

Scoremaster Game Scoring

by Team Indecisive Productivity:

Cristina Bindschatel, Luke Jeries, Will Rieske, Taylor Shephard, Dan Wiseman

SLACK CHANNEL: CSC425.slack.com

Final Document

For: Professor Mark Allison

On: December 12TH, 2017

**Table of Contents**

[**Abstract**](#_v5el3eygdqiu) **3**

[1.1. Purpose of System](#_l7hnrimd4c1b) 3

[1.2. Scope of System](#_3tiluuu69mp2) 3

[1.3. Development Methodology](#_7ty6z63phlwj) 3

[1.4. Definitions, Acronyms, and Abbreviations](#_u80e205zxhi7) 3

[1.5. Overview of Document](#_l6bc9ujhc8dt) 3

[**2. Current System**](#_3ilo6zvh8qk7) **3**

[**3. Project Plan**](#_1t3h5sf) **4**

[3.1. Project Organization](#_4d34og8) 3

[3.2. Software and Hardware Requirements](#_2s8eyo1) 4

[3.3. Work Breakdown](#_95nr2cmtli2e) 5

[**4. Requirements of System**](#_3rdcrjn) **6**

[4.1. Functional and Nonfunctional Requirements](#_lnxbz9) 6

[4.2. Use Case Diagram](#_5rcs9alzrihi) 6

[4.3. Requirements Analysis](#_35nkun2) 7

[**5. Software Architecture**](#_1ksv4uv) **7**

[5.1. Overview](#_xj0sw1nnjk5q) 7

[5.2. Subsystem Decomposition](#_2jxsxqh) 7

[5.3. Persistent Data Management](#_z337ya) 8

[**6. Object Design**](#_3j2qqm3) **9**

[6.1. Overview](#_czp3w6px3hkt) 9

[6.2. Object Interaction](#_4i7ojhp) 9

[6.3. Detailed Class Design](#_aubwh1f87jvr) 10

[**8. Glossary**](#_3whwml4) **1**2

[**9. Appendix**](#_c2fnqkdssog7) **1**2

[9.1. Appendix A – Use Cases](#_b8yvw3mqifyj) 12-19

[9.2. Appendix B – User Interface Designs](#_1pxezwc) 20-34

9.3. Appendix C [–](#_1pxezwc) Personas 35

**Abstract**

Overview of the system

**1. Introduction**

In this chapter we will introduce the motivation for building the system

## 1.1. Purpose of System

The purpose of the Scoremaster app is to provide a convenient, easy to use alternative to paper score sheets for the game of 8-ball. In addition, the underlying structure of the system is built in such a way that additional game support could be easily added.

## 1.2. Scope of System

Scoremaster is a mobile application that allows scorekeeping for the popular billiards game 8-ball. With unique features for both single-player outings and regular group-play, Scoremaster provides an easy-to-use scoring experience.

This system will allow players to score games, view a match in progress that is being scored by another player, track your own statistics, and keep track of the groups of players you consistently find yourself playing with. It will not facilitate matchmaking within a group, allow for concurrent score editing, or provide immediate support for scoring games other than 8-ball.

## 1.3. Development Methodology

Agile

## 1.4. Definitions, Acronyms, and Abbreviations

Actors:External entities that interact with the system.

APA:American Poolplayers Association

Game: A single round of 8 ball

Match: A number of games of 8-ball played between two players, cumulating in one final result. 5 matches are played between 10 players.

DD:Design Document.

Deliverable:Work product for client.

## 1.5. Overview of Document

The rest of the document to follow

# 2. Current System

Not applicable.

# 3. Project Plan

## 3.1. Project Organization

**Phase 1 (09/05/2017 – 10/2/2017)**

|  |  |
| --- | --- |
| Taylor Shephard | Designer |
| Will Rieske | System Architect |
| Luke Jeries | Leader |
| Cristina Bindschatel | Designer |
| Dan Wiseman | Editor |

**Phase 2 (10/3/2017 – 12/20/2017)**

|  |  |
| --- | --- |
| Dan Wiseman | Leader |
| Luke Jeries | Validator/Architect |
| Cristina Bindschatel | Minute Keeper |
| Taylor Shephard | Designer |
| Will Rieske | Editor |

## 3.2. Software and Hardware Requirements

Hardware: Laptop/PC, Android cell phone

Software:Android Studio, Slack, MS Word or similar, Firebase

## 3.3. Work Breakdown

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Task #** | **Task** | **Description** | **Duration** | **Dependencies** |
| 1 | Project Plan | Get to know team members, brainstorm, assign roles, decide project topic | 3 days |  |
| 2 | Create use cases | Identify use cases, assign use cases to team members, each team member develops their assigned use cases | 8 days | 1 |
| 3 | Review and completion of use cases | Present use case diagrams to professor, correct use cases | 5 days | 2 |
| 4 | Storyboarding, User Experience | Create the basic layout of the application, think about the user's experience with the app and adjust accordingly | 7 days | 3 |
| 5 | Creation of SRD | Complete SRD, create Power Point presentation, rehearse presentation | 12 days | 3 (M1) (D1) |
| 6 | Presentation of SRD | Present SRD to class, submit SRD to professor | 1 day | 5 |
| 7 | Software Architecture | Divide project into subsystems, identify objects, complete design document, | 2 days | 5 |
| 8 | Object Design | Transition of software models into source code. | 18 days | 5 |
| 9 | Creation of DD | Finalize the DD. | 26 days | 8 (M2) (D2) |
| 10 | Implementation | Database design. Interface Layer, Application Layer and Storage Layer coding. | 40 days | 9 (M3) |
| 11 | Testing Process | Subsystem, System and Evaluation tests. Creation of the User’s Guide. | 16 days | 9 |
| 12 | Creation of FD | Complete FD, create Power Point presentation. | 23 days | 10, 11  (M4) (D3) |
| 13 | Presentation of FD | Present and submit the FD. | 1 day | 12 |

M – Milestone D – Deliverable

# 4. Requirements of System

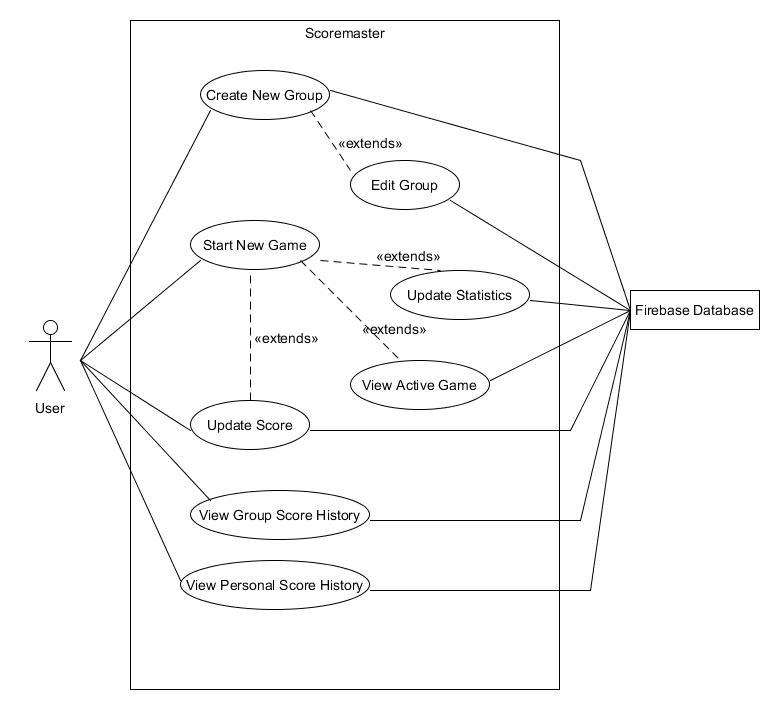
## 4.1. Functional and Nonfunctional Requirements

Functional Requirements: Game score tracking, viewing completed match results or match in progress, create and maintain persistent groups of players, track individual players statistics based on matches they’ve completed.

Nonfunctional Requirements: Score update time of <10 seconds, Match view retrieval time <5 seconds, Statistics calculation time < 5 seconds

## 4.2. Use Case Diagram

The next figure depicts the interaction between the actors and the use cases.



*Figure 1: Use Case Diagram*

## 4.3. Requirements Analysis

After considering our own goals for the project as well as the requirements of our users we, as the designers of the system, believe that our use cases and the requirements that they reflect have fully encompassed the needs of our potential clients. We will devise the most suitable software architecture that meets the needs of our users and design our product in such a way that the user has a very streamlined experience. This will ensure the attainment of customer’s expectations as well as providing a functional final product.

# 5. Software Architecture

## 5.1. Overview

After evaluating and analyzing the main structure and functionalities of our project, the Client Server Architecture was the best fit for our development team. Separating the needs of the system into these two major development environments allows our team members to specialize and focus on the sub systems they’re best suited for.

## 5.2. Subsystem Decomposition

## 

Figure 2 - Client-Server Architecture

## 5.3. Persistent Data Management

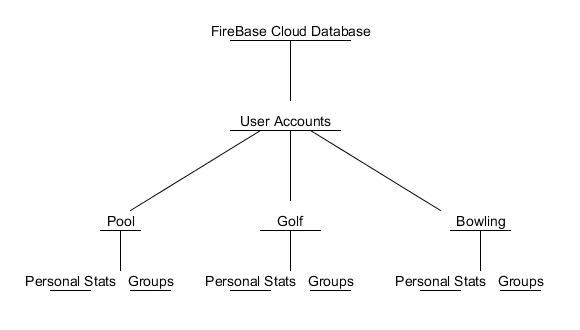


Figure 3 - Persistent Data Management

# Firebase(Cloud) Storage

* User Accounts
  + Pool
    - Personal Stats
    - Groups
  + Golf
    - Personal Stats
    - Groups
  + Bowling
    - Personal Stats
    - Groups

# Device Storage

* Device settings
* Remember Username
* Remember Password

# 

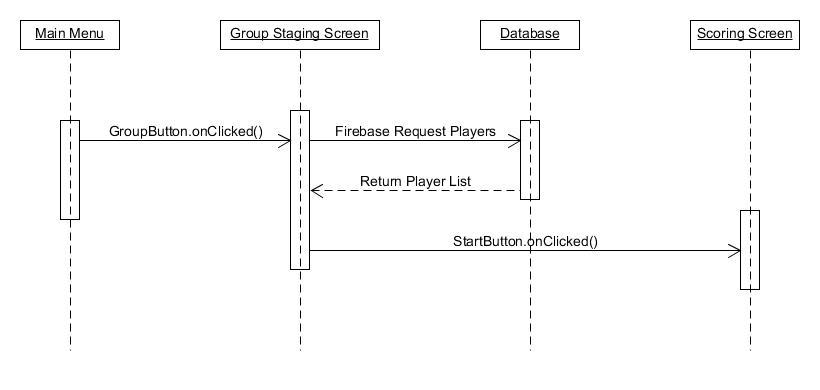
# 6. Object Design

This section will build on the architecture and design.

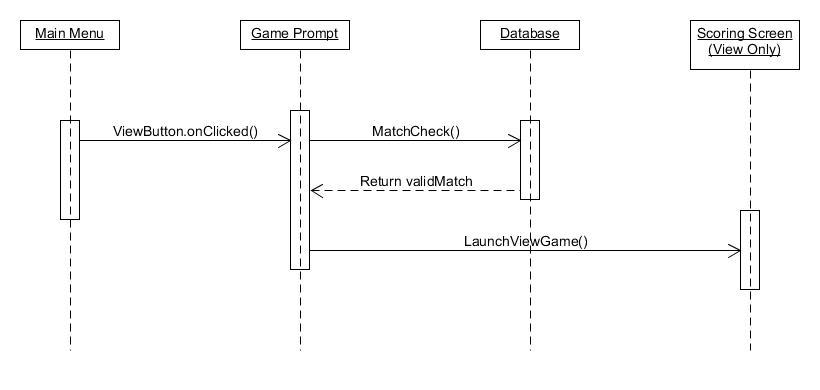
## 6.1. Overview

Our object design focused on our plan to make as much as possible of our system easily expandable. Finding the bare minimum ways to tailor classes to a specific game made it so most of our system is applicable to all game types within the Scoremaster app family.

## 6.2. Object Interaction



*Figure 4: Sequence Diagram for “Start a new Group Game”*

**

*Figure 5: Sequence Diagram for “View Active Game”*

## 6.3. Detailed Class Design

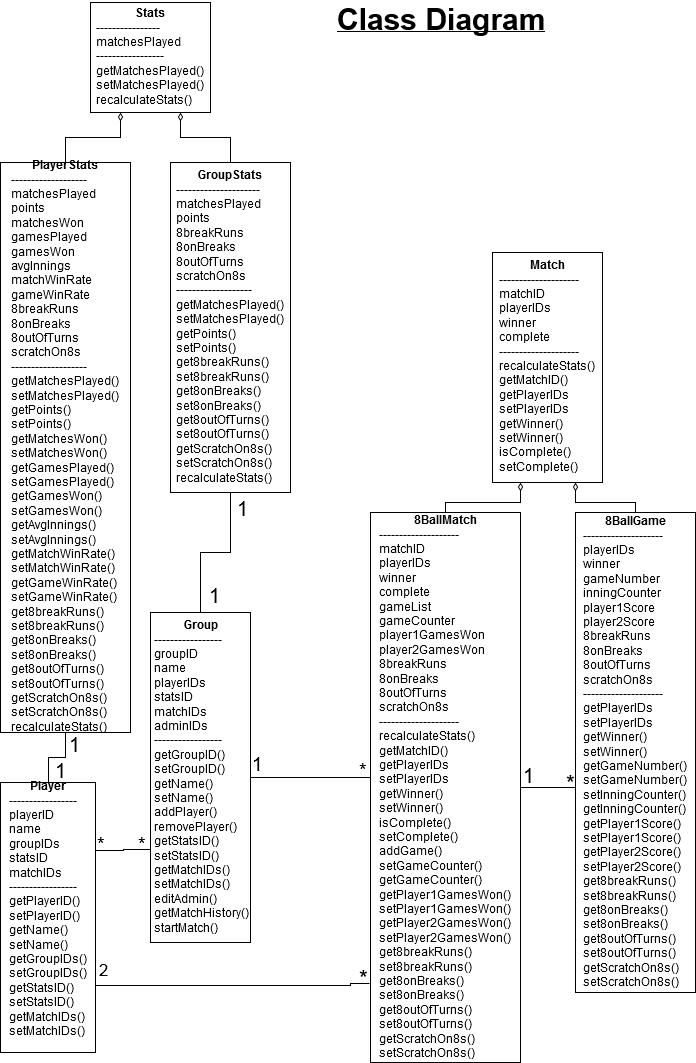


Figure 6- Full Sequence Diagram

# 

Figure 7 - Compressed Class Diagram

# 8. Glossary

# 9. Appendix

## 9.1. Appendix A – Use Cases

*Use Case ID*: **SM801**

*Scenario*: **Start a new Group Game**

*Actor:* User

*Pre-Conditions:*

1. User is logged in

2. User is part of a group with more than 1 member

*Description:*

1. User opens the app
2. User selects the “groups” option on main menu
3. System switches to Groups screen
4. User selects the group they wish to play with from drop down menu.
5. System fetches the group player list
6. User selects the players for the game and confirms
7. System logs a new game instance
8. System loads game scoring screen

*Post Conditions:*

1. A new game entry has been stored for the group.
2. Group statistic for games played increases by 1

*Alternate Courses of Action:*

1. In step D.5, if exactly 2 players are not selected, the system shall prompt the user to enter the correct number of players.

*Exceptions:*

1. System fails to retrieve group player list

*Related Use Cases:*

Start a new Quick Game, Enter Score for Round, Edit Group

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency:* About 3 uses per week

*Criticality:* High. Essential to repeated group play

*Risk:* Medium. Misuse is logged, group access not restricted

*Constraints:*

Must log game results in group history within 5 seconds of match completion

Must update statistics of involved players within 3 seconds of match completion

---------------------------------------------------------------------------------------------------------------------**Modification History:**

*Owner:* Luke Jeries

*Initiation date:* 10/12/2017

*Date last modified:* 10/12/2017

*Use Case ID:* **SM802**

*Scenario*: **Add a member to a Group**

*Actor:* User

*Pre-Conditions:*

1. User is logged in
2. User is part of a group

*Description:*

1. User opens the app
2. User selects the “groups” option on main menu
3. System switches to Groups screen
4. User selects the group they wish to add a player to
5. System fetches the group player list
6. System switches to group game staging screen
7. User selects the add new player option
8. System prompts User for new player’s ID
9. User enters new player’s ID and desired nickname
10. System adds the new player to the player list

*Post Conditions:*

1. Added User’s group list includes new group
2. Group player list includes new member

*Alternate Courses of Action:*

1. In step D.9, the user may leave the player ID blank to add a member with no account

*Exceptions:*

1. Invalid player ID is entered, no new player is added to group

*Related Use Cases:*

Edit Group

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency*: Low About 3 uses per week

*Criticality:* High. Essential to repeated group play

*Risk:* Medium. Vulnerable to invite spam

*Constraints:*

New member’s group list must include new group within 45 seconds

Must update member list of all players in group within 30 seconds

---------------------------------------------------------------------------------------------------------------------

**Modification History:**

*Owner:* Luke Jeries

*Initiation date*: 10/12/2017

*Date last modified:* 10/12/2017

*Use Case ID:* **SM803**  
*Scenario:* **View Group Score History**  
 *Actors:* User  
 *Pre-conditions:*

1. User is part of at-least 1 group

2. User is logged in  
 *Description:*

1. User opens the app
2. User Clicks Stats Button on Main Menu
3. System brings User to Statistics Screen
4. User Clicks on Groups drop-down box
5. System provides a list of groups the user is a part of
6. User selects a group they wish to see the history for
7. User clicks the History button under the drop-down box
8. System pulls score screens from previous games the selected group has played
9. System brings user to a new screen and lists the score screens by session number and date played
10. User clicks the session he wants to view the score screen for
11. System shows the score screen for that particular session

*Post-conditions:*

1. User Remains at score screen until he clicks home button or back button  
*Alternative courses of action:*  
1. 7.b User can click on the group drop down box again, then goes to step 4  
*Exceptions:*

1. If system cannot pull the score screens display an error message

2. If system takes to long to pull score screens display an error message and ask  
 user if they want to continue waiting or go back to main menu  
*Related use cases:*  
 Scoring a game

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**  
 *Frequency:* Medium, could be used often, 8 or more uses per week

*Criticality: Medium-High, depending on the user this could be essential or not*

*important at all*  
*Risk:* Low-medium, only apparent risk would be too much time used to implement

this use case

*Constraints:*  
System must pull game score histories in under 8 seconds  
System provides list of groups in under 2 seconds

---------------------------------------------------------------------------------------------------------------------

**Modification History:**  
 *Owner:* Taylor Shephard  
 *Initiation date:* 10/13/2017

*Date last modified:* 10/13/2017

*Use Case ID:* **SM804**  
*Scenario:* **Rename Group**  
 *Actors:* User  
 *Pre-conditions:*

1. User is part of at-least one group

2. User is logged in

3. User must have group edit permissions

*Description:*  
 1. User Starts App  
 2. User clicks on Group button on Main Menu  
 3. System brings user to group screen  
 4. User clicks on Groups drop down box  
 5. System pulls and lists groups the user is currently apart of in drop down

box   
6. User clicks on group he wishes to change name for from drop down

box  
7. System pulls information for selected group and displays on screen  
8. User clicks edit name button located to the right of the group name

(will look like a little pencil icon)  
 9. System Displays a text box and asks user to type in new name  
 10. User types in the new name for selected group  
 11. User clicks ok  
 12. System checks if written name is the same as another groups  
 13. Group name is not taken so system sets selected group name to

name typed in   
14. System displays that the name of the group has been changed

*Post-conditions:*  
1. The selected group's name is set to the new one  
*Alternative courses of action:*  
13.b Group name is taken so system displays that the name is already taken and asks user to enter a name, then goes to step 12  
*Exceptions:*  
1. If system cannot pull group info in specified time display error message

2. If system cannot change name of group display error message  
*Related use cases:*  
Creating a Group, editing a Group, adding member to Group.

---------------------------------------------------------------------------------------------------------------------**Decision Support:**  
 *Frequency:* Very low, no more than once a week  
 *Criticality:* Low, not very essential   
 *Risk:* Medium, could cause issues or be problematic if the group name is

changed abusively  
*Constraints:*  
System must pull group info in less than 2 seconds.

---------------------------------------------------------------------------------------------------------------------**Modification History:**

*Owner:* Taylor Shephard  
 *Initiation date:* 10/13/2017  
 *Date last modified*: 10/13/2017

*Use Case ID:* **SM805**

*Scenario:* **View active game**.

*Actor:*SM user.

*Pre-conditions:*

1. User must be logged into the system

2. Another user must have previously created an active game

*Description:*

1. Use case begins when SM user opens the app
2. The system shall provide the SM user with the menu
3. The SM user shall choose the **Join Game** button
4. When the request is received, the system shall prompt the SM user to scan a game code (possibly QR code) via the camera.
5. The SM user shall provide the code.
6. The system shall validate the code with the DB.
7. Use case ends when the system can find game and present the SM user with the **Gameplay Screen** in view-only mode.

*Post-conditions:*

1.The number of join-game requests stored in the system has increased by one.

2. The request has been saved in the system.

*Alternative Courses of Action:*

1. In step 4, the user has the option to cancel the request.
2. In step 6, if the code is incorrect an error message will be displayed.

*Exceptions:*

1. If there is no internet connection, the code won’t be able to check with the database.

*Related Uses Case:*

Creating a Game.

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency:* On average 2 requests are made weekly by SM user.

*Criticality:* Low. This feature is not a show stopper, but we would like to implement this to make the app stand out from others.

*Risk:* Medium. Implementing this use case requires camera accessibility for QR code (if we do a number, then there is less risk).

*Constraints:*

Non-functional requirements – Users must be able to access the new game screen within 5 seconds. With proper internet connection, code must validate within 10 seconds.

---------------------------------------------------------------------------------------------------------------------

**Modification History:**

*Owner:* Cristina Bindschatel

*Initiation date:* 10/12/17

*Date last modified:* 10/12/17

*Use Case ID:* **SM806**

*Scenario:* **Scorekeeping a Game**.

*Actor:*SM user.

*Pre-conditions:*

1. User must be logged into the system

2. User must be the creator/admin of a game

3. The game has begun, or is in progress

*Description:*

1. Use case begins when SM user selects an element to update via toggle arrows or a checkbox (time out, 8 on break, 8 ball and run, 8 out of turn, scratch on 8, or win)
2. The interface updates the changes.
3. A request is sent to the database to save the new score.

*Post-conditions:*

1. The request has been saved in the system.

*Alternative Courses of Action:*

*Exceptions:*

1. If there is no internet connection, the score won’t be able to update with the database.

*Related Uses Case:*

Creating a Game.

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency:* On average 5 requests are made weekly by SM user.

*Criticality:* High. This is the main feature of the entire app.

*Risk:* Medium. Making sure all of the options work properly will be the biggest challenge.

*Constraints:*

Non-functional requirements – Users must see the interface update instantly.

When a winner is selected, the game must end and move onto the next one. A function we would also like to implement is that the game automatically ends when either of the players maximum wins are met.

---------------------------------------------------------------------------------------------------------------------

**Modification History:**

*Owner:* Cristina Bindschatel

*Initiation date:* 10/12/17

*Date last modified:* 10/12/17

Use Case ID: **SM807**

Scenario: **Log in**

*Actor:* User

*Pre-conditions:*

1. User has account registered.

*Description:*

1. Use case begins when user opens app.
2. User clicks on log in.
3. User types in credentials.
4. System validates credentials.
5. User is taken to home screen.

*Exceptions:*

1. User is not registered.
2. User types in wrong credentials.

*Post Conditions:*

1. The user is logged in to account.
2. User is given access to create group, view history, quick play.

*Related Use Cases:*

Validate account.

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency*: High. User’s will login every time they open the app.

*Criticality:* High. User’s need account to use the app.

*Risk:* High. Logging in will allow users to use app functionality.

*Constraints:*

Non-functional requirements.

---------------------------------------------------------------------------------------------------------------------

Use Case: **SM808**

Scenario: **Create group**

*Actor:* User

*Pre-conditions:*

1. User has logged into account.

*Description:*

1. Use case begins when user clicks Create Group button.
2. The UI shows user’s name with a blank box for a score.
3. The UI shows Add Player option.
4. The user clicks add player to group.
5. The UI shows blank space to type in Player name.
6. Once the name is entered, it appears in the group under the previous player name.
7. Once all names have been entered, the user selects start game option.

*Exceptions:*

1. User does not want to create a group.

*Post Conditions:*

1. The game has been started for the group of players that was created.

*Related Use Cases:*

Log in, Start Game

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency:* Medium. The number of users creating groups to keep track of score

is medium to high.

Criticality: High. The is one of the main functions of the program.

Risk: High. Implementing the use case allows the users to create groups for

tracking Scores.

*Constraints:*

Non-Functional Requirements.

---------------------------------------------------------------------------------------------------------------------

Use Case: **SM809**

*Scenario:* Check Personal Game History

*Actor:* User

*Pre-Conditions:*

1. Have account
2. Played games previously
3. User is logged into account

*Description:*

1. Use case begins when user clicks View Stats/History button.
2. The UI shows the user the View Statistics Screen.
3. The user clicks the Personal Game History button.
4. The UI shows the personal history of the user.

*Exemptions:*

1. User has no personal game history.

*Post Conditions:*

1. The UI shows the View Statistics Screen.

*Related Use Cases:*

Log in, View Personal Statistics, View Group Statistics, View Group Game

History

---------------------------------------------------------------------------------------------------------------------

**Decision Support:**

*Frequency:* Medium. The number of users checking personal history is medium

to high.

*Criticality:* High. The is one of the main functions of the program.

*Risk:* High. Implementing the use case allows the users to track their personal

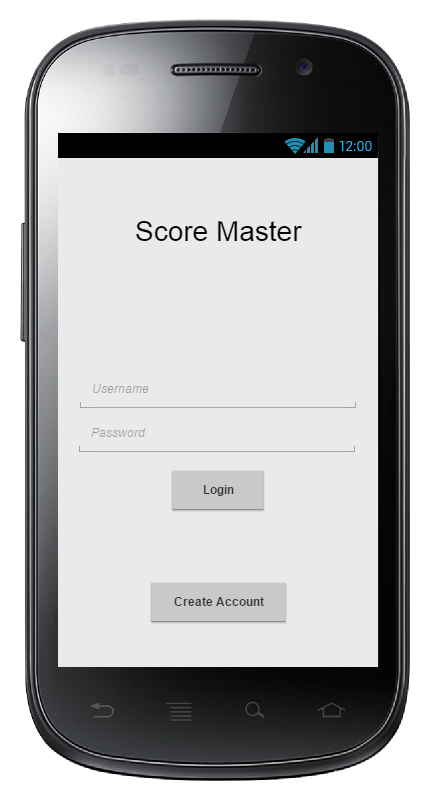
history.

*Constraints:*

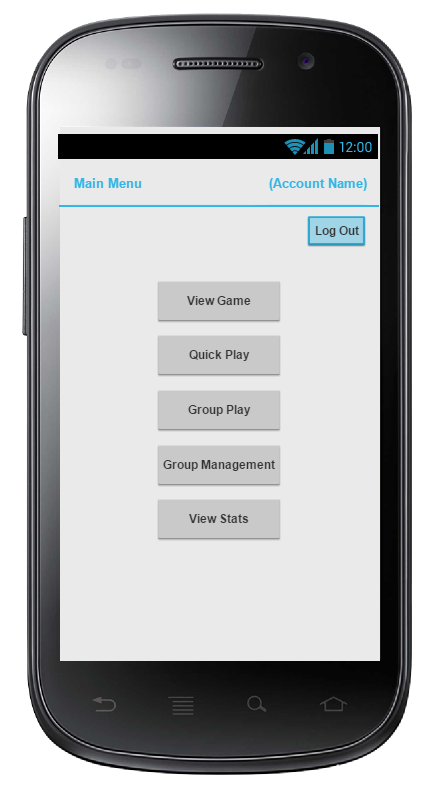
Non-Functional Requirements.

---------------------------------------------------------------------------------------------------------------------

## 9.2. Appendix B – User Interface Designs



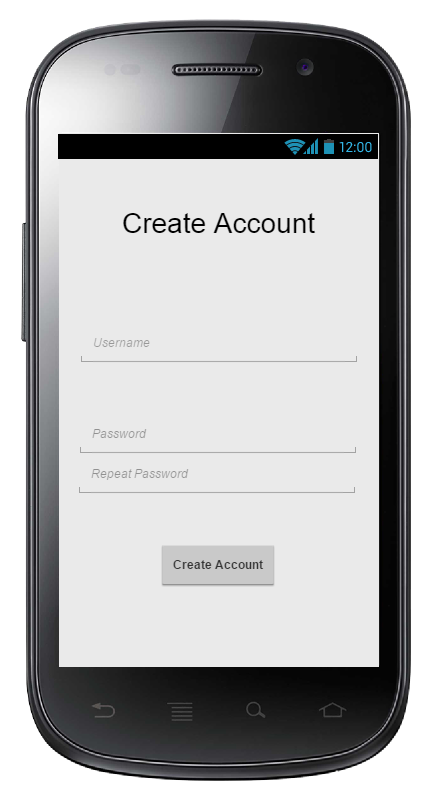
*Figure 8: Login Screen*



*Figure 9: Home Screen (Logged In)*



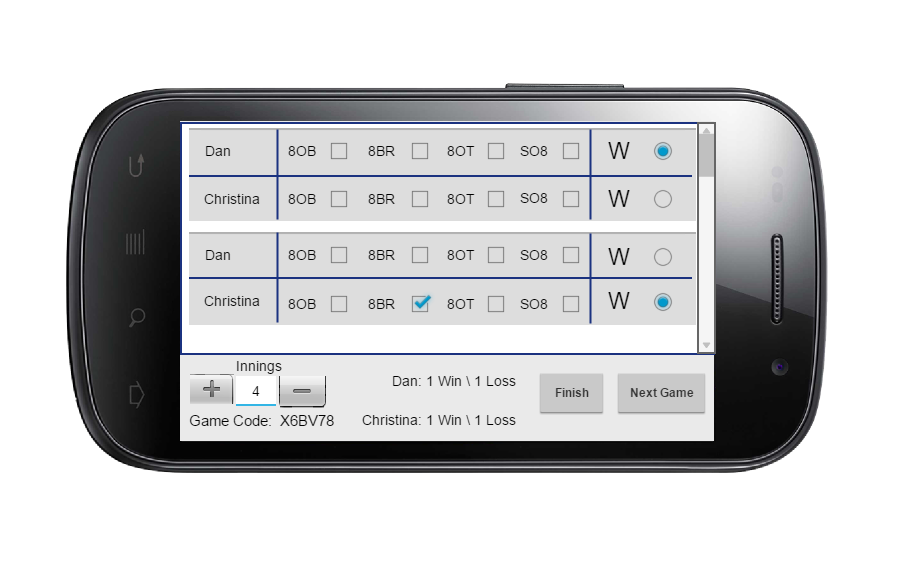
*Figure 10: Home Screen (Not Logged In)*



*Figure 11: Create Account Screen*



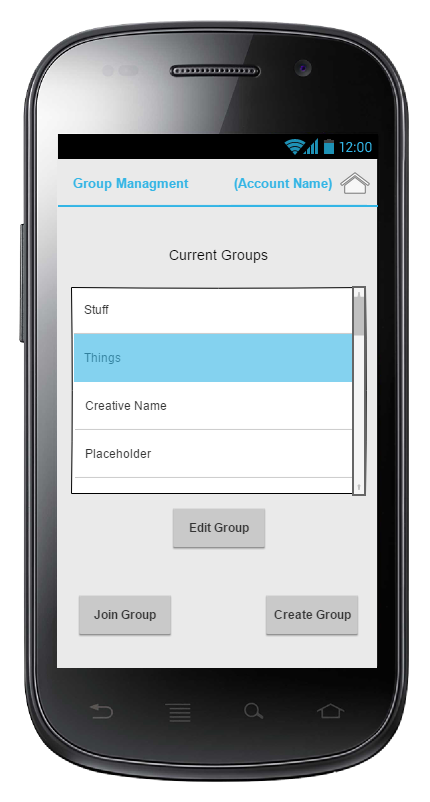
*Figure 12: Splash Screen*



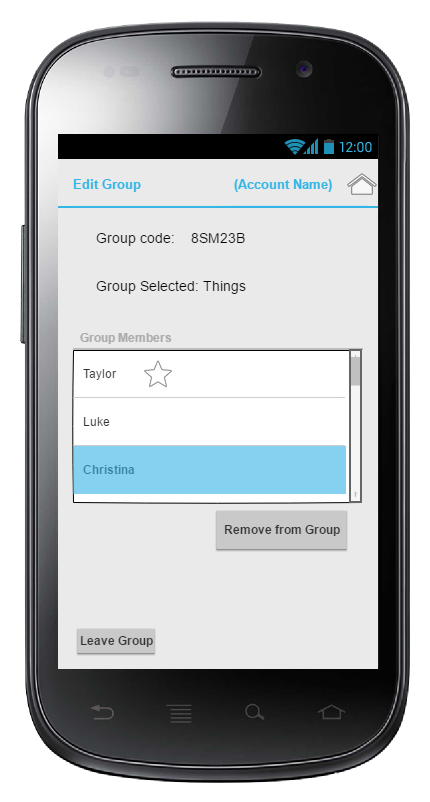
*Figure 13: Scoring Screen*



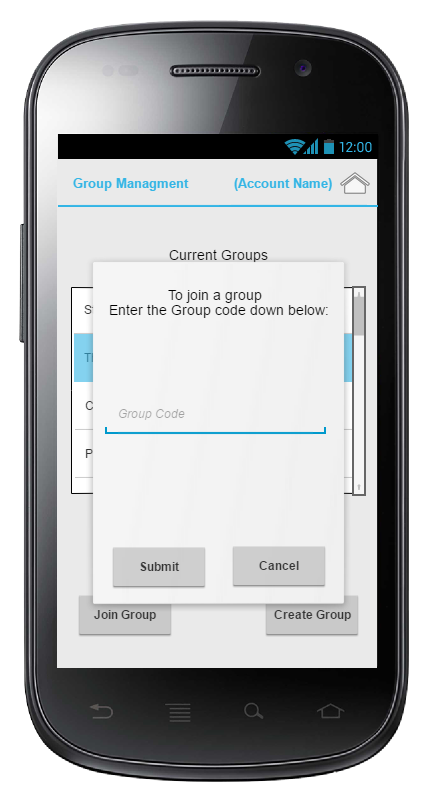
*Figure 14: Post-Game Screen*



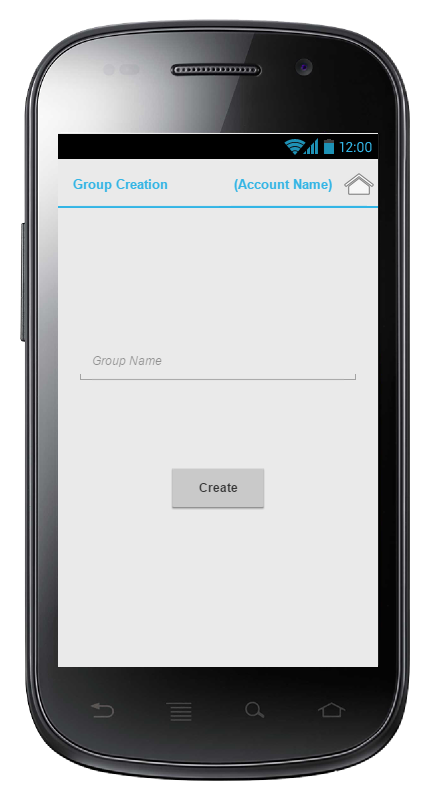
*Figure 15: Group Management Screen*



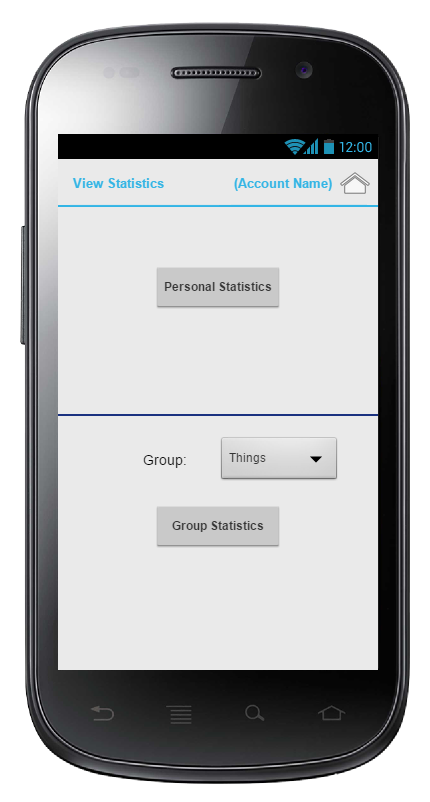
*Figure 16: Edit Group Screen*



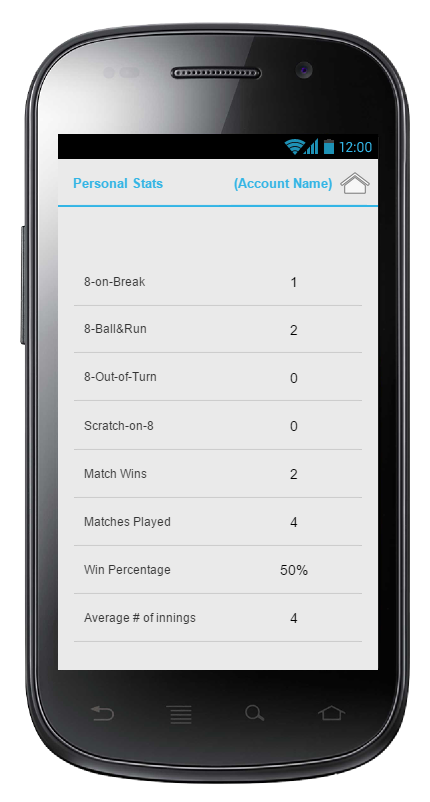
*Figure 17: Join Group Screen*



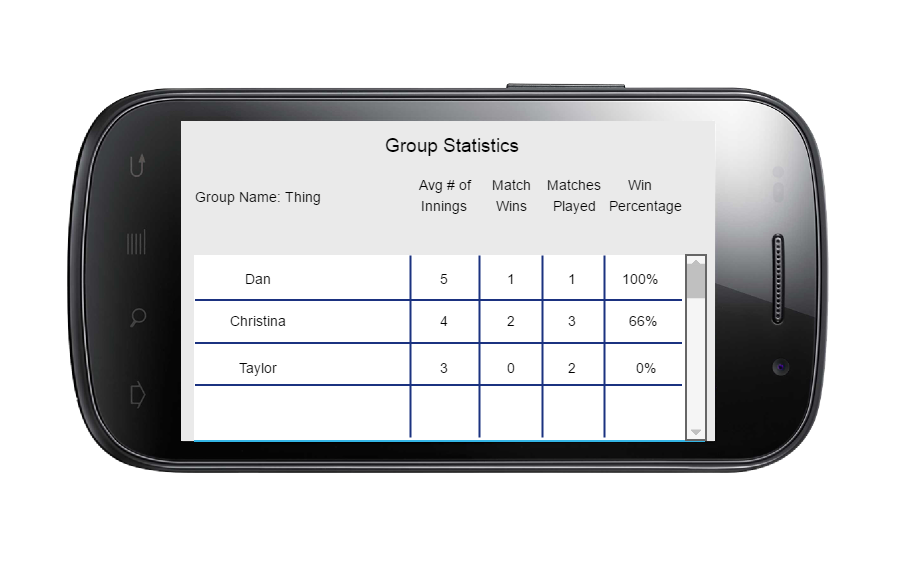
*Figure 18: Create Group Screen*



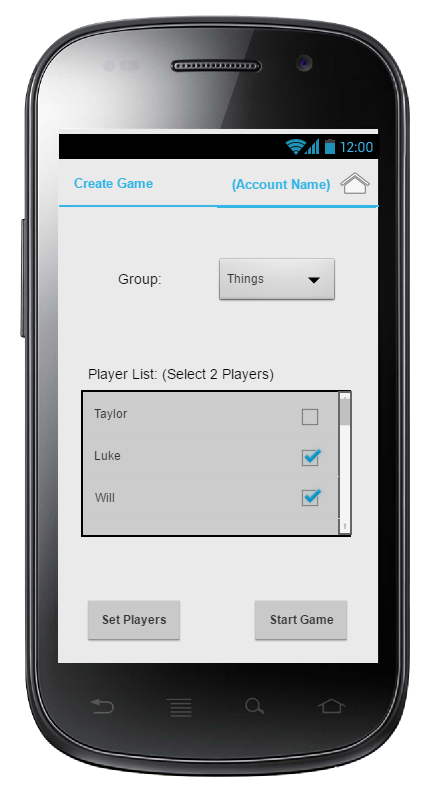
*Figure 19 – Statistics Screen*



*Figure 20: Personal Stats Screen*



*Figure 21: Group Stats Screen*



*Figure 22 – Create Game Screen*

## 9.3. Appendix C - Personas

**Personas**

|  |  |
| --- | --- |
| **Name** | **Description** |
| Bill Jackson | Age 54  Retired from the military  Not very technically savvy  Loves to play pool, darts, and cards with friends  Has basic Android Smartphone |
| Josh Munster | Age 22  Plays in 3 pool leagues  IT enthusiast  Also plays golf, bowling, darts, and volleyball  Always up to date with newer technology and apps |
| Bob Smith | Age 55  Plays pool with friends  Relatively new to using a smartphone  Wants to start keeping track of his matches |
| Tammy Reed | Age 43  Plays in a league  Wants a convenient way to score her team’s matches/games during league |
| Frank Franklin | Age 18  Plays with whoever is at the hall  Wants to track improvement |
| Smallie  Biggs | Age 45  Never has paper to score games  Doesn’t care about persistent tracking |
| Joe  Johnson | Age 20  Plays pool/golf on occasion  Plays for fun  Wants a quick and convenient way to score both games |
| Moe  Monroe | Age 30  Competitive pool player  Wants to be able to track personal stats to see who is the best |