|  |
| --- |
| HDCLOUDJAN |
| Requirements Specification (RS) |
| Basketball Coach Pad |

|  |
| --- |
| Firstname Lastname/x15029042  10/10/2016 |

Requirements Specification (RS)

Document Control

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **Version** | **Scope of Activity** | **Prepared** | **Reviewed** | **Approved** |
| 10/10/2005 | 1 | Create | AB | X | X |
| 17/10/05 | 2 | Update | CD |  |  |

Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Title** | **Version** |
| Eugene McLaughlin | Lecturer |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Related Documents

|  |  |
| --- | --- |
| **Title** | **Comments** |
| Title of Use Case Model |  |
| Title of Use Case Description |  |

**Table of Contents**

[1 Introduction 4](#_Toc316977392)

[1.1 Purpose 4](#_Toc316977393)

[1.2 Project Scope 4](#_Toc316977394)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc316977395)

[2 User Requirements Definition 4](#_Toc316977396)

[3 Requirements Specification 4](#_Toc316977397)

[3.1 Functional requirements 4](#_Toc316977398)

[3.1.1 Use Case Diagram 5](#_Toc316977399)

[3.1.2 Requirement 1 <name of requirement in a few words> 5](#_Toc316977400)

[3.1.3 Requirement 2 <name of requirement in a few words> 6](#_Toc316977401)

[3.2 Non-Functional Requirements 7](#_Toc316977402)

[3.2.1 Performance/Response time requirement 8](#_Toc316977403)

[3.2.2 Availability requirement 8](#_Toc316977404)

[3.2.3 Recover requirement 8](#_Toc316977405)

[3.2.4 Robustness requirement 8](#_Toc316977406)

[3.2.5 Security requirement 8](#_Toc316977407)

[3.2.6 Reliability requirement 8](#_Toc316977408)

[3.2.7 Maintainability requirement 8](#_Toc316977409)

[3.2.8 Portability requirement 8](#_Toc316977410)

[3.2.9 Extendibility requirement 8](#_Toc316977411)

[3.2.10 Reusability requirement 8](#_Toc316977412)

[3.2.11 Resource utilization requirement 8](#_Toc316977413)

[4 GUI 8](#_Toc316977414)

[5 System Architecture 8](#_Toc316977415)

[6 System evolution 8](#_Toc316977416)

# Introduction

## Purpose

The purpose of this document is to set out the requirements for the development of “Basketball Coach Pad” mobile iOS application which allows easy and fast prepare for team’s basketball session, where basketball coach can make a quick note, manage players, draw the strategies on the basketball board and use a score board to track the game.

The intended customers are all level basketball coaches who wants to move their trainings to the next organized level, no more paper notes, basketball boards and score boards, everything will be on one single application.

The main idea of this mobile iOS application is provide for the user the free options to make their own notes by typing, editing, deleting their notes, plans, strategies. Every time the application will be closed if will be saved automatically, and on the next startup it will show all results which was stored in application.

## Project Scope

The scope of the project is to develop a user friendly mobile iOS application for basketball coaches that shall help them develop their skills and put their trainings to the next level.

The system should have the option to allow the user set up the own personal training plans, strategies, teams and use a score board how they want to use it or upload the txt file from iCloud to avoid typing from the mobile keyboard. Every time the application will be closed if will be saved automatically, and on the next startup it will show all previous results which was stored in application.

**Motivation for this project**

* The main motivation for this project is to provide real iOS mobile application for all level basketball coaches who needs that help. Another motivation is that my friend who is a basketball coach asked me to develop this mobile application and I promise him to develop it.

**Project objectives**

* The main objective of this project is to create a simple and valuable mobile iOS application for basketball coaches where they can easy to set up everything they need for their training session by themselves.
* Upload this application to Apple’s App Store for a worldwide use.
* After application will be lunched, there will be special feature to leave a feedback and suggest some extra features which they want to see on the next update.
* After the uploading the mobile version, I would like to develop same application for iPad, with even more features and more easy to navigate and manage training sessions.

**Project expectations**

* After lunching the Basketball Coach Pad application, create similar application for football.
* The project has potential to become profitable at later iterations.

**Project suppositions and restrictions**

* It would be required to send the application to basketball coaches for feedback, where they can approve or give suggestions for improvements.
* It’d be highly beneficial to get the quick answer from basketball coaches as application is already in progress.

**Project risks and contingency plans**

* Most important risk is time, not enough time to complicate the application.
* Avoid complicated features.

## Definitions, Acronyms, and Abbreviations

Will be updated later…..

# User Requirements Definition

In this section, there are essential requirements that the user must have to access and use the mobile application.

* **iPhone:** User must have at least iPhone or later.
* **iOS 10:** User should preferably have at latest iOS 10 installed, but application is not limited to iOS10 and will run on iOS 8 and iOS 9.
* **iTunes App Store:** User should have an Apple ID in order to download the application.
* **Internet Access:** User will need Internet access via Wi-Fi or Mobile internet in order to download the application.

# Requirements Specification

This section contains all of the functional requirements of the application and gives main details description of the system and all its features.

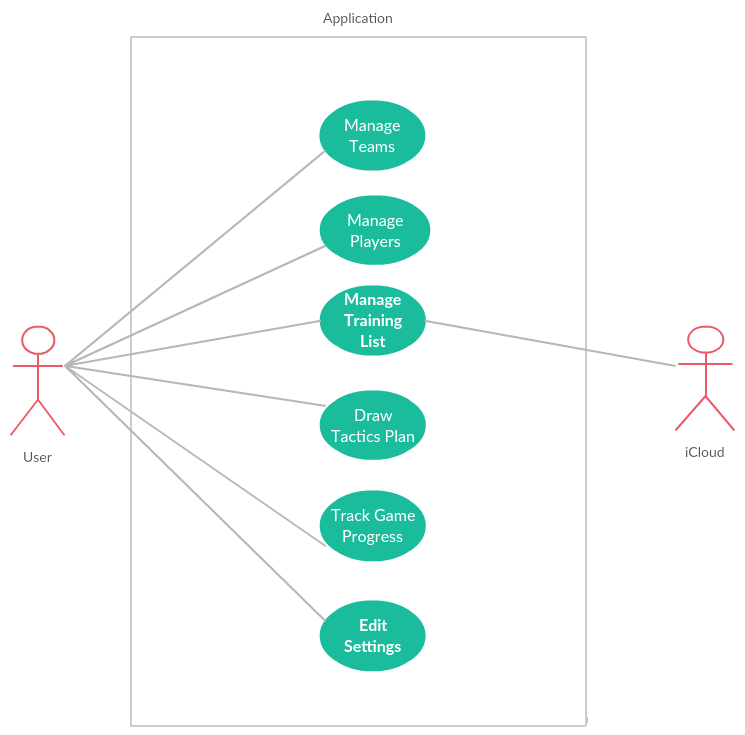
## Functional requirements

This section of functional requirements will identify the statement what the application must do and perform when user will start to use the application.

* When the user will start application, first view will be lunch screen storyboard view with the application main image and copyright information with the developer’s name. Inside the application will be easy navigation with a 5 sets of view controllers, which each of them will represents a tab bar item on the button of the application, which will provide information about specific tab bar and supplies with the view to be displayed when the item is selected.
* Buttons which will be implemented with intercept touch events and sends action to target object when it’s tapped. Buttons will be set with images and other appearance properties of a button which will be easy to understand and press.
* Fast navigate and manage stack of view controllers, each of which will represents information about the view, like title and the navigation item which will associate with the view. When view controller will have pushed back, the navigation controller will update the navigation bar and will give the previous view controller appropriately.
* **Add Team, Players, Training Plan:** The app will allow the user to add teams, players and training plan.
* **Edit Team, Player, Training Plan:** The app will allow the user to change the particular list of teams, players and training plan.
* **Display Team, Players, Training Plan:** The app will display on screen the saved list of teams, players, training plan.
* **Exit Keyboard Mode Via Background Touch on Screen:** The app will allow the user to exit from the keyboard by the click outside the text field.
* **Exit Keyboard Mode Via Done Button:** The app will allow user to exit from the keyboard by pressing done button.
* **Next Line Via Return Button:** The app will allow user to jump to the next line in the text field area.
* **Swipe Left on Table View to Delete:** The app will allow user to swipe left to delete information what is stored in table view.
* **Draw and Erase Mode:** The app will allow user to draw and erase the strategy drawing on provided drawing board.
* **Count Game Score:** The app will allow user to count the game score by adding +1, +2, +3, reset the score and subtract by 1.
* **Game Timer:** The app will allow to user to set game timer with pause, resume and reset mode.

### Use Case Diagram

Figure 1: Use Case Diagram of a System “Basketball Coach Pad”.



The Figure 1 presents basic functionalities and features of “Basketball Coach Pad” system by representation of actor as the main users interactions with the system with use cases.

### Requirement 1: Manage Teams

#### Description & Priority

In the team option section user will open team list. User has the options to add new team, update the existing team or delete it. When team is created user can edit it in the team list.

This use case is high priority use case as this is one of the main features and functions of this application.

#### Use Case

**Scope and Description**

The scope of this use case is to define how team list is managed.

**Flow Description**

**Activation**

This use case starts when “team” tab is selected. “Team” is selected by default when application starts.

**Main flow**

1. Teams list is displayed.
2. User click on “add” button to create new team.
3. User fill title field.
4. When user press “done” button, new team is added to the list.
5. User returns to team list screen.

**Alternate flow 1**

1. Teams list is displayed.
2. User click on “add” button to create new team.
3. User fill title field.
4. When user press “back” button, nothing is stored.
5. User returns to team list screen.

**Alternate flow 2**

1.Teams list is displayed.

2. When user swipe left on the existing team’s list item, team is deleted.

**Alternate flow 3**

1.Teams list is displayed.

2. When user clicks on an existing team, opens editing mode.

**Termination**

This use case is terminated when user clicks on other tab.

### Requirement 2: Manage Players

#### Description & Priority

In the team option section user will open team list. User has the options to add new player, update the existing player or delete it. When player is created user can edit it in the players list.

This use case is high priority use case as this is one of the main features and functions of this application.

#### Use Case

**Scope and Description**

The scope of this use case is to define how players list is managed.

**Flow Description**

**Activation**

This use case starts when “team” tab is selected and picked existing team

**Main flow**

* + - 1. Players list is displayed
      2. User click on “add” button to create new player in the existing team.
      3. User fill specifics field provided by this use case.
      4. When user press “done” button, new player is added to the list existing team which was picked by user.
      5. User returns to players list screen in existing team which is picked.

**Alternate flow 1**

1. Players list is displayed
2. User click on “add” button to create new player in the existing team.
3. User fill specifics field provided by this use case.
4. When user press “back” button, nothing is stored.
5. User returns to existing team’s players list screen.

**Alternate flow 2**

1.Player list is displayed.

2. When user swipe left on the existing player’s list item, player is deleted.

**Alternate flow 3**

1.Player list is displayed.

2. When user clicks on an existing player, opens editing mode.

**Termination**

This use case is terminated when user clicks on other tab.

### Requirement 3: Manage Training List

#### Description & Priority

In this training plan option section user can add training plan. User has the options to add new training plan, update the existing training plan or delete it.

This use case is high priority use case as this is one of the main features and functions of this application.

#### Use Case

**Scope and Description**

The scope of this use case is to manage basketball training sessions by creating training plan or notes, editing and deleting them from the table view scene.

**Flow Description**

**Activation**

This use case starts when user selects the “training” tab.

**Main flow**

1. Training session list is displayed.
2. User clicks on “add” button to create new training plan.
3. User fill title field and text field.
4. When user press “add” button, new training plan is added to the lis
5. User returns to team list screen

**Alternate flow**

1. Training plan list is displayed.
2. User click on “add” button to create new training plan.
3. User fill title field and text field.
4. When user press “back” button, nothing is stored.
5. User can return to team list screen or add another team

**Alternate flow 2**

1. Training list is displayed.
2. When user swipe left on the existing training plan item, training plan is deleted.

**Alternate flow 3**

1. Training plan list is displayed.
2. When user clicks on an existing training plan, opens editing mode.

**Exceptional flow**

1. Training session list is displayed.
2. User clicks on “add” button to create new training plan.
3. When user press “add from iCloud” button, text is copying to text field.
4. When user press “add” button, new training plan is added to the list.
5. User returns to team list screen.

**Termination**

This use case is terminated when user clicks on another tab.

### Requirement 4: Draw Tactics Plan

#### Description & Priority

The scope of this use case is to draw basketball tactics plan on basketball board which is provided on a “strategy” tab. User can draw and erase drawings.

#### Use Case

**Scope and Description**

The scope of this use case is to draw basketball tactics on the basketball board.

**Flow Description**

**Activation**

This use case starts when user selects the “strategy” tab.

**Main flow**

1. User can see basketball tactics board on the strategy scene.
2. User can draw the basketball tactics plan on the basketball board which provides this tab.
3. User can erase some parts or drawings.
4. User can reset all drawings from the basketball tactics board.

**Alternate flow ????**

**Termination**

This use case is terminated when user clicks to another tab.

### Requirement 5: Edit Settings

#### Description & Priority

The scope of this use case is to show settings information about the application. User can check version of application, share the link where another user can download from App Store. User also can send the feedback or suggestion to developer.

#### Use Case

**Scope and Description**

The scope of this use case is to show settings information about the application.

**Flow Description**

**Activation**

This use case starts when user press the “settings” tab.

**Main flow**

1. User can see all settings information what is included in the scene.
2. User can send the feedback to developer.
3. User can share the App Store link to another user to download this application.

**Alternate flow ???**

1. User can see all settings information what is included in the scene.
2. User can share the App Store link to another user to download this application or click on the screen background to cancel the share option

**Termination**

This use case is terminated when user clicks to another tab.

### Requirement 3 Track Game Progress

#### Description & Priority

The scope of this use case is to track game progress by counting the game score and controlling time.

#### Use Case

**Scope and Description**

The scope of this use case is to allow user to track the game score and control the time.

**Flow Description**

**Activation**

This use case starts when user press “game” tab.

**Main flow**

1. User can check pregame score and time.
2. User can count home and guest scores by adding 1, 2, 3 and subtracting 1.
3. User can refresh all home and guest score result to 0.
4. User can set up the game time, make a pause, resume and refresh the time.

**Alternate flow ???**

**Termination**

This use case is terminated when user clicks to another tab.

## Non-Functional Requirements

Specifies any other particular non-functional attributes required by the system. Examples are provided below. **Remove the requirement headings that are not appropriate to your project.**

### Performance/Response time requirement

* The application performance and response time should depend on iPhone’s model, the latest model the fastest performance.
* The application will be built on iOS 10 operating system, using iOS 8 and iOS 9 shouldn’t cause any problem with performance and response.

### Availability requirement

The application shall be available in the Apple App Store before November 30.

### Recover requirement

For application backup shall save the project files daily.

### Robustness requirement

The application has to be able to cope with the errors and bugs during the executing process, but not the developing process.

### Reliability requirement

When the application will be uploaded to App Store it will be available for 100% till any specific bug or update.

### Maintainability requirement

It should be easy to modify the code and adapt or update the application.

### Portability requirement

It will be only available for iPhone, but in nearest future it will be design for iPad’s as well.

### Extendibility requirement

Same application will be developed for Android phones.

### Reusability requirement

In the application where Table View will be used, it will be reused with reusable cells, to minimize the application’s cache memory.

### Resource utilization requirement

# GUI

Include mock-ups of the key pages or stages of the system. Explain how they are linked. Explain how you addressed above requirements in the design. It is important that the mock-ups are in line with the functional requirements above, e.g., if one of your requirements is “user registration” then one of the screens listed in this section should show a registration page.

# System Architecture

Use a class diagram to outline the structure of the system. Explain briefly why you have chosen this architecture. You might want to use Visio or Rational Rose to create these.

# System Evolution

This section describes how the system could evolve over time.