Skillcourt Backend

Andy Martinez & Matthew Santiago

Florida International University

Senior Project

## Legal Notices

Missing

## Abstract

There is a lot involved with the training of soccer players. The current system for training is primitive usually involving an instructor and a physical field for playing. The primary objective is to produce a new, modern, and system for training soccer players. The system will be a program with features that will assist players for learning the skills required on their own.

Implementing this system is revolutionary to the way avid players train in the sport. With the functionality and portability that SkillCourt offers, the user can create a personalized regimen for improving skills; thus, SkillCourt offers an overall improvement to both the soccer training and playing experience for players.

Table of Contents

[Legal Notices i](#_Toc410638547)

[Abstract ii](#_Toc410638548)

[Table of Contents iii](#_Toc410638549)

[1. Introduction 1](#_Toc410638550)

[1.1 Problem Definition 1](#_Toc410638551)

[1.2. Scope of the System 1](#_Toc410638552)

[1.3. Terminology 1](#_Toc410638553)

[1.4. Overview of Document 2](#_Toc410638554)

[2. Current System 3](#_Toc410638555)

[3. Project Plan 4](#_Toc410638556)

[3.1. Project Organization 4](#_Toc410638557)

[3.2. Identification of Tasks, Milestones and Deliverables 5](#_Toc410638558)

[3.3. Cost Estimate 6](#_Toc410638559)

[4. Proposed System Requirements 7](#_Toc410638560)

[4.1. Functional Requirements 7](#_Toc410638561)

[4.2. Analysis of System Requirements (missing) 12](#_Toc410638562)

[5. Glossary 13](#_Toc410638563)

[6. Appendix 14](#_Toc410638564)

[6.1. Appendix A – Complete use cases 14](#_Toc410638565)

[6.1. Appendix B – Use case diagram using UML 18](#_Toc410638566)

[6.2. Appendix C – Static UML diagram 19](#_Toc410638567)

[6.3. Appendix D – Dynamic UML diagrams 20](#_Toc410638568)

[6.4. Appendix E – User interface designs 22](#_Toc410638569)

[6.5. Appendix F - Diary of Meetings and tasks 23](#_Toc410638570)

[7. References 27](#_Toc410638571)

## **Introduction**

(missing 1 or 2 paragraphs to introduce the introduction)

### Problem Definition

As it stands, training for soccer can be a very cumbersome feat for the average person to endeavor unaided. Without proper guidance, progression dwindles and a person may not feel obliged to continue.

### Scope of the System

missing

### Terminology

#### 1.3.1 Acronyms

Missing

#### 1.3.2 Definitions

* **Pad Simulator**: An emulated device which will take the place of SkillCourt pads for testing showcasing purposes. This device will offer all of the features a SkillCourt Pad will offer.
* **SkillCourt**: A system which uses SkillCourt Pads and a player interface for training soccer.
* **SkillCourt Arena**: A 20’x40’ room with SkillCourt Pads on the walls used for soccer training.
* **SkillCourt Pad**: A physical device with a flat surface that can measure and transmit when and how much pressure it received.

### 1.4. Overview of Document

Missing

## **Current System**

Missing

## **Project Plan**

As part of our project’s development process, we have created a detailed plan to organize our work throughout the semester. By following it, we will be able to keep track of both the project’s progress and our performance according to our requirements. This plan includes our individual roles, the personnel organization, and the hardware and software resources needed. It also includes a description of our tasks, milestones and deliverables.

### Project Organization

This section discusses the roles of all team members as well as the technologies that will be used to develop the Skillcourt system.

#### 3.1.1 Project Personnel Organization

This section lists the main role of each of the 2 team members working on the project.

Figure 1. Project Personnel Organization

### Identification of Tasks, Milestones and Deliverables

Table 1. Description of tasks, milestones, and deliverables

|  |  |
| --- | --- |
| Tasks | Task Dependencies |
| 1. Collect User Stories |  |
| 1. Requirement Analysis |  |
| 1. Create Product Backlog |  |
| 1. Setup Development Environment |  |
| 1. Initial Feasibility Study |  |
| 1. Initial Project Plan |  |
| 1. Initial System Design |  |
| 1. Initial Object Design |  |
| Milestone: Feasibility Study and Project Plan Document |  |

Not complete

### Cost Estimate

Table 2. Cost Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Item | Description | Quantity | Cost |
| Human Resources | Team members working on the design, development, and testing processes | 2 | $ 0.00 |
| Hardware Resources | Personal computer | 2 | $ 0.00 |
| Localhost Server to host website | 1 | $ 0.00 |
| Android mobile devices | 1 | $ 0.00 |
| Software Resources | Software (All open source) | - | $ 0.00 |
| Total |  |  | $ 0.00 |

Needs revision

## Proposed System Requirements

This chapter defines the requirements that will be implemented in the SkillCourt-Backend system. It starts by presenting a description of the system in section 4.1. This description is based on the functional and nonfunctional requirements of the system. Then, section 4.2 follows with an analysis of the listed system requirements.

### Functional Requirements

This section presents the functional requirements of the SkillCourt-Backend system. The requirements have been classified in two groups: “The mobile device”, which includes the requirements having to do with the functions of the application in mobile device, and “the webpage”, which includes the requirements having to do with the functions of the webpage.

#### The mobile device application (change them)

* The system shall allow the player to create account with new credentials
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to sign in using his or her credentials
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to enter the application as a guest
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to choose to connect to the pads
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to choose a cognitive skill to practice
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to play goal simulation routine
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to play real game simulation routine
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to create his or her own routine
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to play his custom routine
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to compete against himself
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to compete against other players
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to share his/her performance in social media
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to access his/her statistics
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the coach to log out and exit
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.

#### The webpage

* The system shall allow the player or coach to create account with new credentials
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player or coach to sign in using his/her credentials
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to access his/her statistics
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to share his/her performance in social media
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the player to log out and exit
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the coach to access the statistics of all his/her players
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.
* The system shall allow the coach to log out and exit
* **Usability**: The feature must be easy to follow.
* **Reliability**: System must work flawlessly 99% of the time.
* **Performance**: Data must be save within 5s seconds.
* **Supportability**: This functionality must work for any browser.

### Analysis of System Requirements (missing)

#### Scenarios

#### Use case model

#### Static model

#### Dynamic model

## Glossary

## Appendix

### Appendix A – Complete use cases

#### The mobile device application:

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #112 | |
| **Name:** | Account Creation | |
| **Details:** | This use case allows the players to create their SkillCourt accounts using their credentials | |
| **Actors:** | Player | |
| **Pre-Conditions**:   * Player has entered the application in the mobile device | |
| **Description**:   * As a player, I want to be able to create an account so that I can log in and log out.   **Steps:**   * Use case begins when user presses the “Create Account” button * The system should provide the user with a form to be filled, which includes name, username, email address, password, date of birth, and coach username (optional). * The user should fill out the provided form and press “Submit” * System should record the user’s information in the database and increment the number of members by 1 * The use case ends when the system provides the user with a confirmation of the new account creation. | |
| **Post-Conditions**:   * The user information must be stored in the database | |

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #141 | |
| **Name:** | Enter the application as guest | |
| **Details:** | This use case will allow the user to enter the application as a guest without logging in to a account | |
| **Actors:** | Player | |
| **Pre-Conditions**:   * Player has entered the application in the mobile device | |
| **Description**:   * As a player, I want to be able to enter the application as a guest so that I can play without having to enter my credentials.   **Steps:**   * Use case begins when user presses the “Log in as Guest” button * The system should show a warning saying that if the user enters as a guest, his performance will not be recorded, and present two options: “Accept”, “Go back” * The user presses “Accept” * The use case ends when the system displays the Guest Home View. | |
| **Post-Conditions**:   * The system is aware that the user is a guest and is ready to handle it as such | |

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #142 | |
| **Name:** | Connect to pads | |
| **Details:** | This use case allows the player to find pads that are closed to him and connect to them via Bluetooth. | |
| **Actors:** | Player | |
| **Pre-Conditions**:   * The player has already logged into the account or entered as a guest. | |
| **Description**:   * As a player I want my application to connect to the pads so that I can send information to the pads regarding what routine to play.   **Steps:**   * Use case begins when user presses the search for pads button. * The system will display a list of possible pads. * The user will select the pad he wants to connect to. * The use case ends when the system uses the Bluetooth address to establish a connection.. | |
| **Post-Conditions**:   * The system has all the information needed from the pad in order to start to transfer data. | |

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #192 | |
| **Name:** | Account Management | |
| **Details:** | This use case allows the player to change his account information after his account has been created | |
| **Actors:** | Player | |
|  |  | |
| **Pre-Conditions**:   * The player has already created an account and is logged into said account | |
| **Description**:   * As a user I want to be able to modify my account information so that I can make changes and updates as needed   **Steps:**   * Use case begins when user presses the edit account button. * The system will display the user’s account information * The user will press edit Information * The system will allow the user to modify his information * The user will change the information he wants to change and press done * The use case ends when the system moves to User’s profile view. | |
| **Post-Conditions**:   * The user’s new account settings will be properly changed in the database | |

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #109 | |
| **Name:** | Access Performance Statistics | |
| **Details:** | This use case will allow the user to access his statistics | |
| **Actors:** | Player | |
| **Pre-Conditions**:   * Player has logged in with his credentials | |
| **Description**:   * As a player I want to be able to access the statistics of my performances so that I can isolate weaknesses and see my improvements to focus in the areas of training that I need the most.   **Steps:**   * Use case begins when the user presses the “Statistics” button * The use case ends when the system accesses and displays the player’s statistics | |
| **Post-Conditions**:   * A window with the user’s statistics data is displayed | |

|  |  |  |
| --- | --- | --- |
| **Use case ID:** | SkillCourt-Backend #145 | |
| **Name:** | Collect performance statistics | |
| **Details:** | This use case will allow the user to have his performance statistics stored. | |
| **Actors:** | Player | |
| **Pre-Conditions**:   * The user is already connected to a set of pads | |
| **Description**:   * As a player I want to be able to record my performance statistics so that they are stored.   **Steps:**   * Use case begins when the user chooses the routine he wants to play * The android application will use the name of the routine to get the routine’s specifications from the database and will transmit every step to the pads * The pads will receive the routine and will start illuminating the corresponding lights * The user will hit the illuminated pads * The pads will send information about the user’s hits to the android application * The android application will store the information provided by the pads in the database * The use case ends when the pads inform the android application that the time for the entire routine has expired | |
| **Post-Conditions**:   * All performances of the player are stored in the database | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend #265 |
| **Name:** | Choose from a list of positions |
| **Details:** | This use case will allow the user to choose the position he plays from a list of positions. |
| **Actors:** | Player |
| **Pre-Conditions**:   * The player is in the create account Screen | |
| **Description**:   * As a player I want to choose the position I play from a list of predefined positions so that I don't have to type it in.   **Steps:**   * Use case begins when the user presses position * The system will display a drop-down list of all possible soccer positions * The user will select the specific position he plays * The use case ends when the system displays the name of the position in the position field | |
| **Post-Conditions**:   * The position field is filled. | |

#### The webpage

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend 001 |
| **Name:** | Create Account |
| **Details:** | This use case allows the players and the coach to create their SkillCourt accounts using their credentials |
| **Actors:** | Player, Coach |
| **Pre-Conditions**:   * Player has entered the webpage | |
| **Description**:   * Use case begins when user clicks the “Create Account” button * The system should provide the user with a form to be filled, which includes name, username, email address, password, and coach username (optional). The form should also ask the user to select whether he is a player or coach * The user should fill out the provided form * The user presses “Submit” * System should record the user’s information in the database and increment the number of members by 1 * The use case ends when the system provides the user with a confirmation of the new account creation. | |
| **Post-Conditions**:   * The user information must be stored in the database | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend 004 |
| **Name:** | Access Player Statistics – as player |
| **Details:** | This use case will allow the user to access his statistics |
| **Actors:** | Player |
| **Pre-Conditions**:   * Player has logged in with his credentials | |
| **Description**:   * Use case begins when the user presses the “Statistics” button * The use case ends when the system accesses and displays the player’s statistics | |
| **Post-Conditions**:   * A window with the user data is displayed | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend 003 |
| **Name:** | Log in |
| **Details:** | This use case will allow the users (either player and coach) to enter the application with their credentials |
| **Actors:** | Player, Coach |
| **Pre-Conditions**:   * User has entered the SkillCourt webpage | |
| **Description**:   * Use case begins when the user presses the “Log in” button * The system should provide the user with a form to be filled, asking for username and password * The user should fill out the form and press “Log in” * The use case ends when the system enters the webpage using the user credentials | |
| **Post-Conditions**:   * The system should recognize whether the user is a player or a coach and handle them as such | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend 004 |
| **Name:** | Password Recovery |
| **Details:** | This use case allows the players to recover a forgotten password |
| **Actors:** | Player |
| **Pre-Conditions**:   * The player has already created an account | |
| **Description**:   * Use case begins when user presses the “Trouble Logging In?” button * The system will allow the user to enter the email address associated with the account | |
| **Post-Conditions**:   * The system will send the user his credentials to the associated email address | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend |
| **Name:** | Edit Account |
| **Details:** | This use case allows the player to change his account information after his account has been created |
| **Actors:** | Player |
| **Pre-Conditions**:   * The player has already created an account and is logged into said account | |
| **Description**:   * Use case begins when user presses the edit account button. * The system will display the user’s account information allowing for it to be edited * After the user has edited the information, the user will click the save button | |
| **Post-Conditions**:   * The user’s new account settings will be properly changed in the database | |

|  |  |
| --- | --- |
| **Use case ID:** | SkillCourt-Backend |
| **Name:** | Reset Password |
| **Details:** | This use case allows the player to change the password that is associated with his username. |
| **Actors:** | Player |
| **Pre-Conditions**:   * The player has already created an account and is logged into said account | |
| **Description**:   * Use case begins when user presses the reset password button. * The system will display a form for the user to enter the new password. * After the user has entered the new password, the user will click the reset button. | |
| **Post-Conditions**:   * The user’s new password has been updated in the database.  |  |  | | --- | --- | | **Use case ID:** | SkillCourt-Backend | | **Name:** | Connect to pads | | **Details:** | This use case allows the player to find pads that are closed to him and connect to them via Bluetooth. | | **Actors:** | Player | | **Pre-Conditions**:   * The player has already logged into the account or entered as a guest. | | | **Description**:   * Use case begins when user presses the search for pads button. * The system will display a list of possible pads. * After the user selects the SkillCourt pad he wants to connect to, the system will get the Bluetooth name and address of that pad. | | | **Post-Conditions**:   * The system has all the information needed from the pad in order to start to transfer data. | | | |

Not complete

### Appendix B – Use case diagram using UML

Figure . Use Case Diagram

### Appendix C – Static UML diagram

Figure Class Diagram

### Appendix D – Dynamic UML diagrams

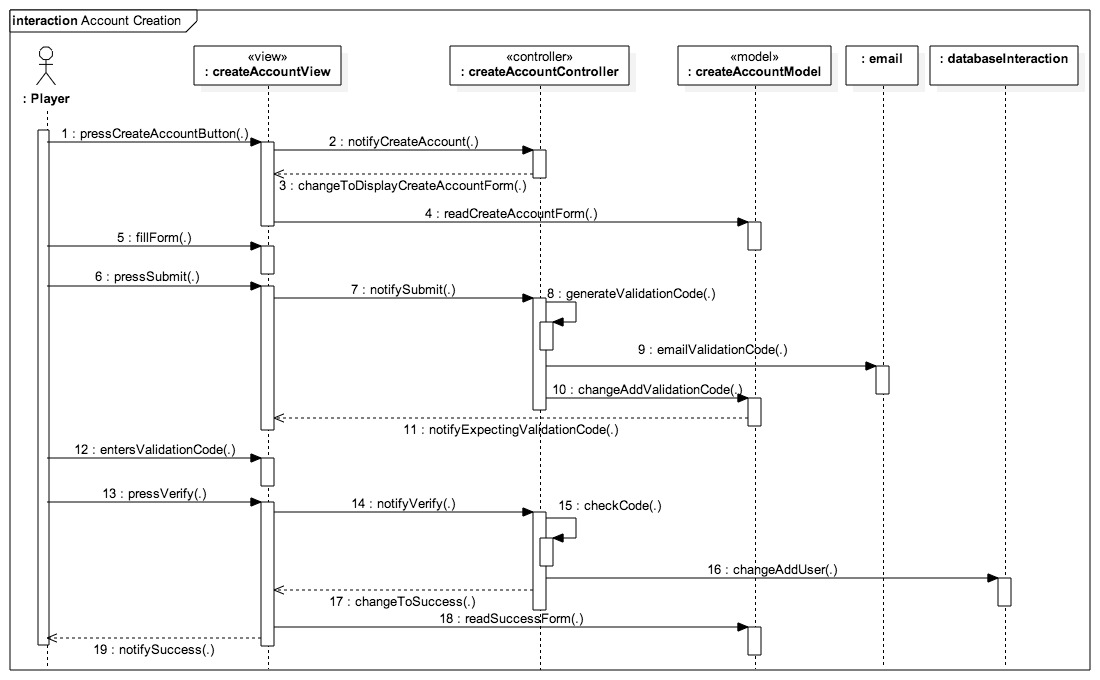


Figure Account Creation Sequence Diagram

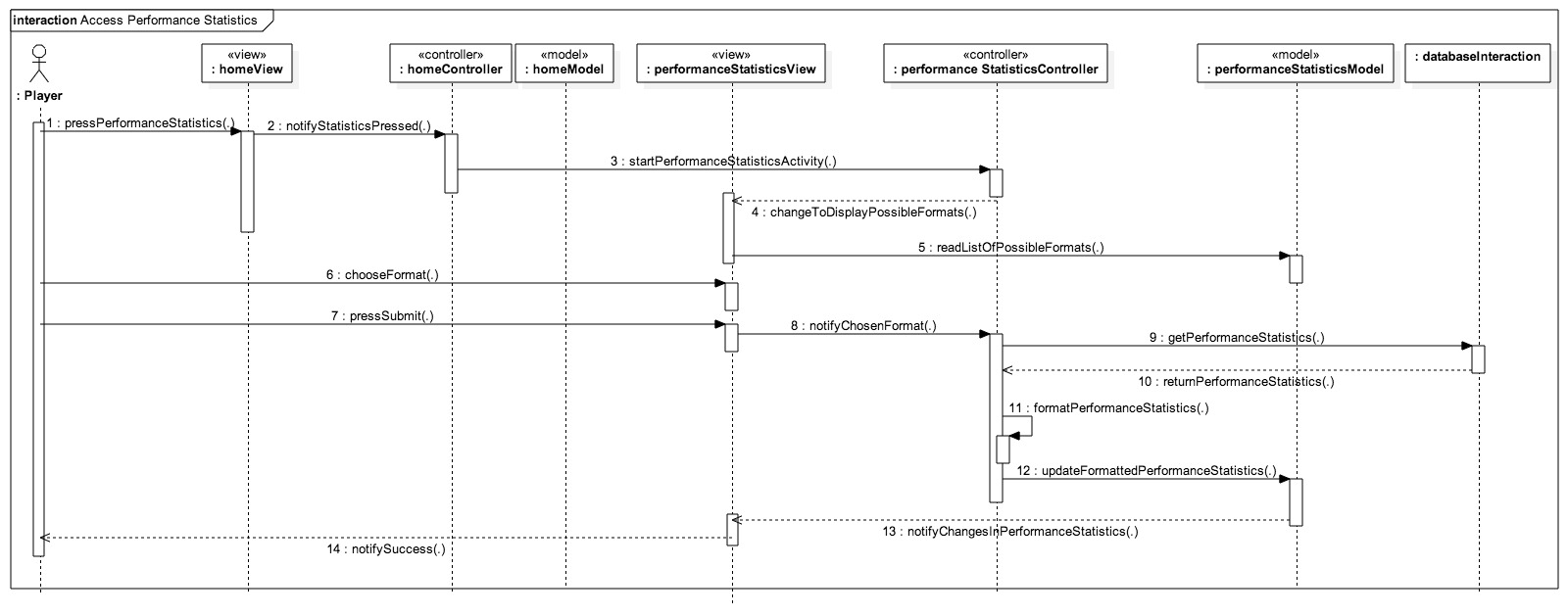
****

Figure Access Player Statistics Sequence Diagram

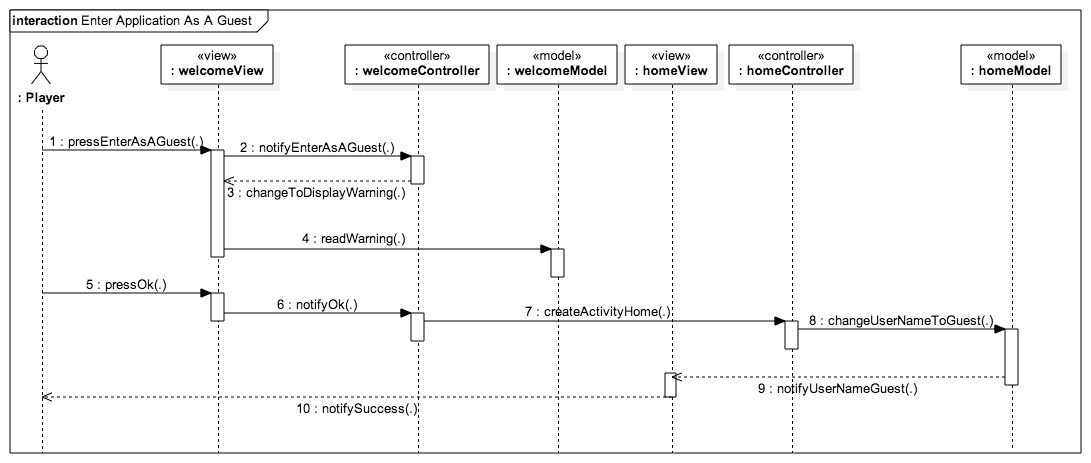


Figure Enter Application as A Guest Sequence Diagram

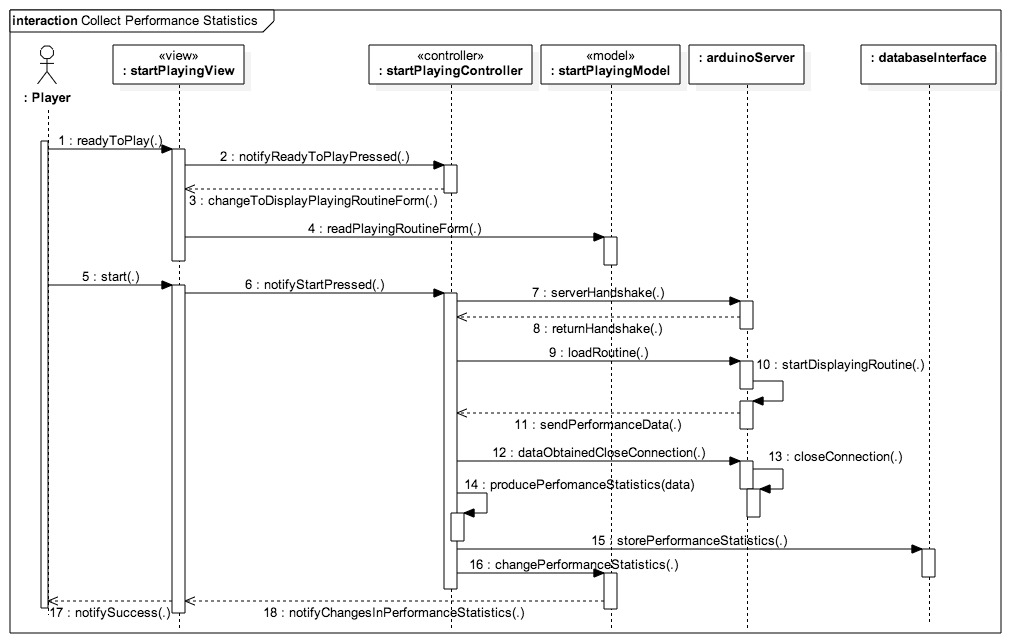


Figure Collect Performance Statistics Sequence Diagram

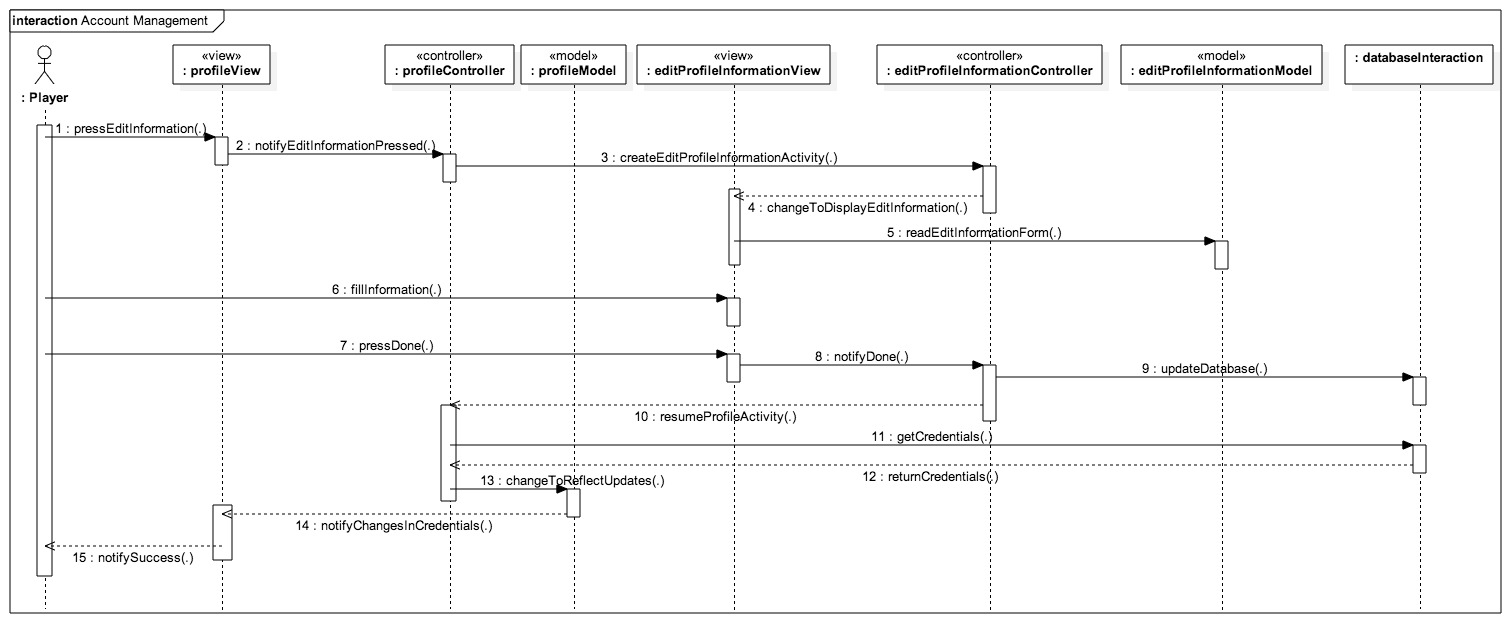


Figure Account Management Sequence Diagram

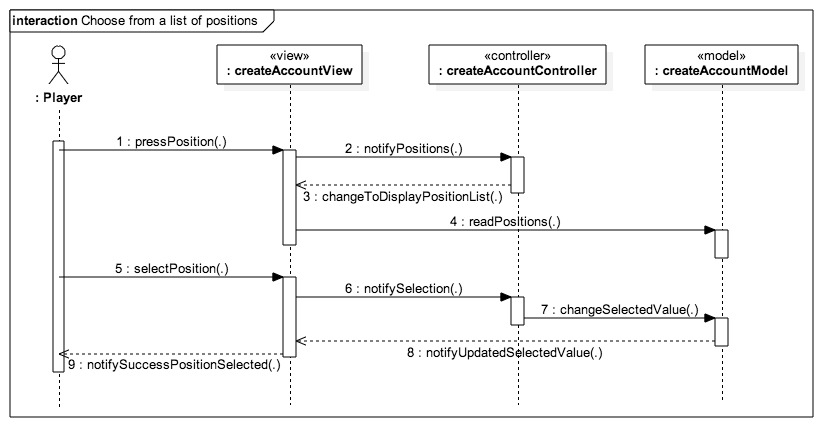


Figure Choose from a list of positions

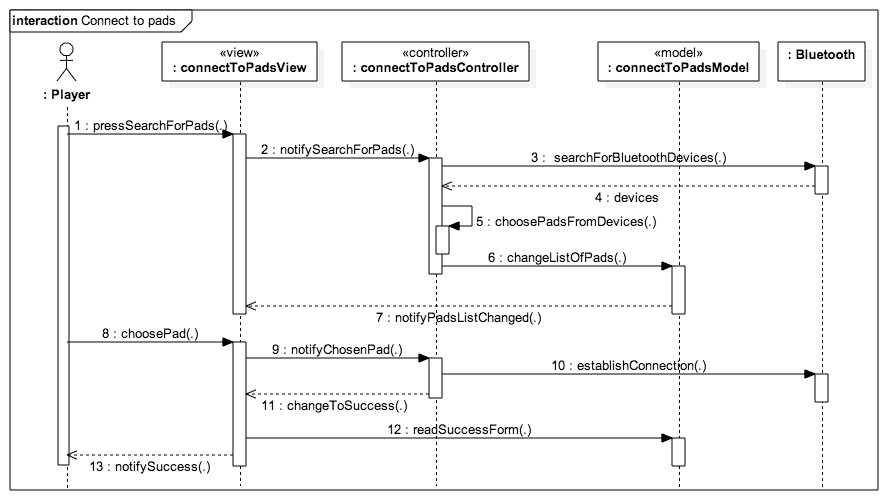


Figure Connect to Pads

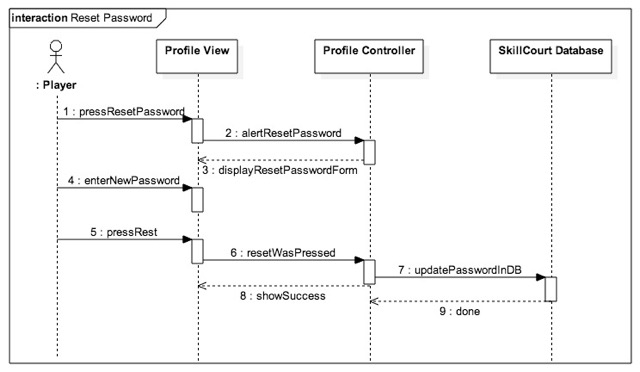
****

Figure Reset Password Sequence Diagram

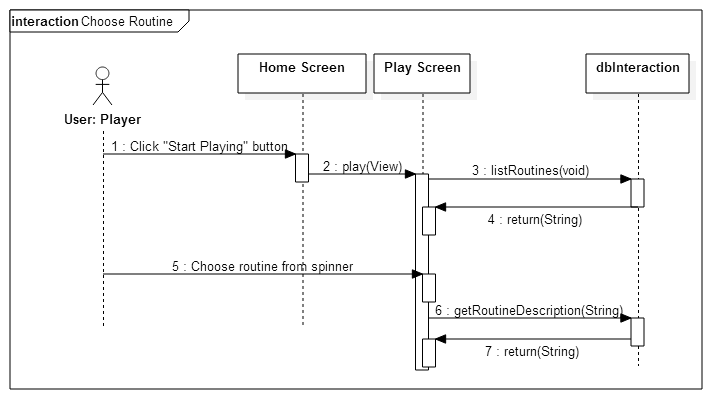


Figure Choose Routine Sequence Diagram

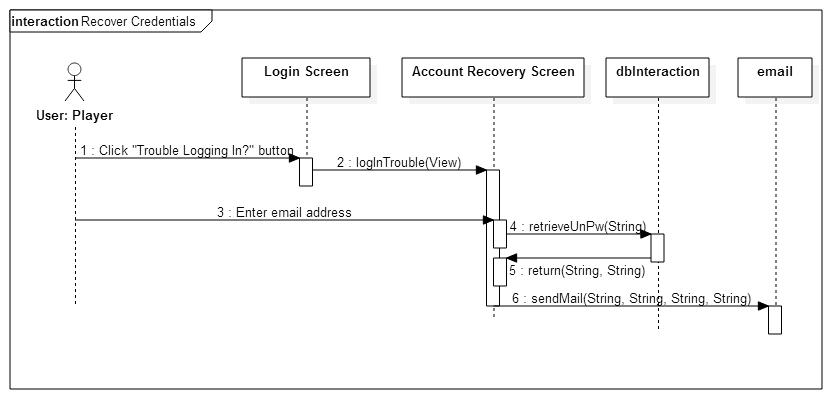


Figure Recover Credentials Sequence Diagram

### Appendix E – User interface designs

### Appendix F - Diary of Meetings and tasks

1. **Meeting 1:**

Monday 1/19/15

@8:57pm

Meeting on Mingle starts

In attendance: Andy Martinez, Matthew Santiago

@9:08pm

Arranging possible meeting times for Requirements Elicitation:

* Tuesday 1/20 9:00 pm
* Wednesday 1/21 before 2:00 pm
* Wednesday 1/21 after 7:30 pm

@9:23pm

Chose for Andy Martinez to be Scrum Master for the first sprint

@9:28pm

Sent message to @Product Owner for their preferred meeting time:

* Awaiting reply…
* **1/20 @10:43am** Reply Received, Meeting Tuesday 1/20 at 9:00 pm confirmed

@9:33pm

Meeting Dismissed

1. **Meeting 2:**

Tuesday 1/20/15

@9:00 pm

Conference Call Started

In attendance: Matthew Santiago, Andy Martinez, Jaime Borras, Gummi

@9:02 pm

Introductions

@9:12 pm

Stories:

1. Skill development (accuracy, speed, intensity)
   1. User chooses skill to train from Android App
   2. Measure pressure, time
   3. 3 colors red blue green
   4. Change to red when underperforming
   5. Ball should take 5 seconds to hit the pad, turn red if user takes too long
2. Tracking and analysis
   1. accuracy, speed, intensity, reaction time
3. Single Player/2 Player
   1. Social Media Connection
4. Mapping a game
   1. Pad light up, hit pad with ball in proper time, another pad lights up, hit with ball in proper time, continue until you make goal (certain number of successful iterations)
   2. Beginner, Intermediate, and advanced levels
5. Simulator (simulate a pad)
   1. Android application?
6. (Backend look at particular game (real game) and simulate a player or play within that game)

@9:32 pm

Set meeting with Gummi on 1/21 at 11:00 am in GL 693

@9:38 pm

Meeting Dismissed

1. **Meeting 3:**

Thursday 1/22/15

@9:00

Conference Call start

In attendance: Andy Martinez, Matthew Santiago, Jaime Borras

@9:07

Review stories

Predetermined Routines:

* Separate Goal Simulation into its own story

Performance Statistics:

* Add statistics about specific game

Competition Mode:

* Separation of social media into a new story is good
* Have different routines for single and multiplayer

Custom User Routine:

Pads Simulator:

Real Game Simulation:

Social Media Sharing:

Website:

* (Differentiate between coach and player?)
* Webpage access

@9:46

Meeting Dismissed

1. **Meeting 4:**

Tuesday 1/27/15

@ 6:30 pm

Meeting starts

In attendance: Jaime Borras, Andy Martinez, Matthew Santiago

**Github upload schedule:**

Matthew upload from 10:00pm - 10:59pm

Andy upload from 11:00pm - 11:59pm

**Weekly in-person meeting:**

Thursday between 2:00pm - 6:00pm to prepare for weekly meetings with project owners

**Coding standards**

Comment Convention:

/\*\*

\* Comment goes here

\* and here

\*/

fun()

{

random code

}

**Indentation:**

1 tab per pair of curly braces

**Variable names:**

private \_variableName

public variableName

**Reports:**

**Matthew:** Feasibility report & Initial System Design

**Andy:**  Project Plan & Initial Object Design

**Read up on:**

Android BlueTooth library

Java Databases

1. **Meeting 5:**

Tuesday 1/30/15

@ 4:00 pm

Conference Call starts

In attendance: Jaime Borras, Andy Martinez, Matthew Santiago, Gummy

## References