# Software Requirements Specification

# *Chessmasters*

# v1.1

*Prepared by:*

*Joshua Kemmerer*

*Shishir Kharel*

*Tuan Nguyen*

# Revision History

| Name | Date | Reason for change | Version |
|---|---|---|---|
| Joshua Kemmerer,<br>Tuan Nguyen,<br>Shishir Kharel | January 18, 2016 | Initial Version | 1.0 |
| Joshua Kemmerer,<br>Tuan Nguyen,<br>Shishir Kharel | February 8, 2016 | Customer feedback | 1.1 |

# Table of Contents

## Contents

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the "Chess Masters" game. This document will illustrate the purpose and complete declarations for the development of system. It will also explain system constraints, interface and interactions between the mobile and web version. This document is primarily intended to be proposed to a customer for its approval, and a reference for developing the first version of the system for the development team.

## 1.2 Scope

This document describes the software requirements for "Chess Masters." This document is intended for the use of the developers, testers, and users of the application.

## 1.3 Overview

This document will discuss information regarding the application "Chess Masters" and its functional and nonfunctional requirements.

# 2 Description

## 2.1 Product Perspective

"Chess Masters" is a web application intended to enable two people to play against each other in a game of chess. It will allow two random users to connect to each other and play a full game of chess.

### 2.1.1 Web User Interface

The "Chess Masters" user interface will be a uniform interface for all users of the application. The web interface will consist of a signup screen, login screen, main screen, waiting-to-connect screen, and a during-game screen.

## 2.2 Product Functions

"Chess Masters" will provide the following functions:

- Allow the user to join the game by entering a name

- Allow the user to connect to another user remotely through the Internet.
- Allow the user to play a chess game with another user remotely.
- Allow the user to forfeit the game while playing.
- Allow the user to replay the game after forfeiting or winning.
- Allow the user to quit the game.

## 2.3 User Description

The users of the "Chess Masters" game are people who would like to play chess with random people on the Internet, or perhaps with friends. The users of "Chess Masters" do not need to have any previous knowledge of how to play chess, although previous knowledge of chess will help the user to win.

## 2.4 Assumptions and Dependencies

For this application, the assumption is that the user has an Internet connection.

## 2.5 Requirements Apportioning

| Priority Level | Description |
|---|---|
| 1 | This is the highest priority level; requirements of this level are essential for the application's functionality and must be fully satisfied and verified in order for the software to be released. |
| 2 | Requirements of this level are not required, but are highly desirable. Requirements of this level are expected to be satisfied, but not fully verified. |

# 3 Specific Requirements

## 3.1 Functional Requirements

### 3.1.1 Web Application

#### *R1.1 General*

R1.1.1 The application will never display a ‘Quit’ button, as the application is on the web, and the user may quit at any time by closing the browser window. **Priority 2**

#### *R1.2 Login Screen*

R1.2.1 The application will display a text input with a label preceding the input field that says ‘Enter your name’. **Priority 1**

R1.2.2 The application will display a button with text saying ‘Log In’. **Priority 1**

R1.2.3 The application will display a message near the text input that the user’s entered name is of invalid length if the user clicks the ‘Log In’ button and the text in the input field is not between 4-20 characters. **Priority 1**

R1.2.4 The application will display a message near the text input that the user’s entered name is already taken if another user has already entered the game with the current user’s entered text.

#### *R1.3 Main Screen*

R1.3.1 The application will display a list of online users. **Priority 1**

R1.3.1a The list will have two columns, one for each user’s name and the other column for a button that says ‘Request Game’

R1.3.1b The list will have one row for every user currently in the main screen

R1.3.2 The application will add the current user to the list described in R.1.2.1, but the current user will not be able to see himself/herself in the list. **Priority 1**

R1.3.3 The application will display a message box to the user, if that user clicks ‘Request Game’ from the list described in R.1.3.1, saying ‘Waiting for user’s response’ with a cancel button to close the message box. **Priority 1**

#### *R1.4 Game Board*

R1.4.1 The application will display a graphical representation of a standard chessboard. **Priority 1**

R1.4.2 The application will display a message box, which can be closed by clicking a ‘Close’ button, to each user to show which player will start the game first upon first starting the chess game. **Priority 1**

R1.4.3 The application will display a timer above the game board for the current user’s amount of time left to make a move. The timer will reset to one minute every time a player makes a move. The timer will count down to zero seconds. **Priority 1**

R1.4.4 The application will warn the user of how many more times the user may allow the timer to run out before automatically forfeiting by display a closable

message box. The user, by default, may allow the timer to run out a total of three times; therefore, the user has a total of three minutes to make a move. **Priority 1.**

R1.4.5 The application will display a button that says 'Forfeit' which will allow the user who clicks this button to forfeit the game. The application will display a prompting message box asking the user if they would really like to forfeit the game, and the message box will have two buttons saying 'Yes' and 'Cancel'. Both players playing a game of chess will be taken to the main menu if a user clicks the 'Forfeit' button and clicks 'Yes' on the prompting message box. **Priority 1**

R1.4.6 The application will allow the user to select a piece on his/her side by clicking on the piece and then the application will highlight all the valid moves for that piece. **Priority 2**

R1.4.7 The application will allow the user to perform a valid move and will not allow the user to make an invalid move. **Priority 1**

R1.4.8 The application will display a message to notify the user that it is his/her turn if it is the current user's turn. **Priority 1**

R.1.4.9 The game will end once either user has reached a checkmate. **Priority 1**

#### *R1.5 Result Screen*

R1.5.1 The application will display a message box to notify each user who is the winner once the game has ended. The message box will contain a button that says 'Proceed to Main Menu' which will allow the user to go back to the main menu. **Priority 1**

## 3.2 Non-functional Requirements (TUAN)

### 3.2.1 Extensibility

R3.1 Chess Masters will be built in a way that it will allow for the program to be easily modified or expanded. Anticipated changes include allowing the user to create the room with password and wait for his/her friend to join the game. **Priority 2**

### 3.2.2 Maintainability

R4.1 Chess Masters will be decoupled into separate components including front end and backend-as-a-service firebase. **Priority 1**

## 3.3 User Interface

### 3.3.1 Web Application Interface

Figure 1 is the screen for creating an account for the web application. It allows the user to enter his/her email address, a password, and his/her name.

Figure 2 is the login screen for the web application. It allows the user to enter his or her username and password or register for an account.

Figure 3 is the main screen of the web application for every user of the "Chess Masters" application

Figure 4 is the waiting screen that a user will see upon clicking "New Game" from the main screen. This screen will only appear momentarily if there is already a user waiting for

another user. If there are no other users, the waiting room will display a notification that the game is waiting for another user.

Figure 5 is the screen that displays the chessboard. There will be a timer at the top for each user to see how much time he/she has for making a move. The screen will also show a "Quit" button for forfeiting the game and going back to the main menu. Lastly, there will be an "Undo" button to request the opposing user whether he/she will allow a reversal of the user's last move.

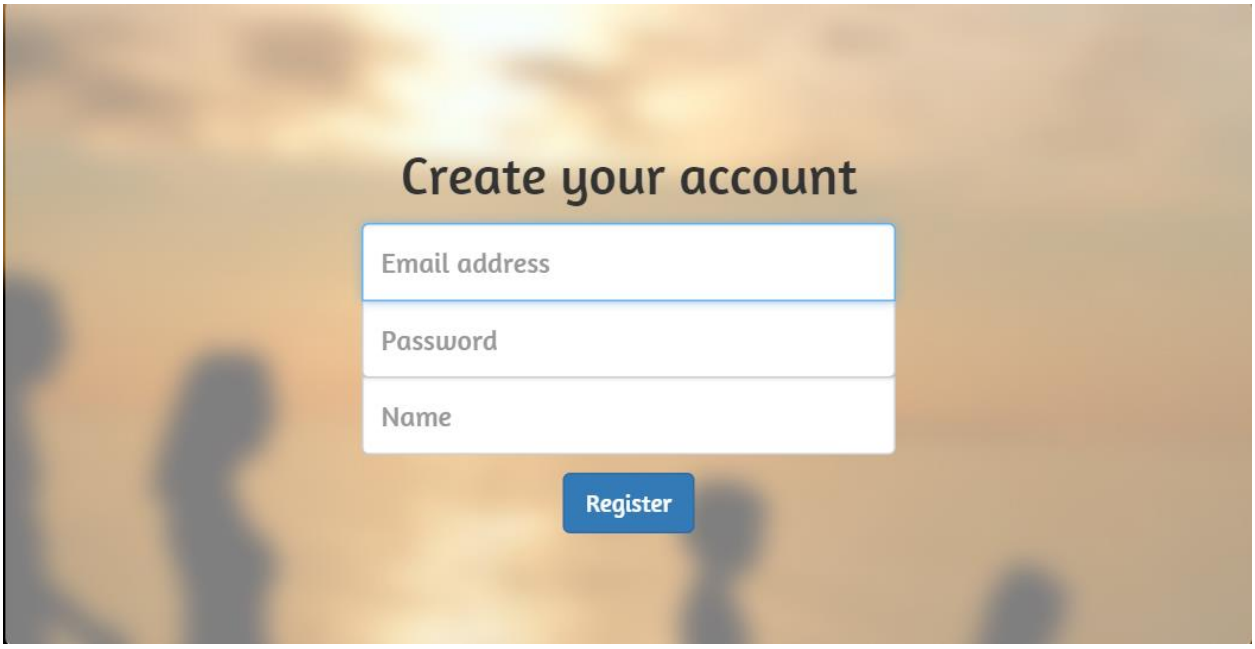


**Figure 1**. Create an account screen for the web application

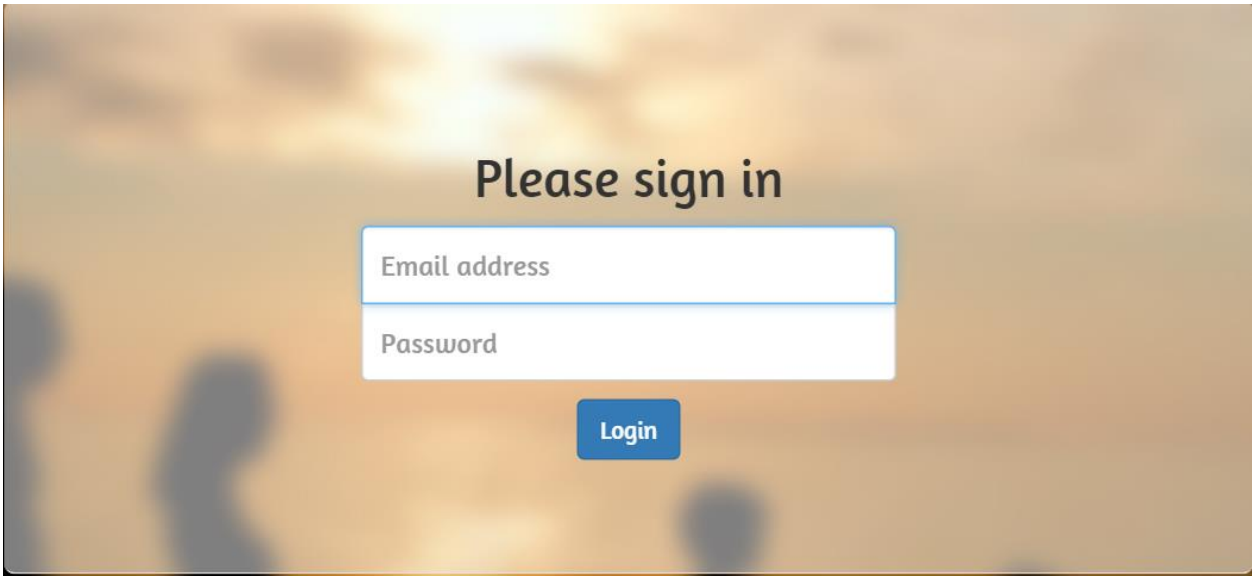


**Figure 2**. Sign in screen for the web application

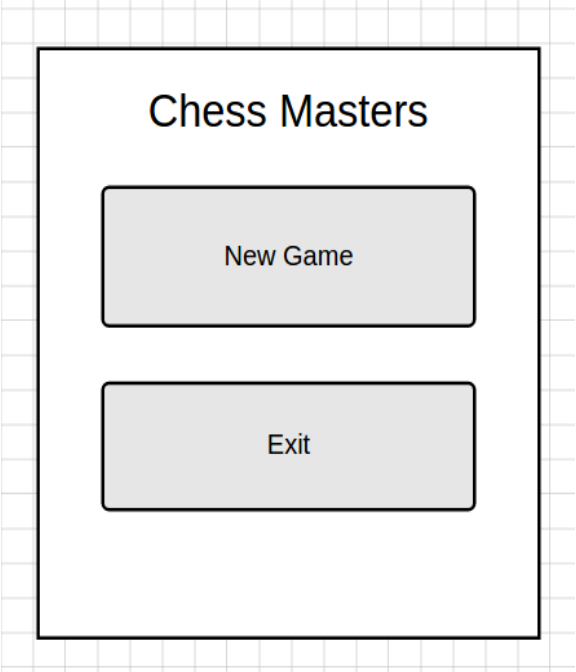


**Figure 3**. The main screen for the web application

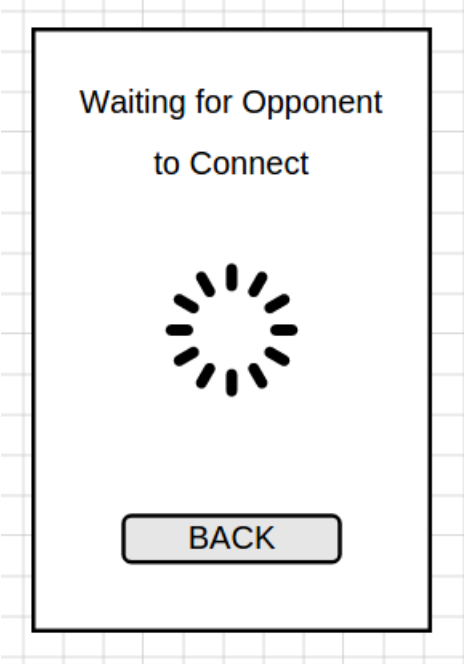


**Figure 4**. The waiting screen for the web application

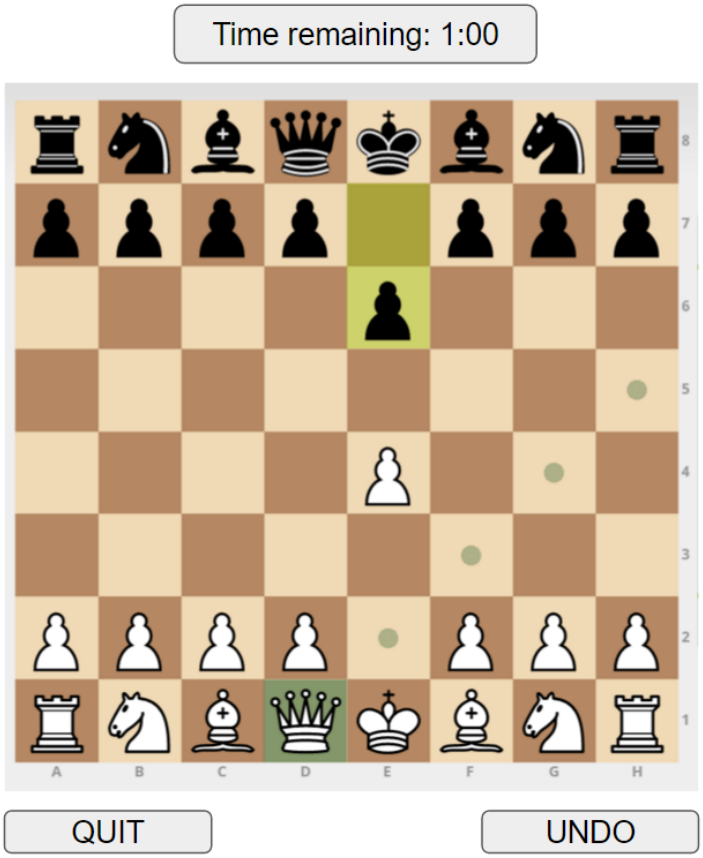


**Figure 5**. The chessboard screen for the web application

# 4 Use Cases

## 4.1 Use Case Flow

### 4.1.1 Logging In

#### Preconditions

**P-1** The player has entered the URL of the “Chess Masters” web application into the player’s web browser

#### Main Flow

The ‘Logging In’ use case is for users to enter the main menu of the “Chess Masters” web application. The player enters his/her name in a text box on the first screen of the web application. Then, the user clicks on the ‘Log In’ button and then the use case ends when the user goes to the main menu (see Use Case ‘Joining a Game’) if the player’s entered name is not already taken (see 4.1.1A-1).

#### Alternate Flows

**A-1** Another player has already checked in with the player's chosen name. The system alerts the player of this conflict with a message that says 'Name already being used by another player.'

### 4.1.2 Joining a Game

#### Preconditions

**P-1** The player has entered his/her name in the log in screen.
**P-2** The player is at the main screen which displays the list of all other online players, who are not already playing games, with an associated button for each of them to let the player send the request to start a match.

#### Main Flow

The 'Join Game' use case is for users to join a game with another player by choosing a specific player (see 4.1.2S-1), or by being in the main menu and having another user's request sent to them (see 4.1.2S-2). If a user's request to join a game is denied (see 4.1.2S-3), then both users are taken to the main menu. If there was an error in sending a request to join a game then the system will alert the user (see 4.1.2A-1) and both users are taken to the main menu. Once either player in the two-way interaction of a request to join a game has accepted, the system will move both users to the in-game mode. If the system was unable to connect both players to a game, the system will alert both involved users (see 4.1.2A-2), and return both users to the main menu. This use case ends when the user has been successfully connected to another use for a chess game.

#### Sub Flows

**S-1** The player clicks on the 'Send request' button next to the online player he/she would like to play with.
**S-2** The system sends a request to the player with the options of accepting or declining within one minute. After one minute, a request is considered to be declined.
**S-3** A user's request to another player is declined by the requested user clicking the 'Decline' button. The system alerts the requesting user that the request has been denied by a popup that may be closed by clicking a 'close' button.
**S-4** A request is accepted by the requested user clicking the 'Accept' button.
**S-5** The system displays the main menu to both players involved in an "accepted" game request

#### Alternate Flows

**A-1** The player sends a request but there is no response within one minute, or there is a connection problem. In this case, the system alerts the player with a popup, which may be closed by clicking on a 'Close' button, stating that: 'The request has been declined' or 'The request could not be sent due to connectivity problems, please check your connection and try again'. After closing the popup (or reestablishing the connection), the player goes to the beginning of this user case.

**A-2** The system is unable to connect the players in a chess game. The system alerts both involved players that a connection could not be made by displaying a popup that may be closed by clicking a 'Close' button, stating that: 'A connectivity problem has occurred, please check your connection and try again'.

### 4.1.3 Make a Chess Move

#### Preconditions

**P-1** Both players have connected to a game
**P-2** The system is currently displaying the game board to the current user

#### Main Flow

The 'Play a Move' use case is for a player in a chess game with another player to make a chess move. The system will display a timer at the top of both user's screen counting down from one minute starting from the beginning of the current user's move (see 4.1.3S-1). The user will select a piece that the user desires to move (see 4.1.3S-2) if it his/her turn. The system will highlight the valid moves for that chess piece to make. The user will select a valid move on the chess board, or the user may not make a valid move (see 4.1.3S-3). The user selects and clicks a valid square, and the system displays the selected piece at the selected square on the chess board. The system allows the other use to make a move next after the current user has made his/her move.

#### Sub Flows

**S-1** The system will display a textual warning to the current user if the timer goes down to zero seconds. The warning will notify the user how many more times the timer may go down before automatically forfeiting the game. The timer may run out a total of three times.
**S-2** The system will only allow the current user to move his/her own pieces.
**S-3** The system will "deselect" the user's selected piece if the user clicks anywhere other than a valid move and within the web application window.

#### Alternate Flows

**A-1** The system is unable to connect the two users and the system then displays a message to both users, saying 'Cannot connect to the other user.' The system pauses the timer in this case.

### 4.1.4 Forfeit a Chess Game

#### Preconditions

**P-1** Both players have connected to a game
**P-2** The system is currently displaying the game board to the current user

### Main Flow

The ‘Forfeit a Chess Game’ use case is for the playing users to forfeit a game and return to the main menu. The system will display a button that says ‘Forfeit’ below the graphical representation of the chess board. The user will click the ‘Forfeit’ button and the system will prompt the user with a dialog box that says ‘Are you sure you would like to forfeit?’ (see 4.1.4S-1). The user will confirm to forfeit the game and both users playing the current chess game will be taken to the main menu (see 4.1.4A-1).

### Sub Flows

**S-1** The dialog box will display options to ‘Confirm’ and ‘Cancel’. If the user clicks ‘Cancel’, then the game will continue as before.

### Alternate Flows

**A-1** The system is unable to connect the current user’s action of forfeiting with the opposing user, and the system will ask the user to make the request again.

## 4.2 Use Case Diagram

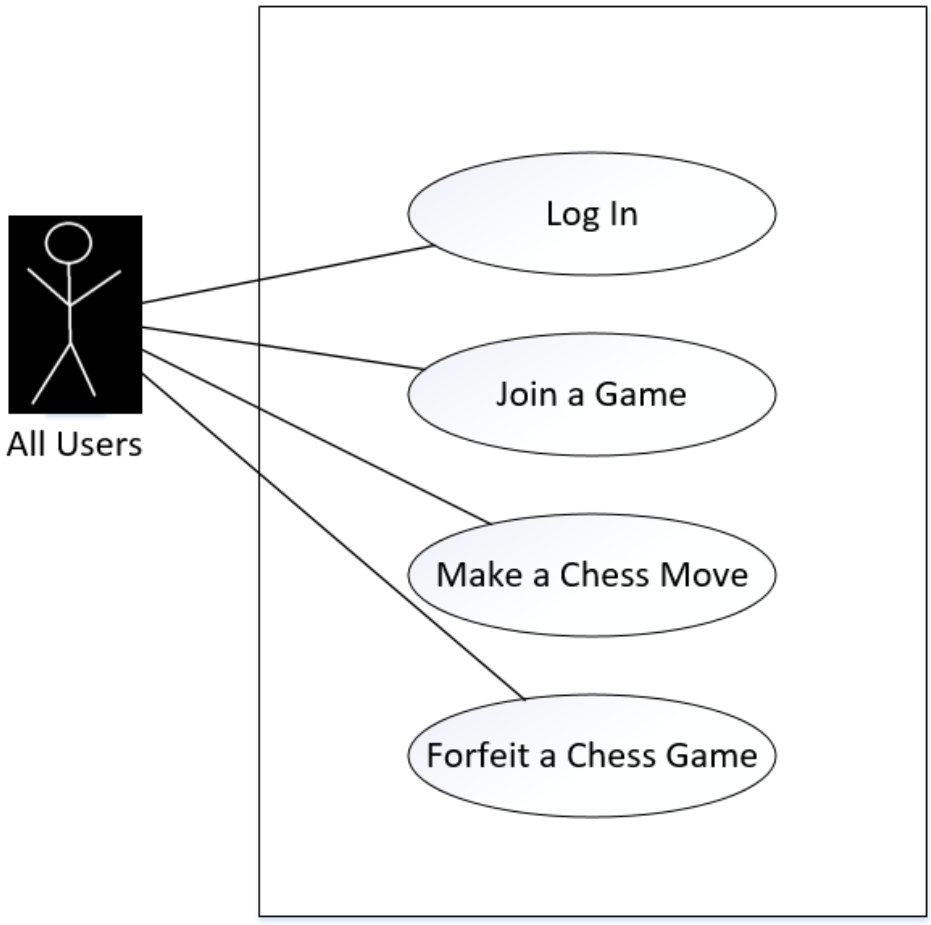


**Figure 6**. The use case diagram for a player using the chess web application

### 4.3 Activity Diagram

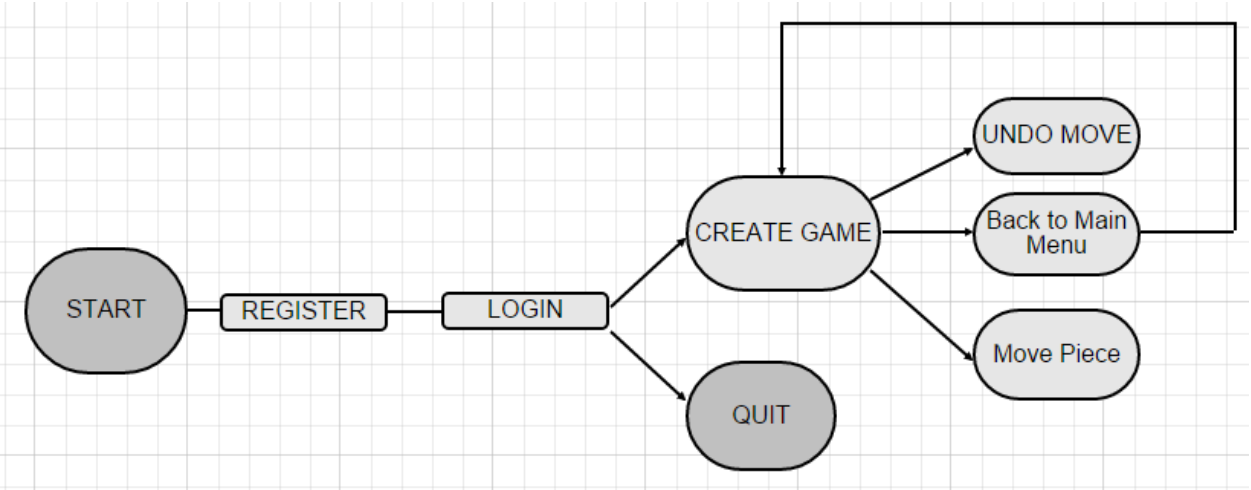


**Figure 7**. The activity diagram for new users' signup, login, and start the game

## 5 System Evolution

Chess Masters is being implemented as a responsive and native mobile & web application that allows users to play chess game remotely while connecting to the Internet. In the future, the app might support allowing user to create a room with password to only allow friends (who knows the password) to join the room and play. The expansion of this application will be based on user's needs.

## 6 Appendix

**Chess Game -** a board game of strategic skill for two players, played on a checkered board. Each player begins the game with sixteen pieces that are moved and used to capture opposing pieces according to precise rules. The object is to put the opponent's king under a direct attack from which escape is impossible.