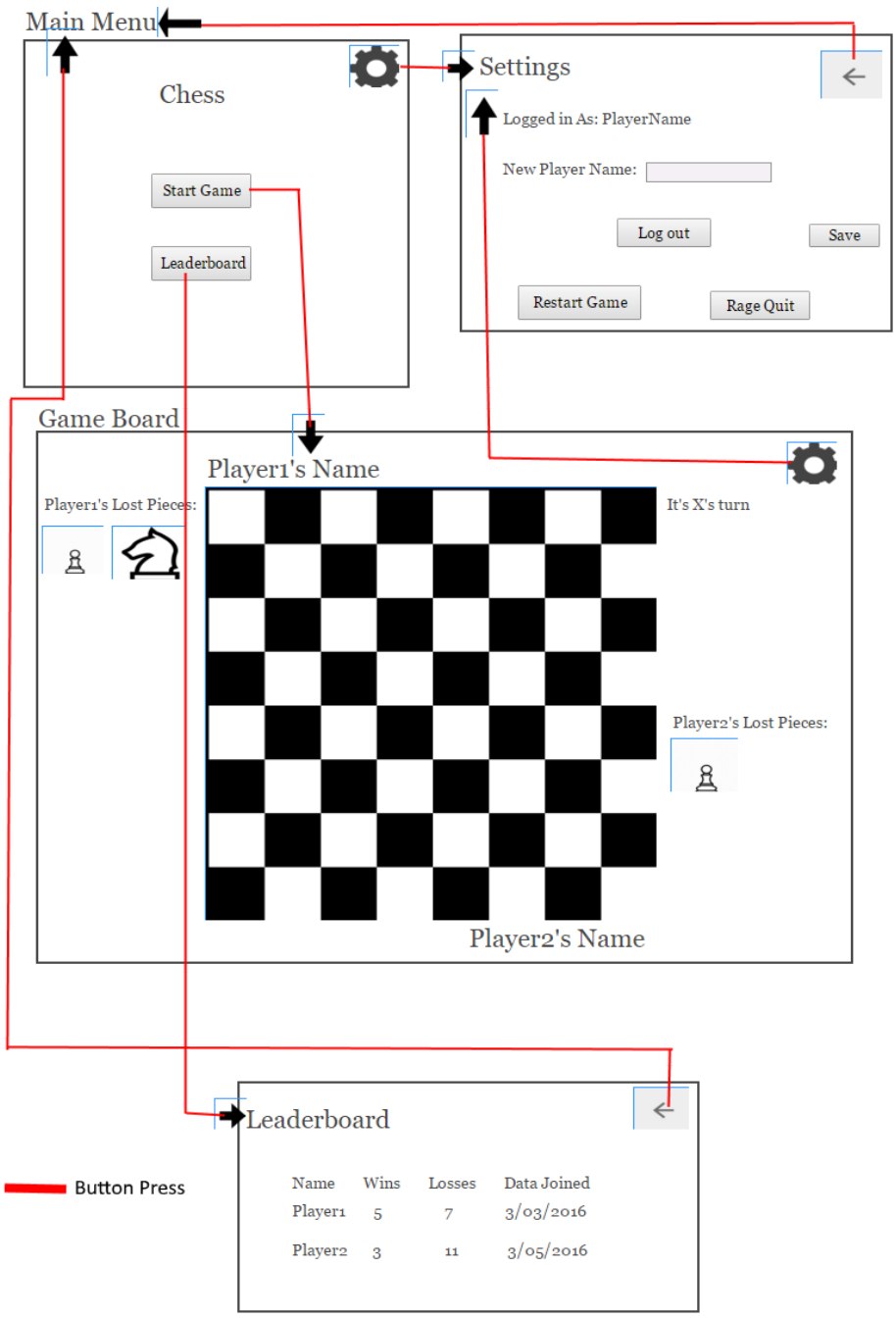



# Data Storage:

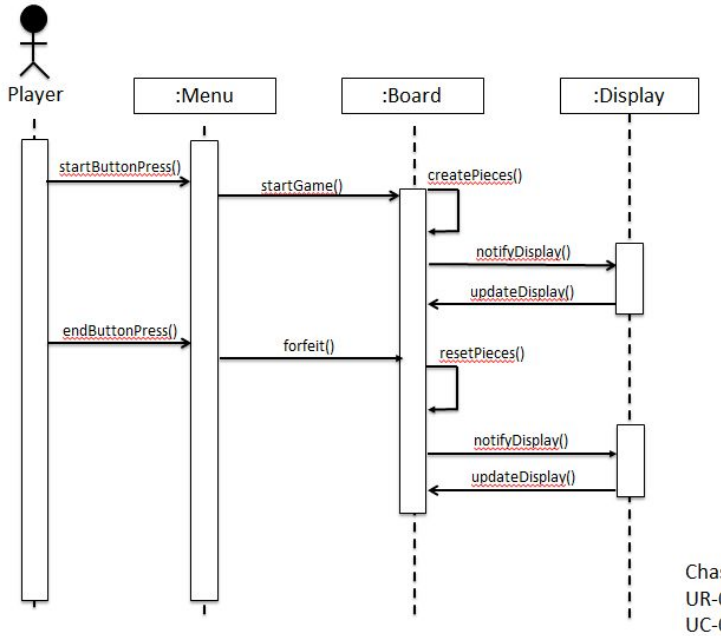
**Data Storage:** JSON files on a server.
**Classes:**

- The following data is stored: usernames, record, current board, current opponent.
- Save class saves the data.
- Stats class reads the data.

# UI Mockup:

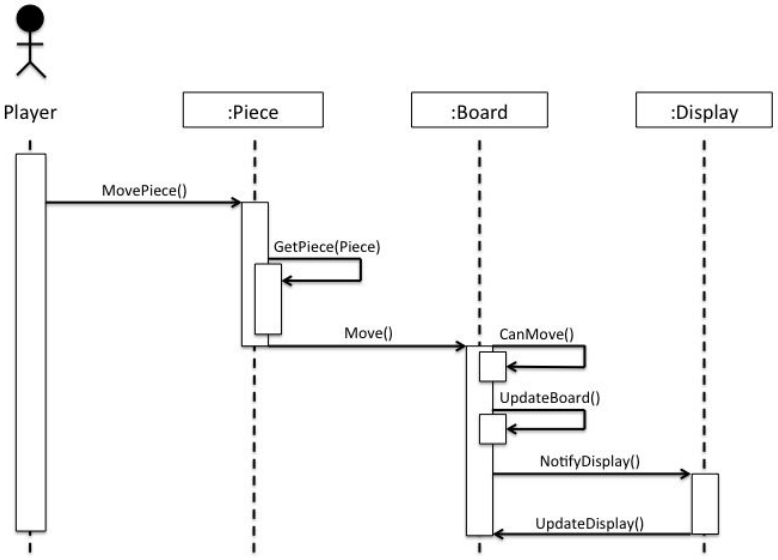

Main Menu
Chess
Start Game
Leaderboard
Settings
Logged in As: PlayerName
New Player Name:
Log out
Save
Restart Game
Rage Quit
Game Board
Player1's Name
Player1's Lost Pieces:
It's X's turn
Player2's Lost Pieces:
Player2's Name
Leaderboard
Name Wins Losses Data Joined
Player1 5 7 3/03/2016
Player2 3 11 3/05/2016
Button Press

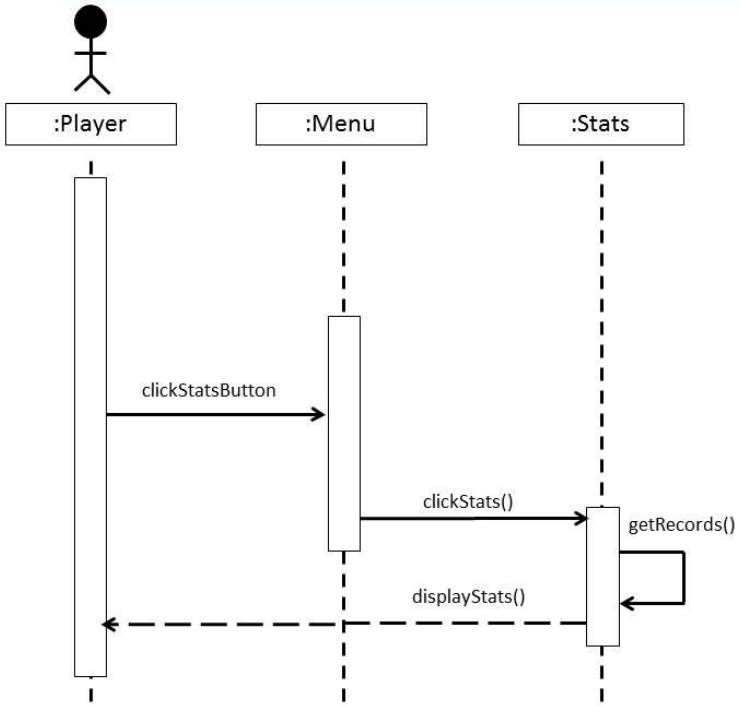
# User Interactions:



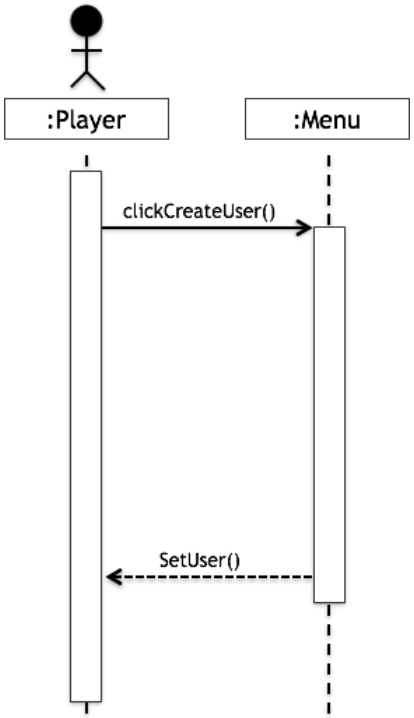

Chase Springer
UR-03, UR-10
UC-04

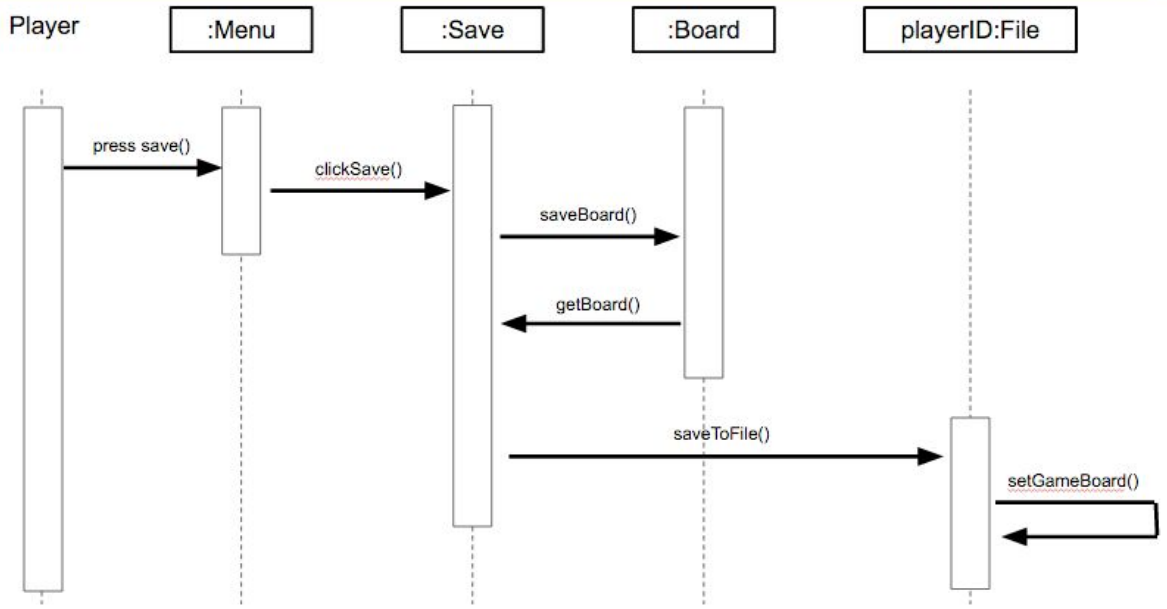
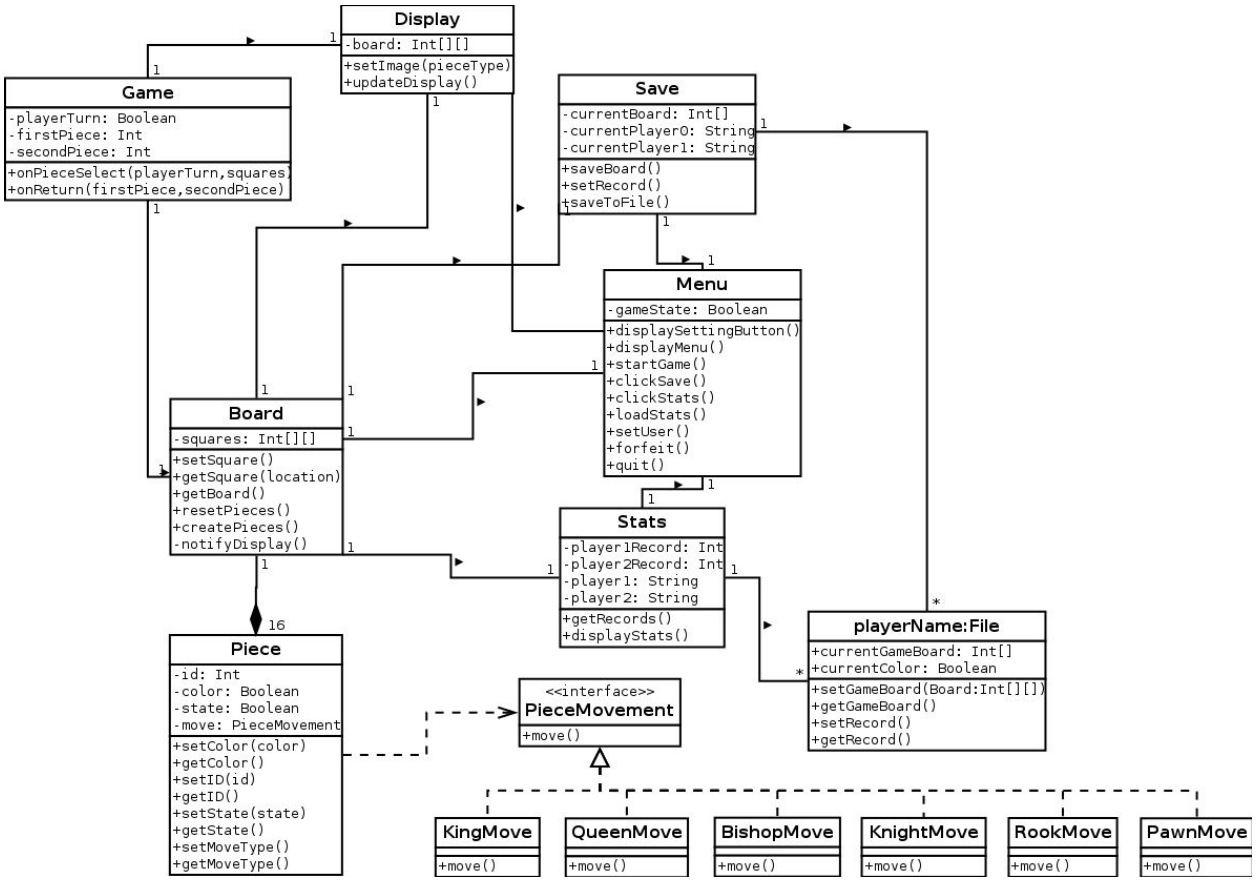
Ryan O'Connell UR-02, UR-04, UC-02, UC-03






Kevin Rau - UR-05




Player
:Menu
:Save
:Board
playerID:File
press save()
clickSave()
saveBoard()
getBoard()
saveToFile()
setGameBoard()

# Class Diagrams: