Mobile Chess Sprint Deliverable

January 26, 2015

**Prepared For:**

Daniel Freeman

Todd Breedlove

**Prepared By:**

Stewart Taylor

Jacob Neal

Aaron Costner

Lewis Sanchez

Table of Contents

[Introduction 2](#_Toc410075991)

[Backlog Items for Sprint 2](#_Toc410075992)

[Traceability Matrix 3](#_Toc410075993)

[Use Cases 4](#_Toc410075994)

[General Information 4](#_Toc410075995)

[Requirements/Feature Trace 4](#_Toc410075996)

[Post-Conditions 5](#_Toc410075997)

[General Information 7](#_Toc410075998)

[Requirements/Feature Trace 7](#_Toc410075999)

[Post-Conditions 8](#_Toc410076000)

[General Information 9](#_Toc410076001)

[Requirements/Feature Trace 9](#_Toc410076002)

[Post-Conditions 10](#_Toc410076003)

[Sequence Diagrams 11](#_Toc410076004)

[User selects chess game pool 11](#_Toc410076005)

[User transmits move to other user 12](#_Toc410076006)

[User connects to dedicated server 13](#_Toc410076007)

[Class Diagrams 14](#_Toc410076008)

[CG\_Server 14](#_Toc410076009)

[CG\_PlayerThread 15](#_Toc410076010)

[CG\_Match 16](#_Toc410076011)

[Screenshots 17](#_Toc410076012)

[Server 17](#_Toc410076013)

[Lobby 18](#_Toc410076014)

[QML Scalability 19](#_Toc410076015)

[Conclusion 20](#_Toc410076016)

# **Introduction**

This deliverable lists all design documents, code and screenshots associated with this sprint. The table of contents lists the order in which these documents were produced. For example -- our group went from the requirements, to the use cases, to the sequence diagrams, to the class diagrams and finally to the code itself.

# Backlog Items for Sprint

There were two backlog items that we worked on this sprint, the game lobby and the server. We set up a foundation for the game pools for either a 1 minute, 5 minute, or 30 minute game and also converted our presentation layer to QML for the login and lobby.

# Traceability Matrix

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **UC001** | **UC002** | **UC003** | **UC004** | **UC005** | **UC006** | **UC007** | **UC008** | **UC009** | **UC010** | **UC011** |
| **1ai** | X |  |  |  |  |  |  |  |  |  |  |
| **1aii** | X |  |  |  |  |  |  |  |  |  |  |
| **1aiii** | X |  |  |  |  |  |  |  |  |  |  |
| **1b** | X |  |  |  |  |  |  |  |  |  |  |
| **1ci** | X |  |  |  |  |  |  |  |  |  |  |
| **2ai** |  |  |  |  |  |  |  |  |  |  |  |
| **2aii** |  |  |  |  |  |  |  |  |  |  |  |
| **3a** |  |  |  |  |  |  |  |  |  |  |  |
| **3b** |  |  |  |  |  |  |  |  |  |  |  |
| **4a** |  |  |  |  |  |  |  |  |  |  |  |
| **4b** |  |  |  |  |  |  |  |  |  |  |  |
| **4c** |  |  |  |  |  |  |  |  |  |  |  |
| **4d** |  |  |  |  |  |  |  |  |  |  |  |
| **5ai** |  | X |  |  |  |  |  |  |  |  |  |
| **5aii** |  |  |  |  |  |  |  |  |  |  |  |
| **6ai** |  |  |  |  |  |  |  |  |  |  |  |
| **6bi** |  |  |  |  |  |  |  |  |  |  |  |
| **7a** |  |  |  |  |  |  |  |  |  |  |  |
| **8** |  |  |  |  |  |  |  |  |  |  |  |
| **9** |  |  |  |  |  |  |  |  |  |  |  |
| **10** |  |  |  |  |  |  |  |  |  |  |  |
| **11** |  |  |  |  |  |  |  |  |  |  |  |
| **12** |  |  | X | X |  |  |  |  |  |  |  |
| **12a** |  |  |  |  |  |  |  |  |  |  |  |

# Use Cases

Below is our only use case for our sprint. The outline follows the same format as the use case outline on the SharePoint site. Some parts of SharePoint outline were omitted because they are not applicable to our project (i.e. Timing execution).

|  |  |
| --- | --- |
| General Information | |
| Use Case ID Number : UC002  Subject Area : Navigating Application Lobby  Description : Selecting a queue for a timed game of chess. | Responsible Analyst : Todd Breedlove |

|  |  |
| --- | --- |
| Requirements/Feature Trace | |
| REQ# | Requirements Name and / or Short Description |
| 5ai | Queues divided into 1 minute, 5 minute, and 30 minute games |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Actors** | | |
| **Actor Name** | **P/S** | **Brief Description** |
| User | P | Someone who wishes to play a 1 minute, 5 minute, or 30 minute game of chess. |
| Server | S | Places a user in a pool and allocates users into a game of chess based on their Elo rating. |
|  |  |  |

|  |  |
| --- | --- |
| **Pre-Conditions** | |
| # | Description |
| 1 | Needs to be logged into the application. |

|  |
| --- |
| **Start Stimulus** |
| Successfully logging in with trigger this use case. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Main Course Steps** | | | |
| **Number** | **Description** | **Adds/Alt UC Name/Number** | **Bus Rule(s)#** |
| 01 | User decides how long of a chess game to play. |  |  |
| 02 | Server places User in a queue for that kind of chess game. |  |  |
| 03 | User will be idle until Server successfully find another User in the queue within 200 points of the User’s Elo rating, or if the user decides to exit early. |  |  |
| 04 | If User is put into a match with another User, they will play a game of chess with that time control. Otherwise if the User left early, User is brought back to the lobby. |  |  |
| 05 | Server randomly decides which User goes first if a match is initiated. |  |  |

|  |  |  |
| --- | --- | --- |
| **Exception Conditions** | | |
| **Exception Situations** | **Action(s) on Exception** | **Adds/Alt Use Case #** |
| User loses connection during idle time | User is removed from queue, logged out, and brought back to log in screen. |  |
| User loses connection in the lobby | User is logged out and returned to log in screen. |  |

|  |  |
| --- | --- |
| Post-Conditions | |
| **#** | **Description** |
| 1 | If successfully put in a match, Users will be put together in a match of chess with the selected time control. |
| 2 | If the User exits early, that User will be returned to the lobby screen. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Assumptions** | | | | | |
| **#** | **Assumption** | **Date**  **Raised** | **Raised**  **By** | **Date**  **Verified** | **Verified By** |
| 1 | That users can speak and read English | 12/13/14 | Aaron Costner |  |  |
| 2 | That there will be more than 1 player in each time control pool/queue | 12/13/14 | Stewart Taylor |  |  |

|  |  |
| --- | --- |
| General Information | |
| Use Case ID Number : UC003  Subject Area : Playing Chess  Description : Sending chess moves via text | Responsible Analyst : Todd Breedlove |

|  |  |
| --- | --- |
| Requirements/Feature Trace | |
| REQ# | Requirements Name and / or Short Description |
| 12 | Allow user to play chess against another player on application. |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Actors** | | |
| **Actor Name** | **P/S** | **Brief Description** |
| White Player | P | This User sends messages as the white player. |
| Black Player | P | This User sends messages as the black player. |
| Server | S | Server gets the information from a User who makes a move and sends it to the other User. |

|  |  |
| --- | --- |
| **Pre-Conditions** | |
| # | Description |
| 1 | Both Users need to have been successfully put into a match with one another for a selected time control. |

|  |
| --- |
| **Start Stimulus** |
| That both Users have been put into a match with one another. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Main Course Steps** | | | |
| **Number** | **Description** | **Adds/Alt UC Name/Number** | **Bus Rule(s)#** |
| 01 | Users are notified of their color, who they are matched with, and that a match is being set up. |  |  |
| 02 | White Player types in a move and sends it to the Server. |  |  |
| 03 | Server will transmit White Player’s move and send it to the Black Player. |  |  |
| 04 | Black Player types in a move and sends it to the Server. |  |  |
| 05 | Server will transmit Black Player’s move and send it to the White Player. |  |  |
| 06 | Steps 02 through 05 will repeat until a Player leaves the match, or an end state is arrived at. |  |  |

|  |  |  |
| --- | --- | --- |
| **Exception Conditions** | | |
| **Exception Situations** | **Action(s) on Exception** | **Adds/Alt Use Case #** |
| Player enters an illegal move | User is notified with an error sound and the move transmitted will not be made. |  |
| Player loses connection during the game | The Player who disconnected will forfeit the match, the Server will update the database, and the connected Player will return to the lobby, whereas the disconnected Player will be returned to the log in screen. |  |

|  |  |
| --- | --- |
| Post-Conditions | |
| **#** | **Description** |
| 1 | An end state is reached, and that state is recorded for each Player from the Server to the database. |
| 2 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Assumptions** | | | | | |
| **#** | **Assumption** | **Date**  **Raised** | **Raised**  **By** | **Date**  **Verified** | **Verified By** |
| 1 | That a player knows how to play chess | 12/13/14 | Stewart Taylor |  |  |
| 2 |  |  |  |  |  |

|  |  |
| --- | --- |
| General Information | |
| Use Case ID Number : UC004  Subject Area : Connecting to Server  Description : Players connecting to a server to play chess | Responsible Analyst : Todd Breedlove |

|  |  |
| --- | --- |
| Requirements/Feature Trace | |
| REQ# | Requirements Name and / or Short Description |
| 12 | Allow user to play chess against another player on application. |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Actors** | | |
| **Actor Name** | **P/S** | **Brief Description** |
| User | P | User connects to Raspberry Pi. |
| Raspberry Pi | S | Runs on port 2500 and handles incoming connections. |

|  |  |
| --- | --- |
| **Pre-Conditions** | |
| # | Description |
| 1 | User must have entered a valid username and password. |

|  |
| --- |
| **Start Stimulus** |
| User clicks log in button |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Main Course Steps** | | | |
| **Number** | **Description** | **Adds/Alt UC Name/Number** | **Bus Rule(s)#** |
| 01 | Server silently listens in background for incoming connections. |  |  |
| 02 | User sends connection request to server |  |  |
| 03 | Server creates new thread to compensate for new request. |  |  |
| 04 | Server sends acknowledge that user is connected to server. |  |  |

|  |  |  |
| --- | --- | --- |
| **Exception Conditions** | | |
| **Exception Situations** | **Action(s) on Exception** | **Adds/Alt Use Case #** |
| Server times out | User notified that server timed out and prompted to retry connecting. |  |
| Maximum number of pending connections exceeded | Server notifies user(s) that server is full and prompted to retry at another time. |  |
|  |  |  |

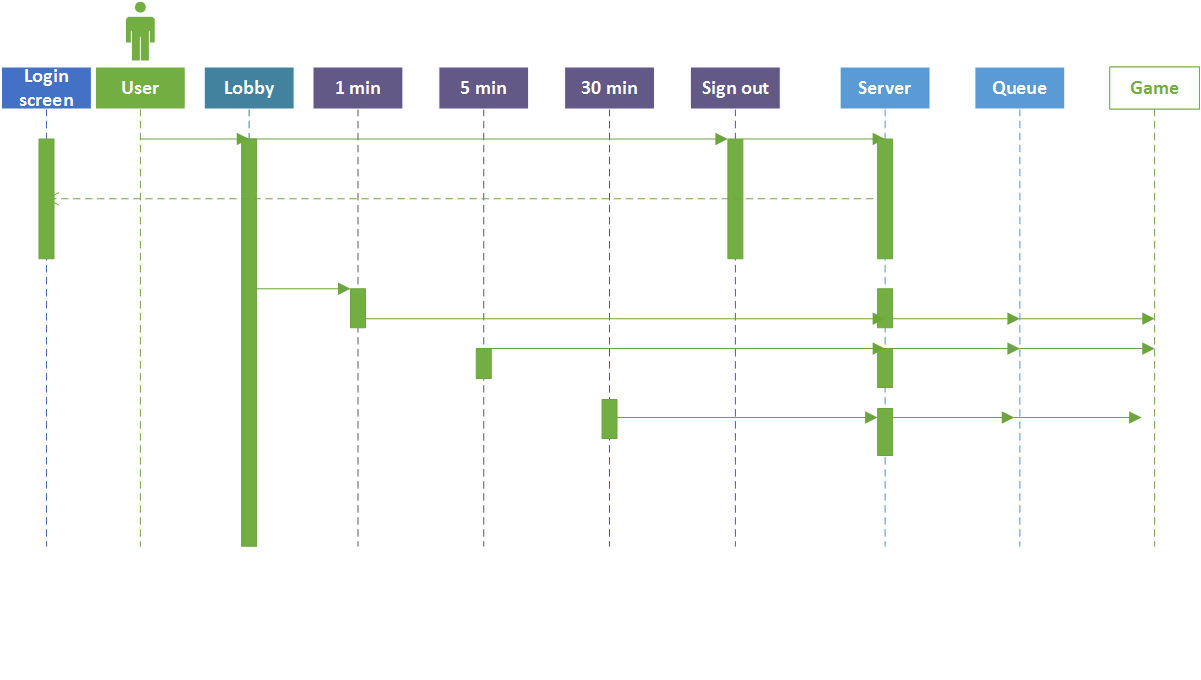
|  |  |
| --- | --- |
| Post-Conditions | |
| **#** | **Description** |
| 1 | A message stating the user is connected to the Chessgames Mobile Server |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Assumptions** | | | | | |
| **#** | **Assumption** | **Date**  **Raised** | **Raised**  **By** | **Date**  **Verified** | **Verified By** |
| 1 | Raspberry Pi is running 24/7. | 12/13/14 | Aaron Costner |  |  |
| 2 |  |  |  |  |  |

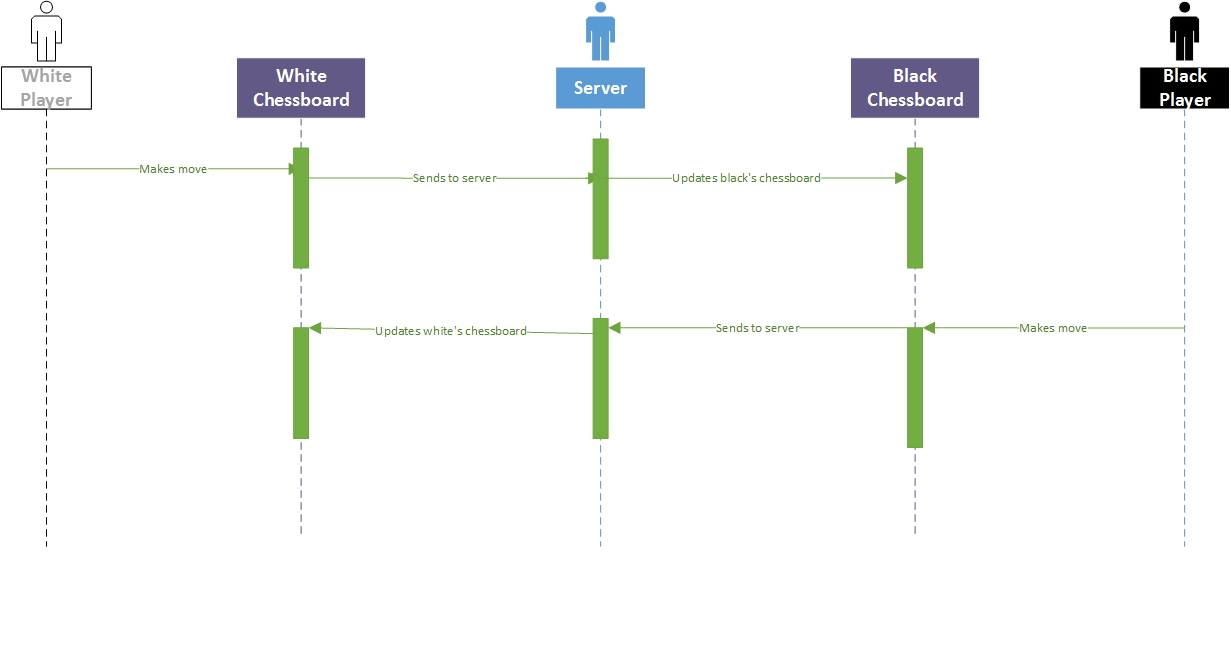
# Sequence Diagrams

Below are three sequence diagrams.

## User selects chess game pool



## User transmits move to other user

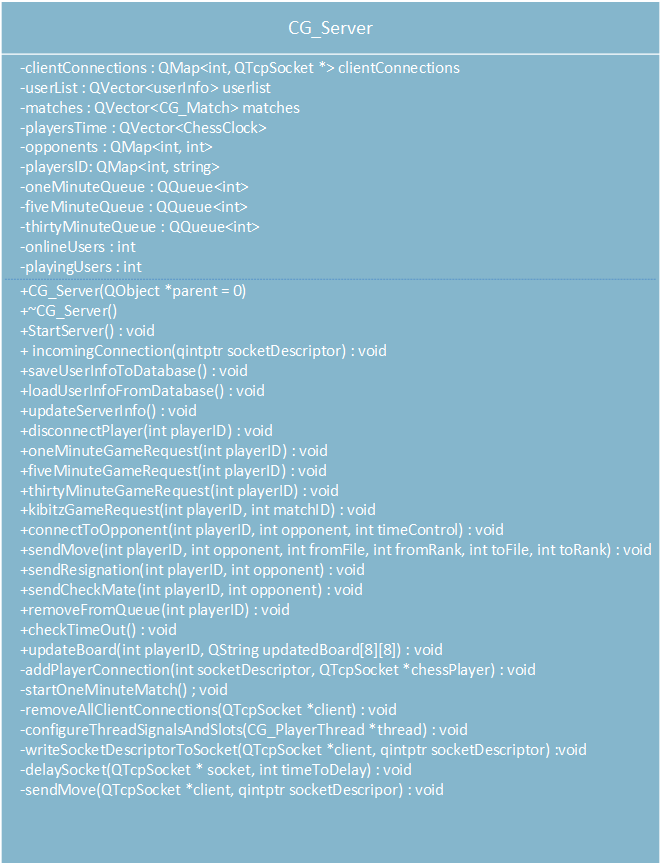


## User connects to dedicated server

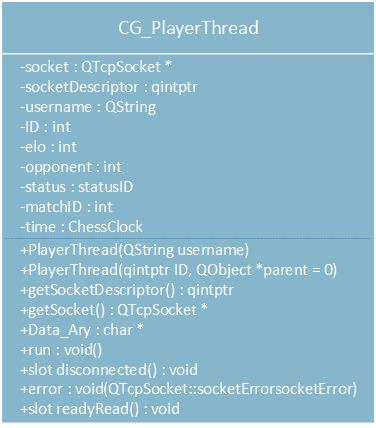


# Class Diagrams

## CG\_Server



## CG\_PlayerThread



## CG\_Match



# Screenshots

Below are screenshots of the mobile chess application running, as well as our server.

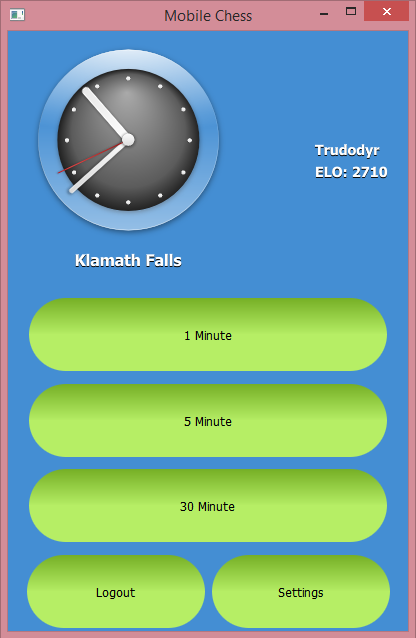
## Server

This is a multithreaded server that handles incoming connection requests from clients.



## Lobby

This is our game lobby where our users can select either a 1 minute, 5 minute, or 30 minute game pool.



## QML Scalability

This demonstrates our control and label scalability in our app after making the change to QML for our presentation layer.



# Conclusion

From the design documents to our PowerPoint presentation, this sprint can be considered a success. We finished the sprint ahead of schedule, went above/beyond the requirements and even received positive feedback from our product owner and stakeholder. Our goal is to continue to maintain this momentum.