

CPASS

(Cricket Player Performance System & Pakistan Army Sports Selection System)



By

Capt Muhammad Imran

Capt Usama Javed

Capt Faisal Iqbal

Capt Adnan Hashmi

Supervisor

Major Sohaib Khan

Submitted to Faculty of Department of Computer Software Engineering National University of Sciences and Technology, Islamabad in partial fulfilment for the requirements of a B.E Degree in Computer Software Engineering July 2020

ABSTRACT

The sports are considered to be an integral part of a healthy life. Not only they contribute to physical health, they have a great impact on mental health as well. In this growing world sports are played not only for fun, but people are in it as professionals who represent their organizations, states and countries on national and international level. Where winning or losing is a fundamental principle of sports, winning factor has a lot of impact on the player, organization, state and nation. CPASS is an application which has been designed keeping in mind the very factor of winning. This application uses principles of statistics and artificial intelligence to predict the possibility of outcome of a match based on information available for the players and other statistics. This application uses an unsupervised algorithm to predict the performance of the player based on his previous performances in similar circumstances.

will help the cricket selection committee to select the best Player on the basis of their performance. They can make the list of best performing player for the current match. They can get all players performance information on simple clicks which saves their time and cost for selecting Players. On the basis of selected players of the team we can also predict the winning and losing Chances it helps the selection committee to foresee that this selected team can win us the match or what the chances of winning and losing.

CERTIFICATE FOR CORRECTNESS AND APPROVAL

It is certified that work contained in the thesis CPASS (Cricket Player Performance System & Pakistan Army Sports Selection System) carried out by Capt Muhammad Imran, Capt Usama Javed, Capt Faisal Iqbal and Capt Adnan Hashmi under supervision of Maj Sohaib Khan for partial fulfilment of Degree of Bachelor of Software Engineering is correct and approved.

Approved By
Major Sohaib Khan
Department of IS, MCS

Dated:

DECLARATION

No portion of the work presented in this dissertation has been submitted in support of another award or qualification either at this institution or elsewhere.

DEDICATION

To our families especially our parents, without their support and cooperation, a work of this magnitude would not have been possible.

To our supervisor, Major Sohaib khan who has given us great support and valuable suggestions throughout the journey of this project.

ACKNOWLEDGEMENTS

There is no success without the will of ALLAH Almighty. We are grateful to ALLAH, who has given us guidance, strength and enabled us to accomplish this task. Whatever we have achieved, we owe it to Him, in totality.

We are also grateful to our parents and family and well-wishers for their admirable support and their critical reviews.

We are extremely grateful to our project supervisor Major Sohaib Khan, for his continuous guidance and motivation throughout the course of our project. Without his help, we would have not been able to accomplish this feat.

We are highly thankful to all the faculty and staff of Computer Software Department of Military College of Signals, NUST for their support and training throughout our coursework. Their guidance helped us to carry out this project.

In the end we would like to acknowledge the efforts of all our friends, colleagues and well-wishers whose prayers and faith helped us in achieving our goal

Contents

Chapter 1: Introduction	1
1.1 Background	1
1.2 Motivation and challenge	1
1.3 Goals and Objectives	2
1.4 Methodology	2
Chapter 2: Literature Review	3
2.1 Problems and Existing Systems	3
2.2 What is Difference	5
2.3 Summary	5
Chapter 3: Software Requirements Specification	6
3.1 Introduction	6
3.2 Purpose	6
3.3 Definitions, Acronyms, Abbreviations	6
3.4 Project Scope	7
3.5 References	7
3.6 Overall Description	7
3.7 Users and Characteristics	9
3.8 Operating Environment	9
3.9 Design and Implementation Constraints	10
3.10 Developing	10
3.11 Dependencies	10
3.12 External Interface Requirements	10
3.13 System Features	11

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

3.14 Other Non-Functional Requirements	12
Chapter 4: Software Design Specification	13
4.1 Introduction.....	13
4.2 System Architecture Description	16
4.3 Detailed Description of Components.....	35
4.4 Reuse and Relationships to Other Products	39
4.5 Design Decision and Tradeoffs.....	39
Chapter 5: Implementation	41
5.1 Introduction.....	41
5.2 Predicting Algorithm	41
5.3 Pseudo Code.....	41
Chapter 6: Testing.....	46
6.1 Introduction.....	46
6.2 Test Cases	46
6.3 Summary	53
Chapter 7: Conclusion and outlook.....	54
7.1 Introduction.....	54
7.2 Achievements and Improvements.....	54
7.3 Critical Review	54
7.4 Future Recommendation/Outlook.....	55
7.5 Summary	55
Chapter 8: Bibliography.....	56

LIST OF FIGURES

Figure 2.1 Existing app GUI	04
Figure 2.2 Cricket Predictor Website	04
Figure 4.1 Work Breakdown Structure	15
Figure 4.2 Block Diagram	16
Figure 4.3 Component Diagram	17
Figure 4.4 Use Case Diagram	18
Figure 4.5 Entity Relationship Diagram	25
Figure 4.6 Class Diagram	26
Figure 4.7 Login Sequence Diagram	27
Figure 4.8 Add Data Sequence Diagram	28
Figure 4.9 Team prediction Sequence Diagram	29
Figure 4.10 Player prediction sequence Diagram	30
Figure 4.11 Previous Result Sequence diagram	31
Figure 4.12 Logical View Diagram	31
Figure 4.13 Activity Diagram	32
Figure 4.14 Class Diagram of CPASS.....	33
Figure 4.15 Structure Chart	35
Figure 5.1 Application Launch GUI	42
Figure 5.2 Sign up & Sign In	43
Figure 5.3 Match Prediction I	43
Figure 5.4 Match Prediction II	44
Figure 5.5 Match Prediction III	44
Figure 5.6 Player Prediction	45

Chapter 1: Introduction

Introduction

Cricket Prediction App (CPASS) has the goal to predict winning team and also predict the best performing players for the match in international or domestic cricket. Cricket prediction app (CPASS) will be use worldwide in different cricket clubs or cricket board but the purpose of the CPASS will remain the same in every club or board. CPASS will help the under dogs of the cricket which will never find the chance to show best cricket in domestic or international. CPASS analyzes and predicts the match outcome and performance of the player before the match against any team.

1.1 Background

Human brain can predict but it also has some feelings, so sometimes it takes one sided decision. From day to day person makes big decisions about multiple things that he/she encounters in his life. These decisions depend upon multiple previous record and prediction on them. If somebody got the prefect prediction result on the previous record, it will help person more than anything in his life.

With the emergence of cricket in whole world and people has craze to watch the cricket, everybody in Asia is too much crazy about the cricket. So we come with an idea to facilitate the cricket enthusiast and build application which will predict the player or match outcome before the match. To full fill that purpose about facilitate the cricket enthusiast search the different predicting algorithm [1].

1.2 Motivation and challenge

The main challenging on working in CPASS was [2]

- Data analysis techniques
- Data Analysis
- Predicting Algorithm
- Static Analysis
- Clearing Data

1.3 Goals and Objectives

We will build an automate system using Artificial intelligent and will produce the result on the machine intelligence and using some type of algorithm of AI. The key goals which CPASS will achieve

- Generate the database of all the player of teams
- Applying the predicting formula
- Develop the android application
- Connecting the android application with the web server
- Updating the data of the player
- Getting the current date of the match which will be predicted

1.4 Methodology

Traditionally, research in the field of AI is very common nowadays. Most of the people are working on the predication in different fields. The cricket is the most popular game in world nowadays everybody wants to know winner and the looser of the cricket match. CPASS is one of the cricket predication applications which will tell the user about player performance. We choose the android application because it is most common OS in Smartphone more than 50% of people use android Smartphone in the world.

our aim to develop the android application which will send the data of the selected player, against team name and ground where the match will be held to the web server for database we will use MySQL for the previous record of the player. On previous record it will find the result using the compound formula through which it generates the signal value of a player. We will produce the player predication on that value.

In this project we need an android phone, internet access and the web database.

Chapter 2: Literature Review

Introduction

The following chapter we will discuss about the concept upon which the application will be developed. The chapter contains a detailed discussion on the definition of teams and details about the cricket prediction systems. The chapter will discuss the baselines on which the application will be built and points that under consideration while the development is in progress. In this chapter we will do a comprehensive literature review that will help us understand what cricket prediction systems are and how these systems are developed and maintained. All the concepts about the cricket prediction system will be enlisted and discussed as per requirement of the project.

2.1 Problems and Existing Systems

There are some applications that are available in the play store that related to the cricket prediction, but they are not reliable. These applications are not using any type of algorithm for prediction. Most of the applications are using the user voting or rating taken by the user to predict the match outcome.

2.2.1 Cricket predictor Application

Feature:

- Android base prediction
- Open source application
- Simple user interface

Problem:

- Don't use any type of predicting algorithm for prediction
- Don't have any type of previous record
- Doing the prediction on the base of user voting
- Don't predicting the player performance
- Don't predicting the match outcome before the match [3].



Figure 2.1: Existing App GUI

2.2.2 Cricket Prediction Website

Feature:

- Producing prediction
- Success of prediction ratio 89%
- Very simple website to use
- Good interface

Problem:

- Don't using any type of predicting algorithm for prediction
- Don't tell about the algorithm for prediction
- Predicted result send by SMS
- Produce the prediction on bases of User feedback [4]

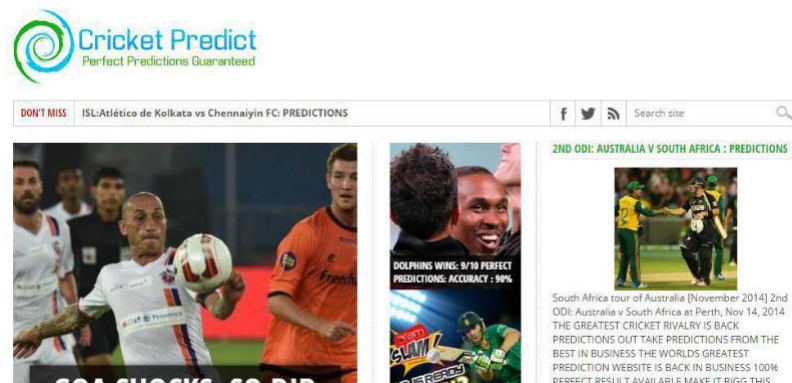


Figure 2.2: Cricket Predictor Website

2.2 What is Difference

In contrast to the problems discussed in the previously developed cricket predication system the main advantage which CPASS will use the predicting algorithm for prediction and it will be android application.

2.2.1 Technologies Which CPASS Uses

CPASS is using different type of technologies some them are uses in existing systems but some are not yet.

- Android operating system
- Predicting Algorithm

2.3 Summary

This chapter gives an overview of a brief market survey & Technologies overview of existing cricket predication systems and what is the limitation of these systems according to the predications.

We will see some more information of requirements gathering of CPASS and why there was the need of such application and what were the core requirements which made us and inspired us to develop such application.

Chapter 3: Software Requirements Specification

3.1 Introduction

Cricket Player Performance and Pak Army Sports Selection System (CPASS) will automate the entire selection procedure of team based on their performance records. Our System will analyze the player performance based on previous records and predict the performance of a certain player for the upcoming tournaments/ series and will suggest the best selection of team based on previous performances. The system is also going to predict the outcome of match.

CPASS can be implemented in different places like GHQ Sports Dte, PCB, local clubs, leagues and Committees which not only help the end user but also help the player to increase its performance in match against team in a specific ground. There are many different prediction applications that provide predictions for player, but these predictions are specific on the user votes. This means that the predictions of a player in these predictions' applications will not use any type of predicting algorithm. These predictions will change because of user point of view related to the player. However, CPASS will predict using AI Algorithms.

3.2 Purpose

CPASS will automate the whole selection system of cricket team and will help the selection committee. Moreover, it will also help the Coach and Captain in selection of playing XI by predicting the specific record of a player in a certain ground and against certain team. CPASS will help in fair selection, helping the under dogs and will eliminate the chances of human errors.

3.3 Definitions, Acronyms, Abbreviations

Abbreviation	Complete
CPASS	Cricket Player Performance and Pak Army Selection System
MCS CSE Dept	MilitaryCollege of Signals' Computer Software Engineering Department
DG	Director General

PCB	Pakistan Cricket Board
Dte	Directorate
Div	Divisions
Bde	Brigades

Table 3.1: Definitions & Abbreviations

3.4 Project Scope

The Cricket Player Performance and Pak Army Sports Selection System (CPASS) application will be developed to predict the result using the Big Data Analysis and Static Analysis. The Application consists of following few things.

- Predict player performance
- Predict match Outcome
- Previous Results

3.5 References

- GHQ Sports Dte
- www.CricInfo.com
- PCB
- Corps/ Div teams

3.6 Overall Description

3.6.1 CPASS Perspective

This CPASS can be useful in following perspective:

- For selection of Army, Corps, Division and Brigade teams
- Can further be used in PSL, Domestic League and PCB.

3.6.2 Functionalities

3.6.2.1 Functional Requirements

3.6.2.1.1 Player Performance

3.6.2.1.1.1 Select Team

In this requirement the user will select the player name from the list.

3.6.2.1.1.2 Select Player

In this requirement the user will select the team of a player that he/she want to predict.

3.6.2.1.1.3 Select against Team

In this requirement the user selects the team from the list that he/she want to prediction against that team.

3.6.2.1.1.4 Select Ground

In this requirement the user will select the ground from the list where the match will played.

3.6.2.1.2 Team Prediction

3.6.2.1.2.1 Select Team 1

In this requirement the user will select the team 1 from the list.

3.6.2.1.2.2 Select Team 2

In this requirement the user will select the team 2(against team) from the list.

3.6.2.1.2.3 Select Team ground

In this requirement the user will select the ground from the list where the match will take place.

3.6.2.2 Non Functional Requirements

3.6.2.2.1 Performance Requirements

3.6.2.2.1.1 Response Time

Response time will be approximate 2minute when an action is performed based on internet performance.

3.6.2.2.1.2 Startup Time

Startup time will be less than 1.5 minute based on the internet Performance.

3.6.2.2.1.3 Shutdown Time

Shutdown time will be less than 5 seconds based on the device performance.

3.6.2.2.2 Graphical User Interface Requirements

User Interface will be consistent between the modules.

3.6.2.2.3 Portability

The minimum recommended android device hardware for the system should be Android version minimum 3.0 versions 512 ram.

- RAM: 512MB or Greater
- Internet Connection

3.6.2.2.4 Availability

The system shall be available depending upon device battery and availability of the internet.

3.6.2.2.5 Usability

The system shall provide user friendly interface so that user is able to use the system. He / She should have basic knowledge of using the android phone.

- Easy to understand
- System must be able to display data with all desired actions
- Password protected

3.7 Users and Characteristics

- User can belong to any type of field but must know how to use Android phone and internet.
- Initial user will be the Offrs/Soldiers of GHQ Sports Dte.

3.8 Operating Environment

The software and languages used are mentioned below:

- Java/Python Platform
- Android Studio

- Rational Rose
- Adobe Photoshop

The Hardware required are:

- Laptop PC
- Android Mobile

3.9 Design and Implementation Constraints

The intent of this project is to predict the match outcome as well as player performance. The design includes:

- Login page
- Player Performance Prediction Page
- Match Prediction Page
- Previous Results Page

3.10 Developing

CPASS is a prediction application. The basic and most important functionality is prediction based on previous data through machine learning algorithms.

3.11 Dependencies

The minimum recommended android device hardware for the system should be Android version minimum 3.0 versions 512 RAM.

- RAM: 512MB or Greater
- Internet Connection

3.12 External Interface Requirements

3.12.1 User Interfaces

- Front end Software:
 - Android Studio

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

- Back end software: ○
PHP/ MySQL

3.12.2 Hardware Interfaces

- Android
- Browser

3.12.3 Software Interfaces

Following software will be used in this project

- Android studio
- PHP / MySQL
- Adobe Photoshop
- Rational Rose

3.13 System Features

3.13.1 CPASS Version1Features

Version 1 of CPASS is based on two main modules. First module is player performance. In this module user can predict the player performance against any team in a specific ground. Second module is match prediction. In this module user can predict the outcome of the match before the match on basis of ground and player of both teams.

3.13.2 Sub CPASS version 1 Features

Sub CPASS version1 is based on single module i.e. previous results. In this module user can see the list of previous matches prediction record those which were predicted by CPASS.

3.14 Other Non-Functional Requirements

3.14.1 Performance Requirements

- User must have Android mobile available
- The minimum recommended android device hardware for the system should be Android version minimum 3.0 versions 512 ram.
 - RAM: 512MB or Greater
 - Internet Connection

3.14.2 Safety and Security Requirements

- CPASS when specifically use for Pakistan Army Team Selection at different level, security is important consideration to cater for
- Only authorize access to be available when dealing with Pak Army Team selection.

3.14.3 Software Quality Attribute

- **Availability:** The data on the different players from different countries/corps/divisions should be available for prediction.
- **Correctness:** The accurate and up-to-data should be available regarding the teams, players and grounds.
- **Maintainability:** The administrator should maintain the latest information on database.
- **Usability:** The CPASS should satisfy the maximum number of user's needs. It should be visually appealing and easy for the user to navigate.

Chapter 4: Software Design Specification

4.1 Introduction

Cricket Player Performance and Pak Army Sports Selection System (CPASS) being developed helps the cricket fan to know about the winner or loser of a coming match. It will predict the match outcome on the base of previous record of team. It not only predicting the match outcome CPASS will also predict the player performance.

The chapter will cover the system designs from architectural level to the detailed level diagrams of use case of the system. The Cricket Player Performance and Pak Army Sports Selection System (CPASS) is an android based system so it will be placed on Google play store. The user will access the system using the google play store which will provide the user all the available features of the application. The application will be able to predict the match outcome and player performance. The user will interact with UI layer which consist of multiple component that are discussed later in the chapter. The UI layer will pass the requests to the web-server layer which will serve as DB layer and provide all necessary data for prediction. Then the requested data pass through the predicting algorithm and produces the result on the UI layer.

4.1.1 Purpose

This software design specification (SDS) document describes the architecture and system design of CPASS. It features different design diagrams and their explanation. The document addresses stakeholders of the details of the design and its creation process. This document will help the developer(s) in implementation and maintenance of the Application (app).

4.1.2 Scope

The Cricket Player Performance and Pak Army Sports Selection System (CPASS) application will be developed to predict the result using the Big Data Analysis and Static Analysis. The Application consists of following few things.

- Predict player performance
- Predict match Outcome
- Previous Results

4.1.3 Definitions, Acronyms, Abbreviations

Abbreviation	Complete
CPASS	Cricket Player Performance and Pak Army Selection System
MCS CSE Dept	Military College of Signals' Computer Software Engineering Department
DG	Director General
PCB	Pakistan Cricket Board
Dte	Directorate
Div	Divisions
Bde	Brigades

Table 4.1: Definitions & Abbreviations

4.1.5 References:

- GHQ Sports Dte
- www.CricInfo.com
- PCB
- Corps/ Div teams

4.1.6 Document Overview

This document shows the design and working of RAD. It starts from higher level details for a non-technical reader to understand just by seeing the diagrams to the lower level details that aid the developer to code and understand other technical details of the application. Section **System Architecture Description** gives a detailed overview of the application. Section **Structure and Relationships** shows the higher level details system working by the means of System Block, Activity, State Transition, and Use Case diagrams. Lower level details are described using the Class, Sequence diagrams and Structure Chart. Section describes how the application is designed to curb the tendency of **User Interface Issues** and problems during User

Interaction. In Section **Detailed Description of Component** it is given to show the working of modules with low level details. It shows the purpose, function, subordinates, dependencies, interfaces, resources, processing and data of the components and their relationships with each other.

Section **Reuse and Relationship to other Products** shows information about work done in the same project before and any reuse of the same work. The section also provides a key to reuse this system for further upgrades. Section **Design Decisions and Tradeoffs** shows the architecture style and design pattern of the application, while in the Section 6 the **Pseudo Code** of the components is given in for human reading rather than machine reading.

4.1.7 Work Breakdown Structure

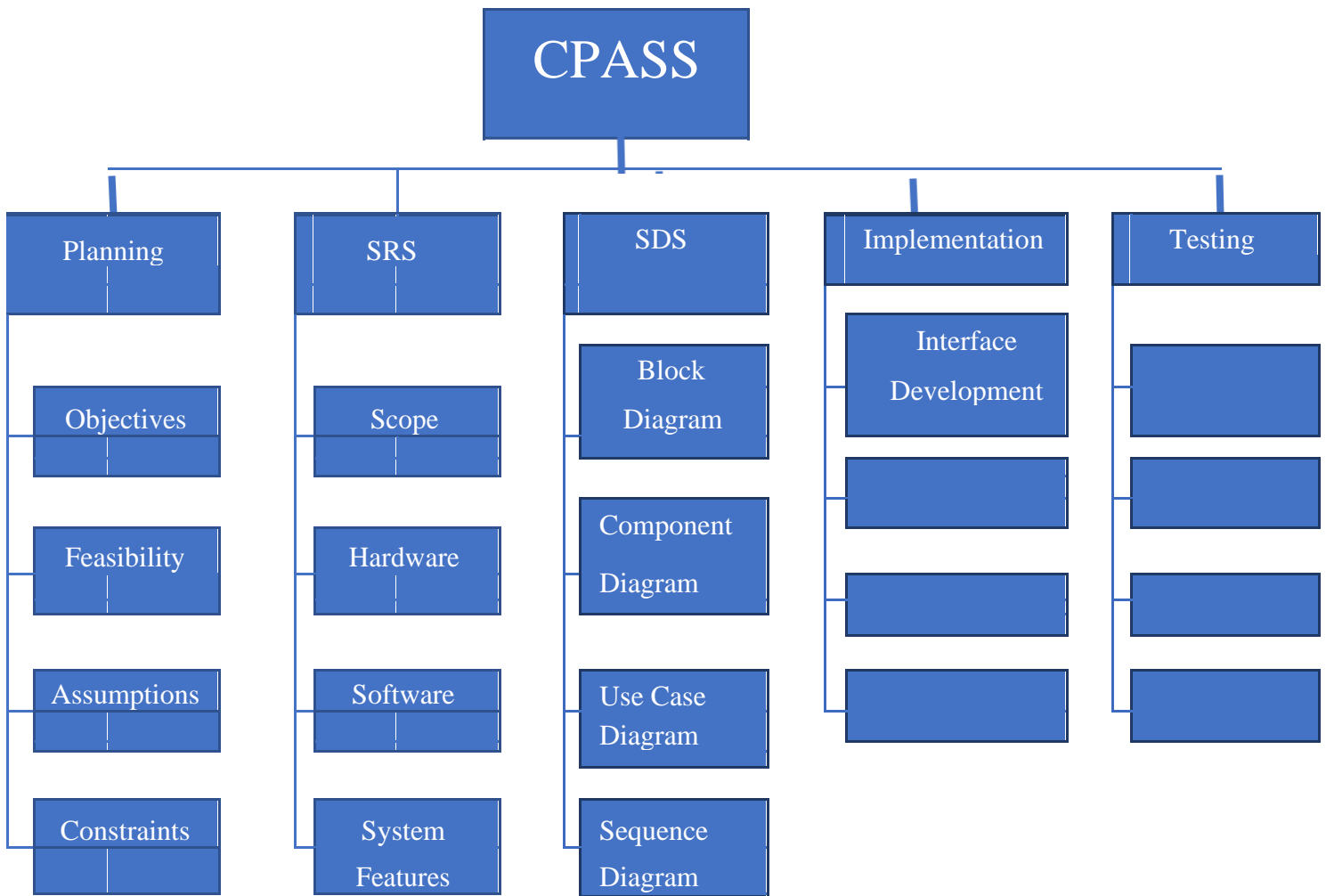


Figure 4.1: Work Breakdown Structure

4.2 System Architecture Description

Detailed description of system architecture and design pattern which this system is going to use is discussed later in the document in section 5 „Design Decisions and Tradeoffs“. This Section gives overview of application, its higher and lower levels details and user interfaces.

4.2.1 Architecture and Relationships

This section covers the overall technical description of RAD. It shows the working of application in perspective of different point-of-views and also shows relationships between different components.

4.2.1.1 System Block Diagram

The diagram(s) show the higher level description of the application(s), generic working of the application(s) and interaction with the user. User inputs its requirements through UI layer (smartphones). AI prediction algorithm reads user input and process/predict the outcome based on the statistical data in the database and displays it to user.

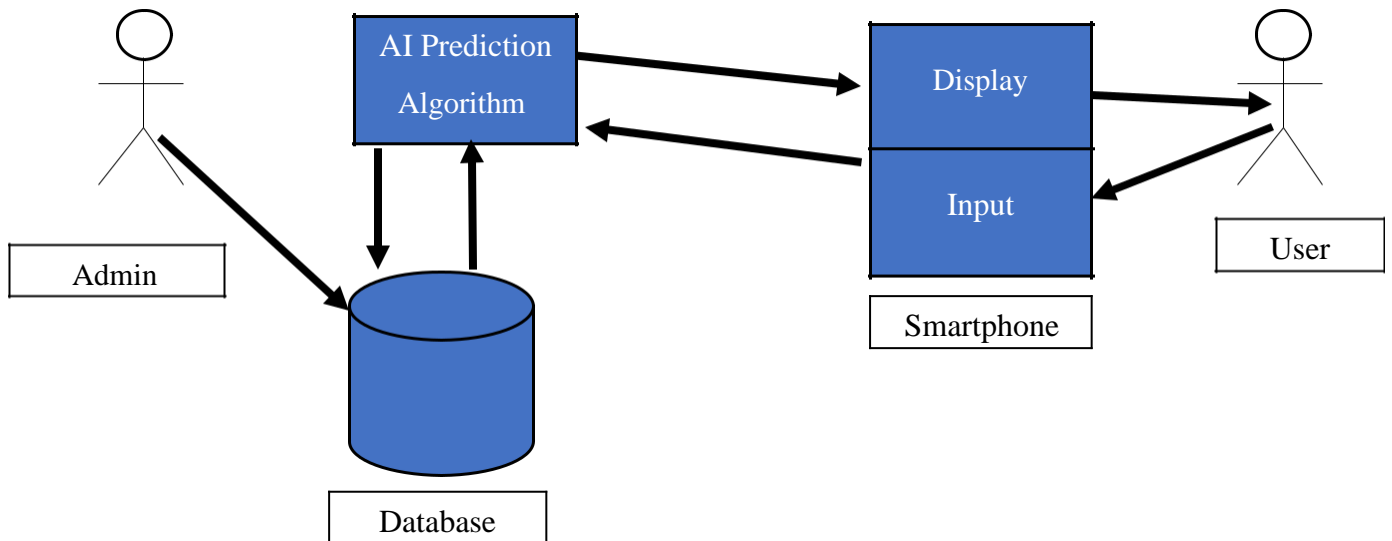


Figure 4.2: Block Diagram

4.2.1.2 Component Diagram

The main components are

- User Input
- GUI
- Unity Engine

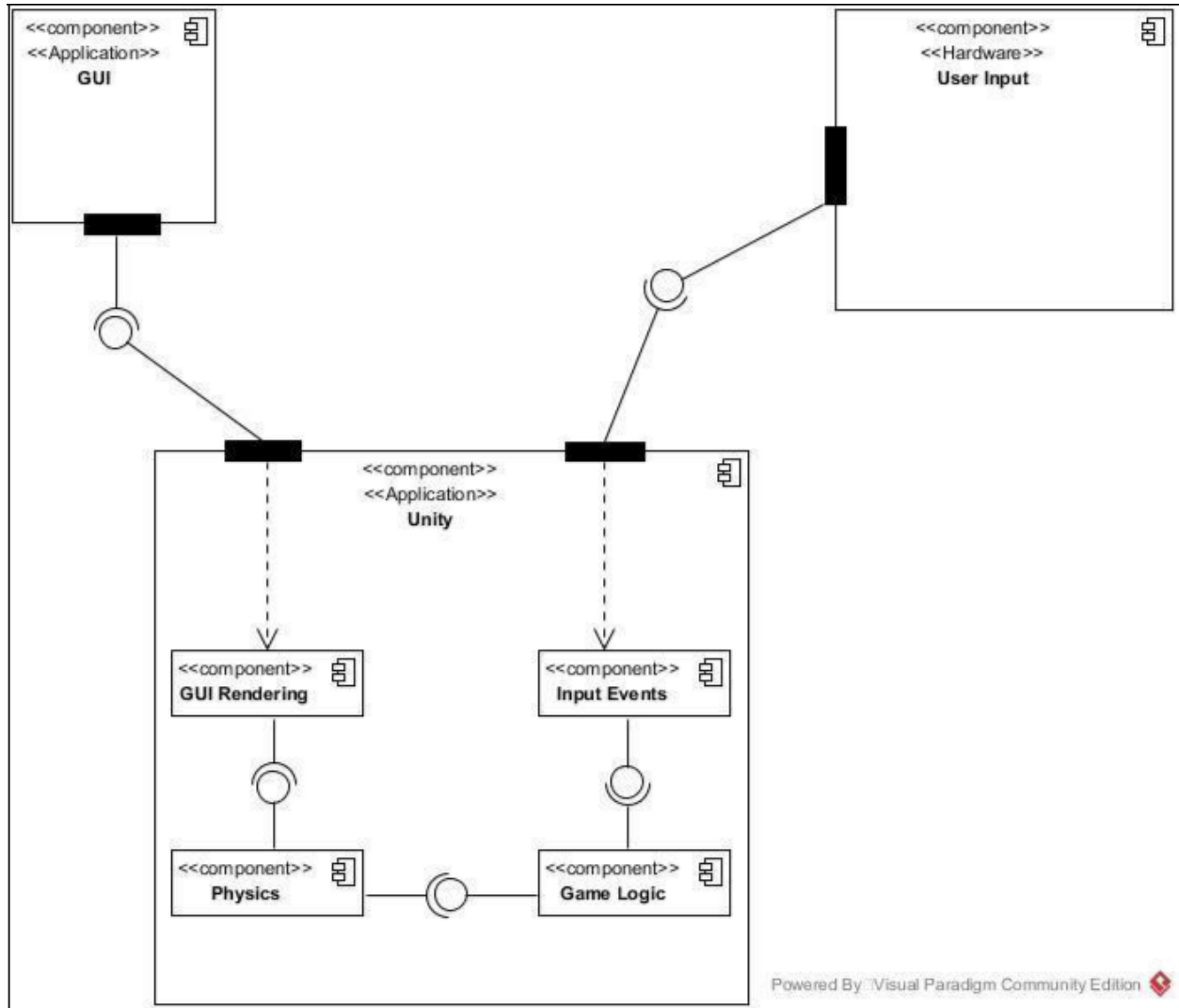


Figure 4.3: Component Diagram

4.2.1.3 User View (Use Case Diagram)

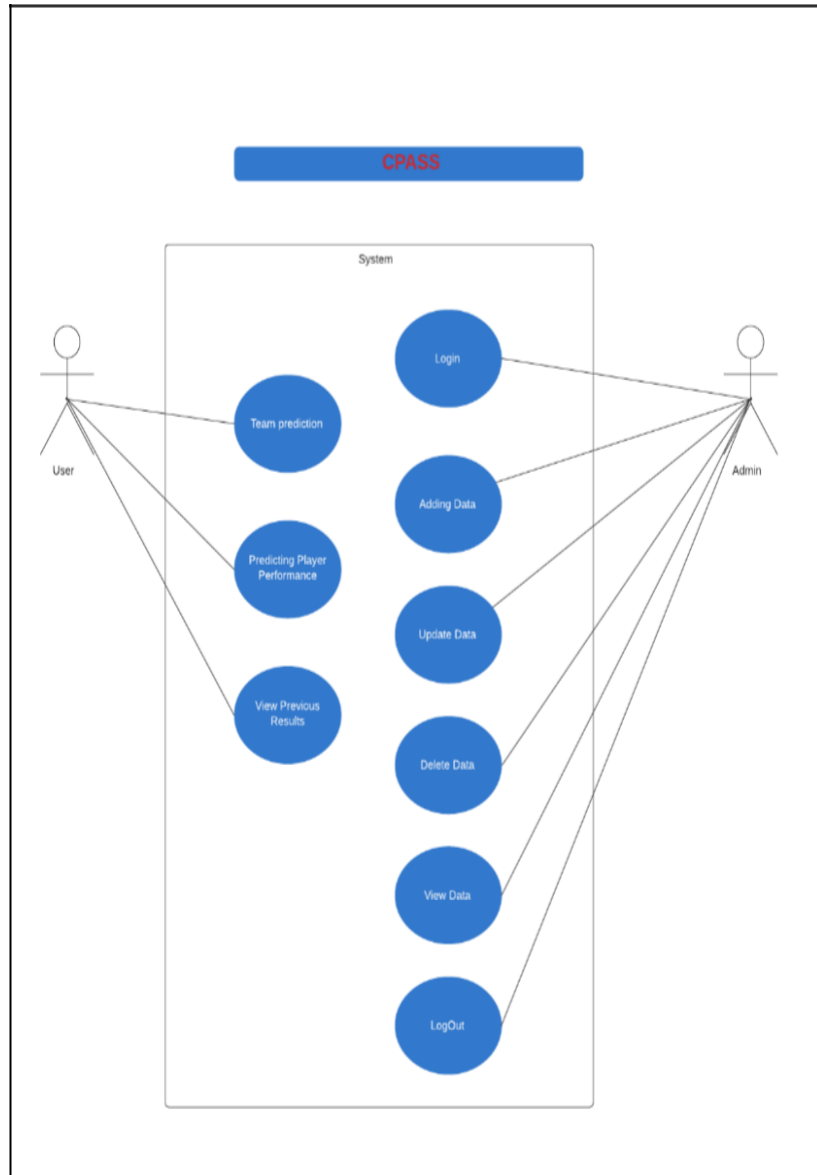


Figure 4.4: Use Case Diagram

Actors:

Primary Actors:

Users

Secondary Actors:

Admin

4.2.1.3.1 Use Case 1: Team Prediction

Use Case ID:	UC-2.0
Use Case Name:	Team prediction
Actors:	User
Relationship:	Association
Description:	User wants to predict team result.
Pre-conditions:	User should download and run the application Device must be connected with internet
Post-conditions:	System shows team prediction
Basic Flow:	<p>System shows the team prediction screen to the user User perform some task</p> <p>User selects team 1 from the drop down menu</p> <p>User selects team 2 from the drop down menu</p> <p>User selects ground from the drop down menu</p> <p>User clicks on “Next” button System shows team 1 Player list</p> <p>User choose eleven players of team 1 from check list</p> <p>User clicks on “Next” button</p> <p>System shows team 2(against team) players list</p> <p>User choose eleven players of team 2(against team) from check list</p> <p>User clicks on “Predict Result” button System shows Predicted Result on the screen</p> <p>User can clicks on team 1 detail System shows result of team 1 players details</p> <p>User press back button.</p> <p>System shows Predicted Result screen again</p> <p>User can clicks on team 2 detail System shows result of team 2 players details</p>

Alternative Flows:	If User press home button. A: System show home window. Else if User press back button B: System show home window Else (User press exit button) C: System show home window.
--------------------	--

Table 4.2: Team Prediction

4.2.1.3.2 Use Case 2: Player Performance Record

Use Case ID:	UC-2.1
Use Case Name:	Player performance
Actors:	User
Relationship:	Association
Description:	User view result of selected teams
Pre-conditions:	User should download and run the application Device must be connected with the internet
Post-conditions:	System shows the player performance prediction
Basic Flow:	User press player performance button System shows player performance window User selects team from drop down menu User selects Player from drop down menu User selects Against Team from drop down menu User selects Ground from drop down menu User clicks on “Predict Player Performance” System shows Result of Player Performance
Alternative Flows:	If User press home button. A: System show home window. Else if(User press back button) B: System show home window Else (User press Exit button) C: System show home window.

Table 4.3: Player Performance Record

4.2.1.3.3 Use Case 3: View Previous Result

Use Case ID:	UC-2.2
Use Case Name:	View Previous Result
Actors:	User
Relationship:	Association
Description:	User view Previous result.
Pre-conditions:	User should download and run the application Device must be connected with internet
Post-conditions:	User can View the previous result
Basic Flow:	User Press Previous Result Button System shows Previous Result. User can view the previous result
Alternative Flows:	If User press home button. A: System show home window. Else if(User press back button) B: System show home window Else (User press Exit button) C: System show home window.

Table 4.4: View Previous Result

4.2.1.3.4 Use Case 4: Administrator Side Use Case

Login:

Use Case ID:	UC-1
Use Case Name:	Log-in
Actors:	Administrator
Relationship:	Association
Description:	User Login to system as an administrator

Preconditions:	Admin have must registered on website
Post conditions:	Admin login successfully
Basic Flow:	Log-in Admin gets Logged-in Admin Enter username Admin Enter password User press “login” button System Check user is register or not If(user register)
	System show “Log in Successful” System start section
Alternative Flows:	If Admin enter wrong account authentication information System will ask to Re-enter account authentication information

Table 4.5: Login

Managing Data

Use Case ID:	UC-1.1
Use Case Name:	Managing data
Actors:	Administrator
Relationship:	-
Description:	Administrator
Pre-conditions:	Admin should be login.
Post-conditions:	Admin Enter data of a player in a database.

Basic Flow:	<p>System show player data entry window. Admin enter basics information Admin select team name from drop down menu Admin Enter player name Admin select player role from drop down menu Admin Enter match information</p> <p>Admin Select Against team From drop down menu Admin Enter date or select from drop down menu Admin select match type from drop down menu Admin enter match inning Admin select match timing from drop down menu Admin select match result from drop down menu Admin Enter the ground information</p> <p>Admin Enter ground name Admin Enter Country name(where ground is located) Admin Enter City name(Where ground is located) Admin enter player statistic</p> <p>Admin enter total run of player in a match Admin enter number of faced balls by the player Admin enter strike rate of a player Admin select status from drop down menu Admin enter number of wickets taken by player in a match Admin enter number of over's Admin enter run conceded</p>
	Admin enter economy of a player
Alternative Flows:	Alt1. Admin enter wrong account authentication information System will ask to Re-enter account authentication information

Table 4.6: Managing Data

Log-out

Use Case ID:	UC-1.2
Use Case Name:	Log-out
Actors:	Administrator

Relationship:	Association
Description:	User Logout from system as an administrator
Preconditions:	Admin Should be login
Post conditions:	System will destroy the session
Basic Flow:	<p>Logout:</p> <p>Administrator press Logout button. System check the user is log-in or not If(user is login system will log-out) Administrator Logged-out from System System Destroy Session.</p>
Alternative Flows:	Alt 1. Admin enter wrong account authentication information System will ask to Re-enter account authentication information

Table 4.7 : Log-out

4.2.1.4 Entity Relationship Diagram

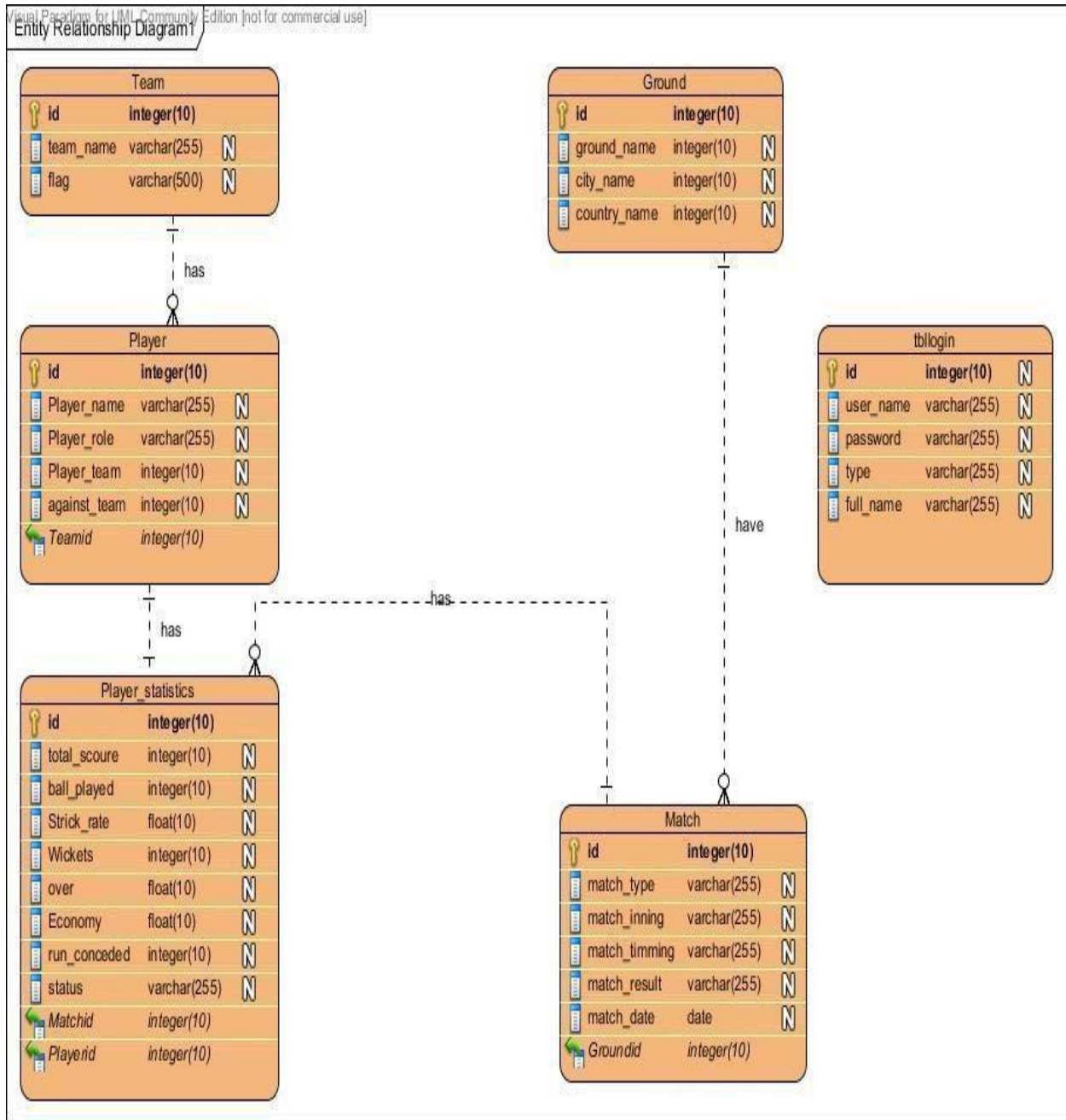


Figure 4.5: Entity Relationship Diagram

4.2.1.5 Class Diagram

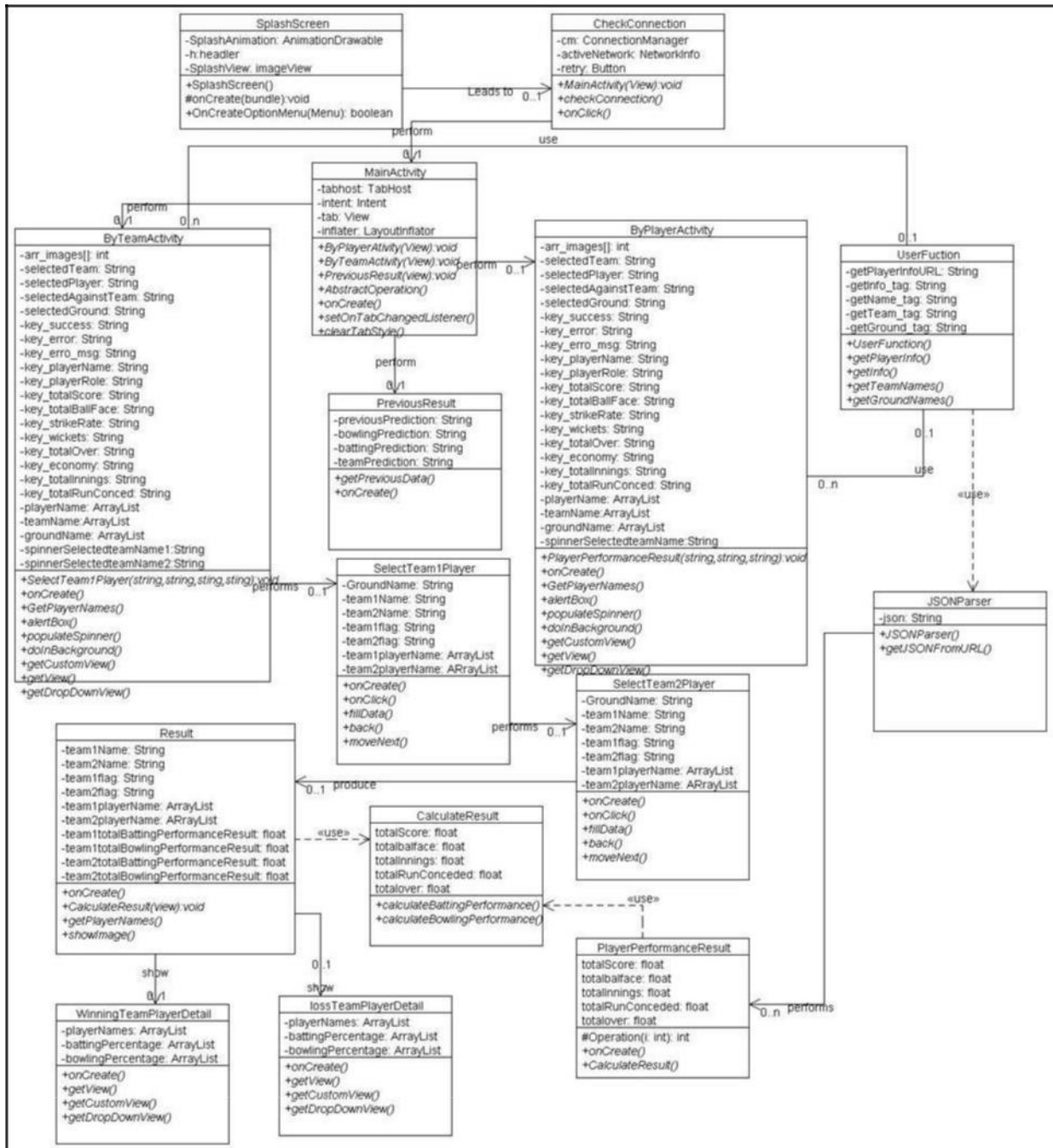


Figure 4.6: Class Diagram

4.2.1.6 System Sequence Diagram

4.2.1.6.1 Login

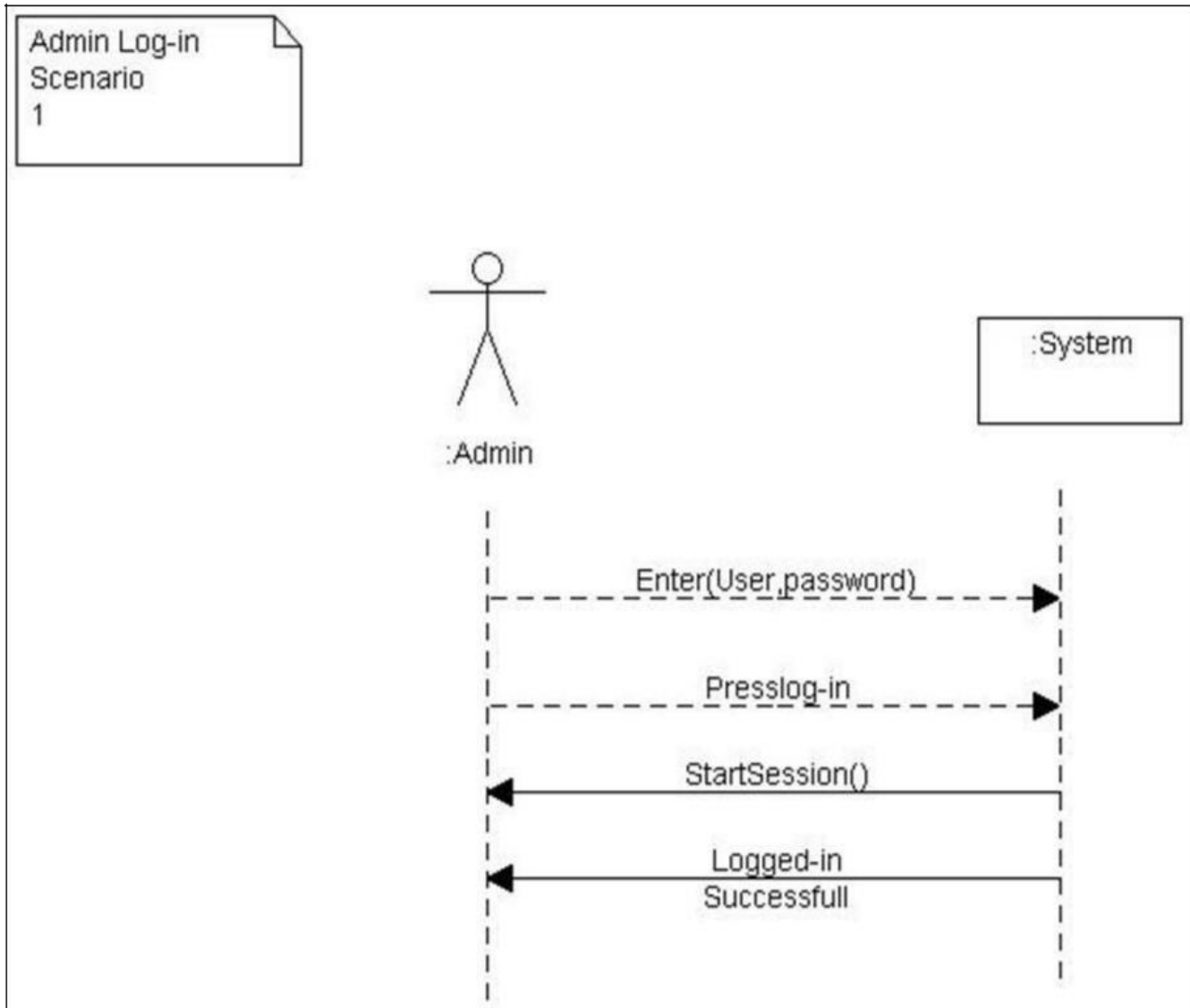


Figure 4.7: Login Sequence Diagram

4.2.1.6.2 Add Data

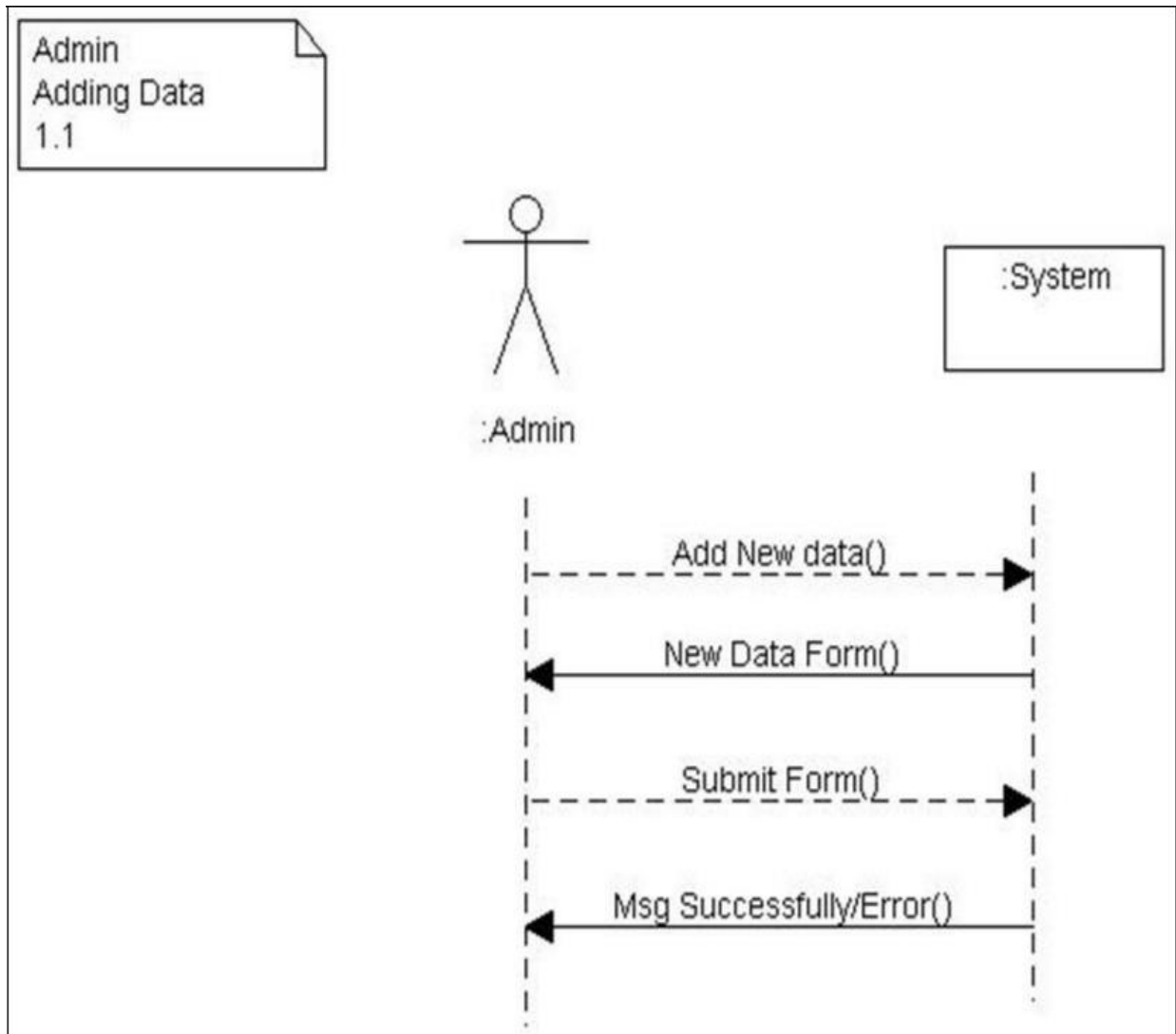


Figure 4.8: Add Data Sequence Diagram

4.2.1.6.3 Team Prediction

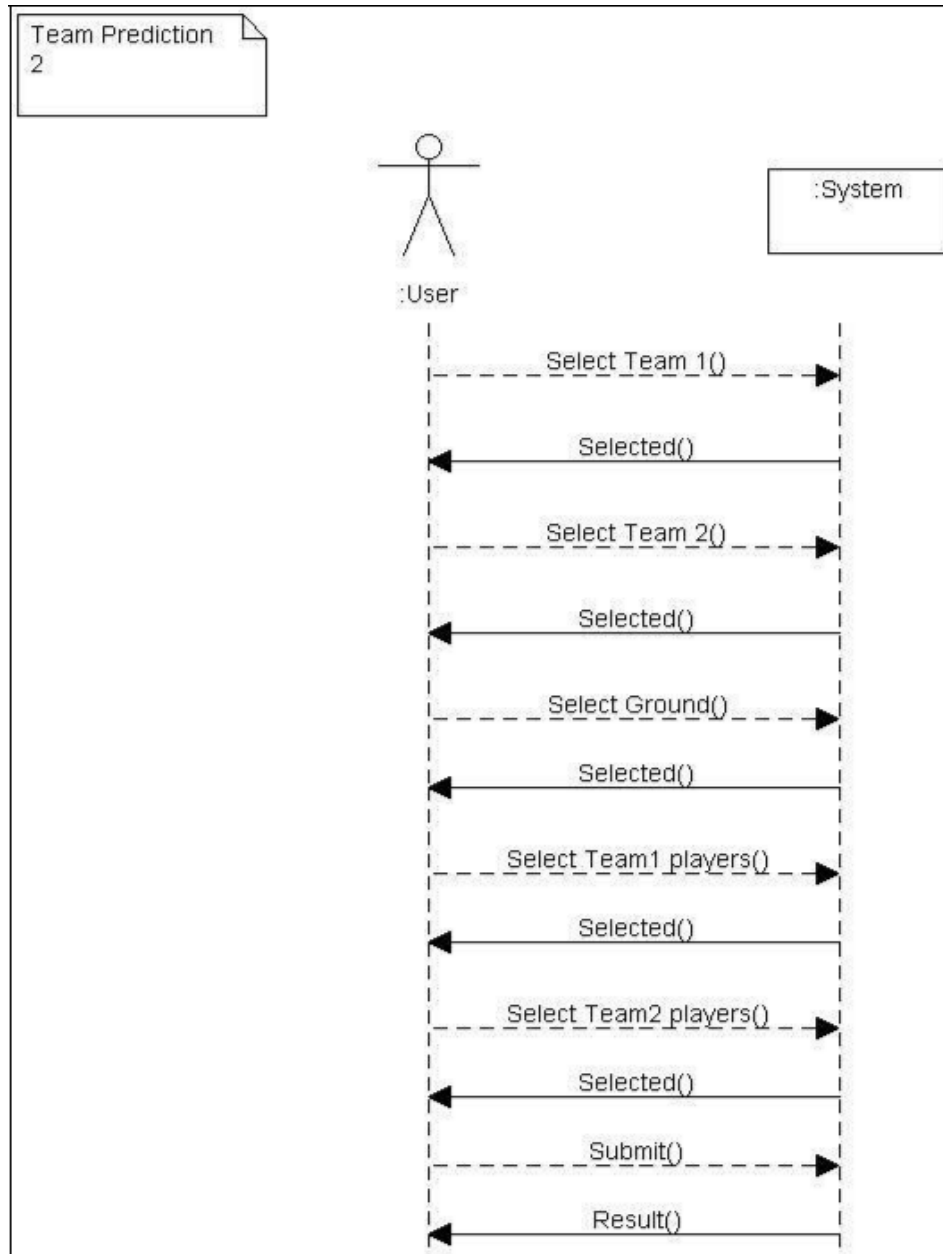


Figure 4.9: Team Prediction Sequence Diagram

4.2.1.6.4 Player Performance

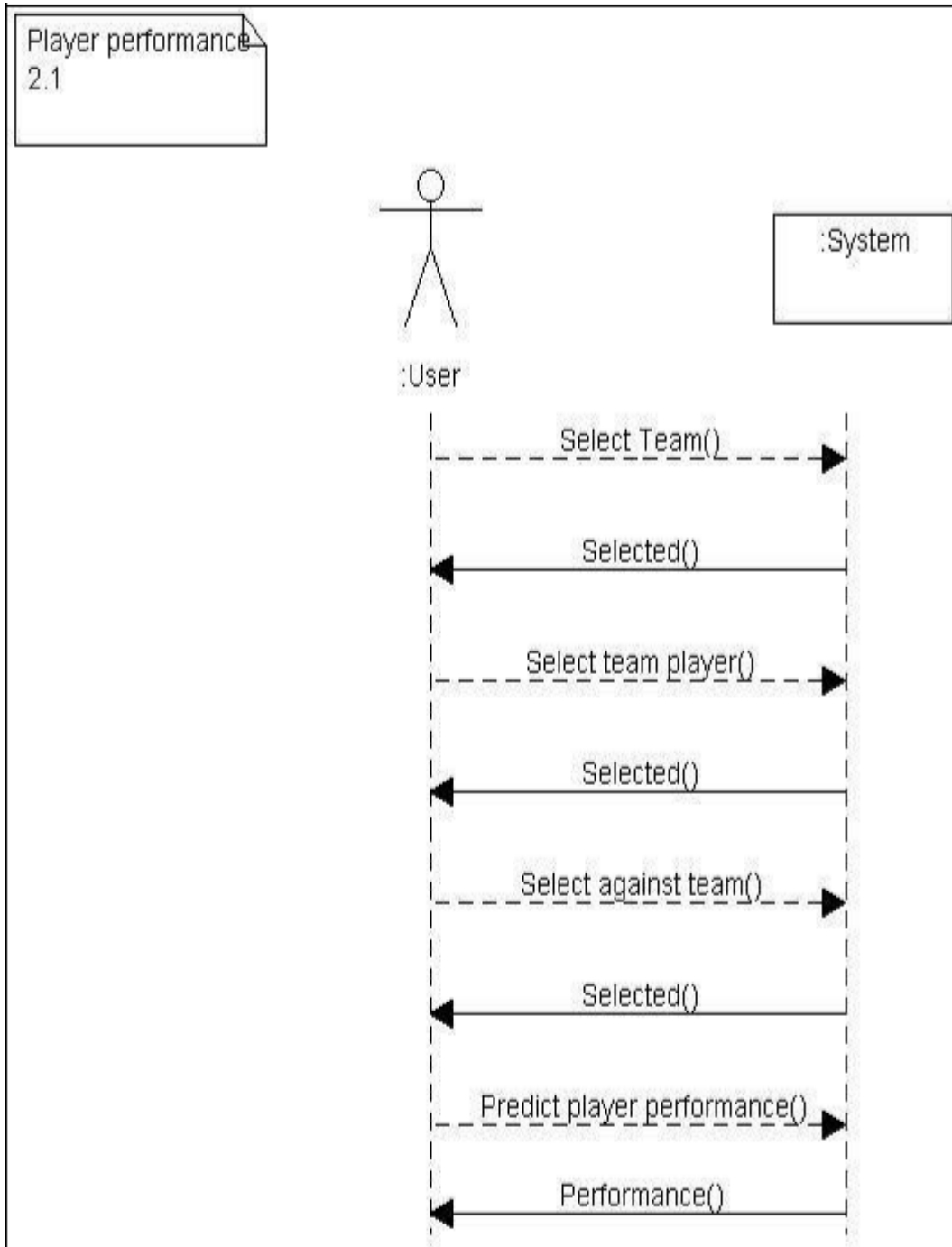


Figure 4.10: Player Prediction Sequence Diagram

4.2.1.6.5 Previous Result

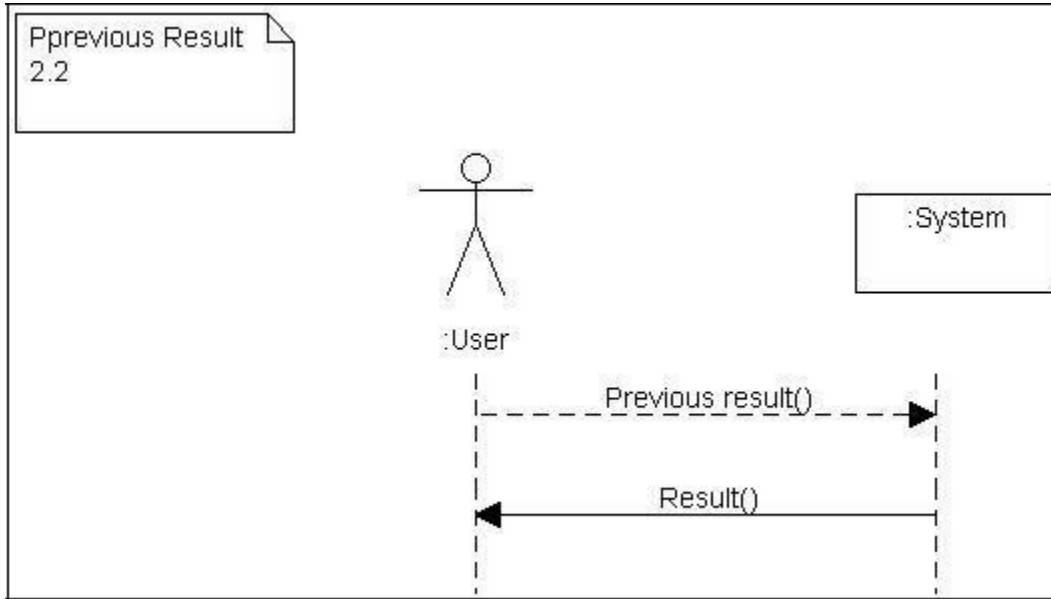


Figure 4.11: Previous Result Sequence Diagram

4.2.1.7 Logical View (State Transition Diagram)

The State Transition diagram is shown in the figure below:

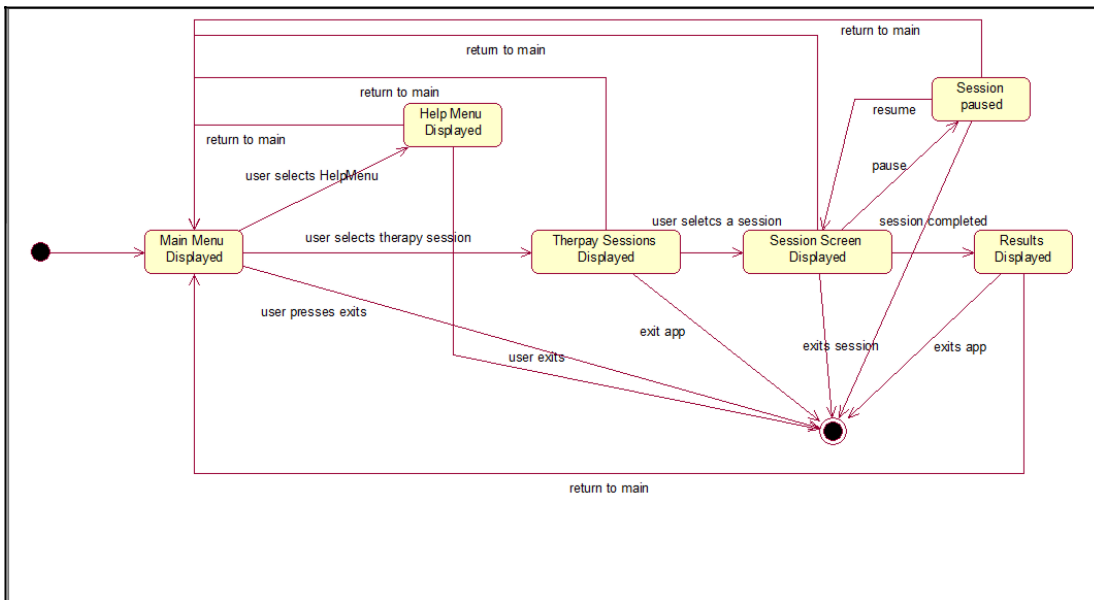


Figure 4.12: Logical View Diagram

4.2.1.8 Dynamic View (Activity Diagram)

In activity diagram, the dynamic view of the system is shown. All the activities are shown concurrently with their respective start and end state

Fig

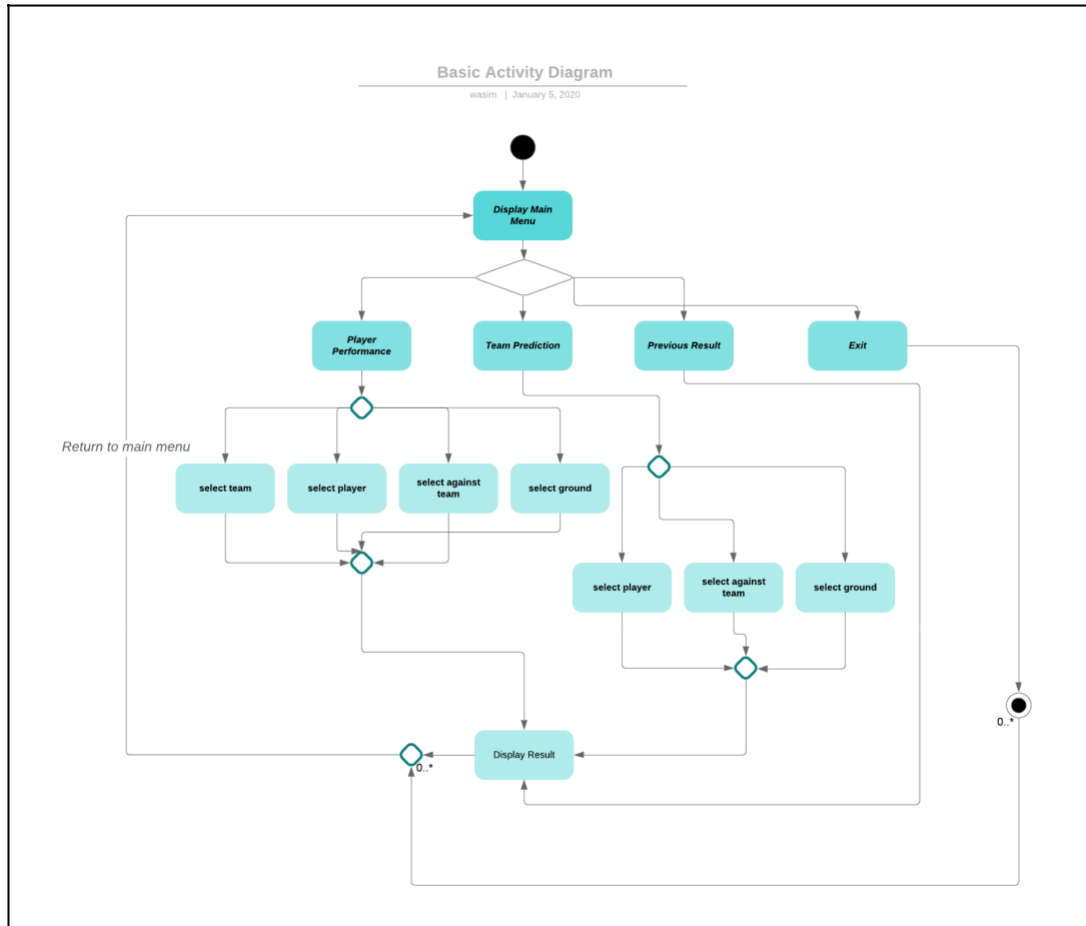


Figure 4.13: Activity Diagram of CPASS

4.2.1.9 Implementation View (Class Diagram)

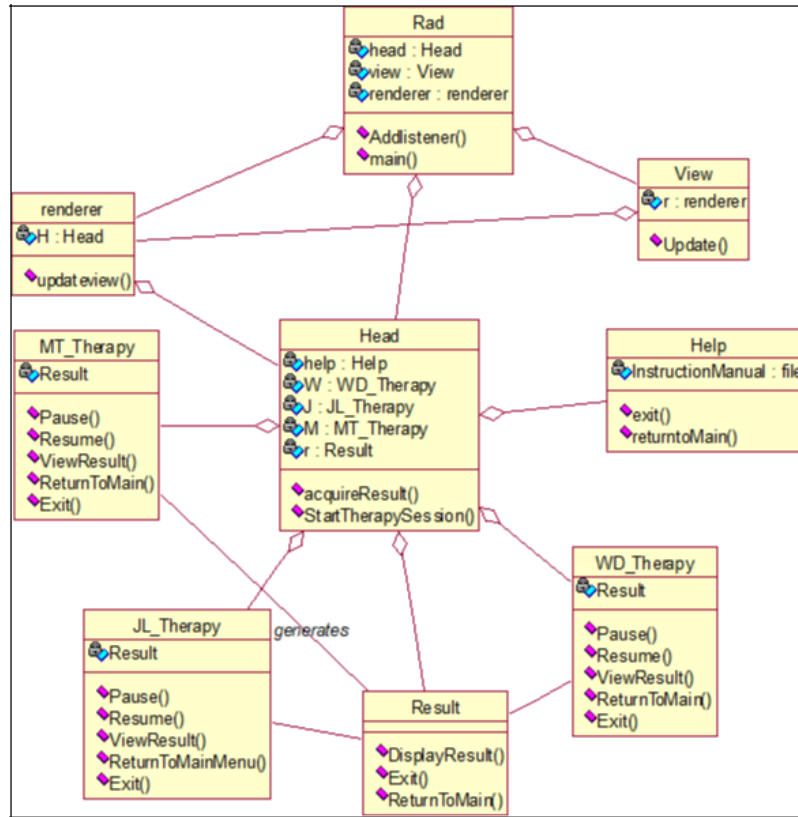


Figure 4.14: Class Diagram of CPASS

4.2.1.9.1 Class Description

Name	Description
Rad	This is main class of the System. It creates the objects of View, Controller and Head classes. It implements the MVC design pattern.
Renderer	This is the Controller class here is performing the MVC's Controller class functionality. It get actions from view and tell model to act accordingly. It invoke the events by making function calls to different methods in brain class of model at update or start actions.

View	This class plays the role of View class of MVC and generate view.
Head	Head plays the main role of MVC's Model class. All the events are generated through it's functions. It contain all the model classes objects to generate events and all the data that is required to generate results and therapy sessions
Help	Displays the instruction manual to the user.
MT_Therapy	Maths Test. This class contains all the functions that deal with the animation, GUI and physics components of this particular therapy. A mathematical exercise with simple arithmetic questions to help patients with right-left direction confusion.
JL_Therapy	Jumbled Letters Therapy. This class contains all the functions that deal with the animation, GUI and physics components of this particular therapy. It contains jumbled up letters game to help patients memorize spellings by interacting with them virtually which they find difficult and boring to learn otherwise.
WD_Therapy	Words Dictionary. This class contains all the functions that deal with the animation, GUI and physics components of this particular therapy. Words dictionary kind of game with difficult words that dyslexic patients normally find hard to grasp and their meanings to help them memorize with repetitive exercises.
Result	This class is responsible for displaying the result of the therapy. It allows user to replay the game or go back to the main menu.

Table 4.8: Class Description

4.2.1.10 Structure Chart

This chart shows the breakdown of the application to its lowest manageable levels. It shows the modules and their corresponding functions which this application will implement. This chart basically shows the structure breakdown of the application starting from main modules to specific functions.



Figure 4.15: Structured Chart

4.3 Detailed Description of Components

4.3.1 User Input

Identification	Name: User input Location: View
Type	Component
Purpose	The purpose of this component is to receive user input. The input through controller will be processed by this component and will be further sent to unity game engine. The user head gestures and hand gestures will be received by this component.

Function	What the component does: Detect input in the form of phone orientation, head movement and hand movement.
Subordinates	Constituents of the component: The component has no sub-components. Functional Requirements: Requirement 1: The user will be able to view the play environment in 360 degrees by looking around. Requirement 2: The user will be able to simulate the movement as per the movement of the phone.
Dependencies	Components using this component: The sub component input events of the unity component will get its input from this module. This input will be further processed by input events component.
Interfaces	The external hardware interfaces interacting with this component will be: HMD Controller/ Default Controller Error messages: Controller Not Found.
Resources	The resources used by this component are HMD Controller (for head gestures inputs) and Default Controller (for hand movement).
Processing	The processing required for this component is receiving the user's input and giving this input to the input event module of the unity game engine
Data	Player inputs (head gestures, hand movement).

Table 4.9: User Input Description

4.3.2: GUI

Identification	Name: GUI (Graphical User Interface) Location: View
Type	Component
Purpose	To display the application.

Function	<p>What the component does:</p> <p>The complete application environment is displayed by this component.</p>
Subordinates	<p>Constituents of the component:</p> <p>GUI basically contains all the game objects.</p> <p>Functional Requirements:</p> <p>Requirement 1: The user will be able to see a play area after selection from the menu.</p>
Dependencies	<p>GUI is dependent on unity engine. The rendering component of the unity will be providing input to the GUI.</p>
Interfaces	<p>External interface requirement for GUI</p> <p>An output display screen which it will be using to display all the objects created by the unity engine and all the menu and app mode screen.</p>
Resources	<p>It will be using graphics engine of the system as per unity requirements.</p> <p>Android smartphone screen</p>
Processing	<p>The processing required of this component is to respond and display results of player input during the game and also during navigation.</p>
Data	<p>User input (hand gestures, head gestures)</p>

Table 4.10: GUI Description

4.3.3: Unity Engine

Identification	<p>Name: Unity engine</p> <p>Location: Model</p>
Type	<p>Module</p>
Purpose	<p>Unity is a game development ecosystem. It is a powerful rendering</p>

	<p>engine fully integrated with complete set of intuitive tools and rapid workflows to create interactive 3D and 2D content. Unity built in physics engine provides component that handles physics simulation with just a few parameters setting. Physics can be controlled from scripts</p>
Function	<p>What the component does:</p> <p>The main functions of this module are as follows:</p> <ul style="list-style-type: none"> Handle game physics Game play rendering Manage game resources on system Manage all other major components of game
Subordinates	<p>Constituents of the component:</p> <p>The other components using this component are:</p> <ul style="list-style-type: none"> GUI Player Input <p>Functional requirements:</p> <p>Requirement 1: The application allows the player to pause game.</p>
Dependencies	<p>Components using this component:</p> <ul style="list-style-type: none"> Player input controls GUI rendering
Interfaces	<p>The external interfaces interacting with unity game engine are</p> <ul style="list-style-type: none"> Processing Head gestures and Hand movements to get Input Speakers
Resources	<p>The resources used by this module are RAM</p> <ul style="list-style-type: none"> Graphic memory

	CPU usage Controller
Processing	The processing done by this module is that it: <ul style="list-style-type: none"> • Initiates game, game objects and all required global variables • Allocates required system resources for game • Manages memory requirements
Data	Float values, integer values, strings

Table 4.11: Utility Engine Description

4.4 Reuse and Relationships to Other Products

RAD is not based on any previous systems neither it's an extension of any other applications at any level. But it can be evolved into a bigger and more complex system with more features and functionality. Developers can also reuse some of the modules of the system. The application can also be enhanced to further include more activities such as a database can be maintained to help user keep a record of his performance throughout and see if any there is any improvement i.e. were the sessions effective? It can also be further enhanced by developing an augmented reality version of the application to make it more immersive.

4.5 Design Decision and Tradeoffs

RAD is an interactive application which requires multiple types of user interface. Developing such systems require thorough consideration on the design factors as it might result in complexity problem. A poorly-designed system results in a system consuming more resources with very little efficiency and a slower response time which directly affects the experience of the target user besides this, poor designs make testing and maintenance activities difficult. MVC pattern will be used for the implementation of this application.

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

Pseudo Code 1.Start 2.Connect to the MySQL Database 3.Create the Database Connection 4.If(predict Team prediction) 1. Send generated queries to the Database 2. Retrieve value from the Database 3. Apply the Predicting Algorithm 4. Show the result 5.If(Predict Player Performance) 1. Send generated queries to the Database 2. Retrieve value from the Database 3. Apply the Predicting Algorithm 4. Show the result

Chapter 5: Implementation

5.1 Introduction

In the following chapter we discuss the work flow and other implementation details that would be considered during the development of the project. The work flow will describe how the queries will be passed down from user interface to the database when algorithm will be implemented and how the values will be shown back to user.

5.2 Predicting Algorithm

For prediction we build the predicting algorithm which will use the predicting formula for to player performance and predict match outcome. Formula as follow

- Predicting formula for batsman = strike rate/number of innings
- Predicting formula for bowler = economy rate/ number of innings

5.3 Pseudo Code

1.Start

2.Connect to the MySQL Database

3.Create the Database Connection

4.If (predict Team prediction)

1. Send generated queries to the Database
2. Retrieve value from the Database
3. Apply the Predicting Algorithm
4. Show the result

1. Send generated queries to the Database
2. Retrieve value from the Database
3. Apply the Predicting Algorithm
4. Show the result

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

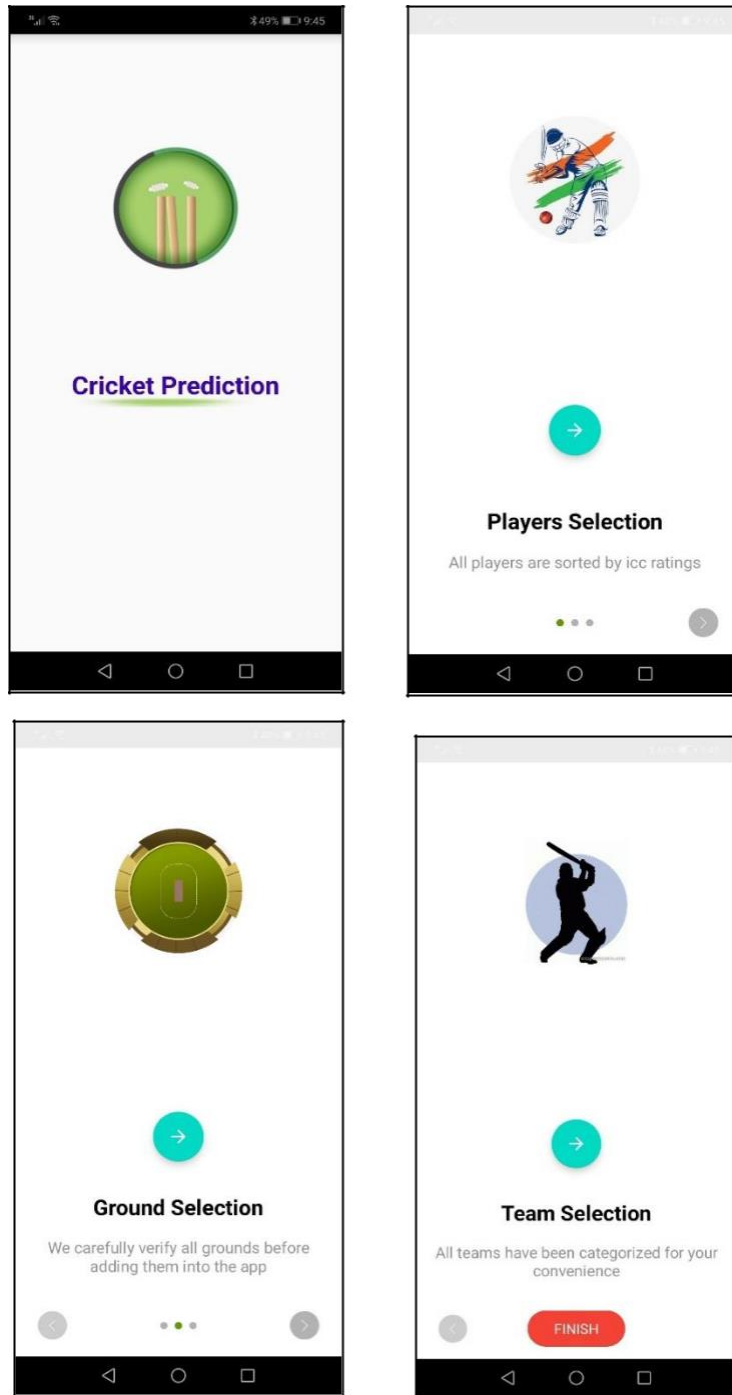


Figure 5.1: Application Launch GUI

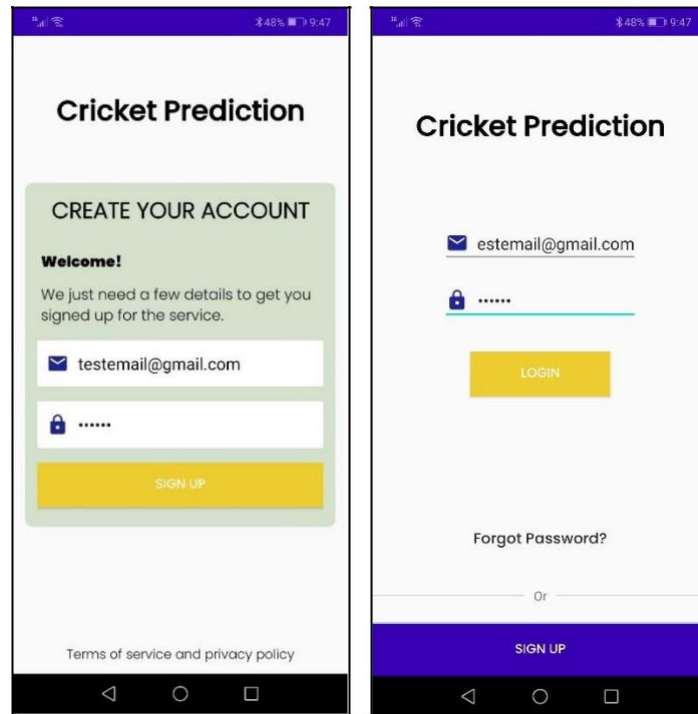


Figure 5.2: Sign Up and Sign In

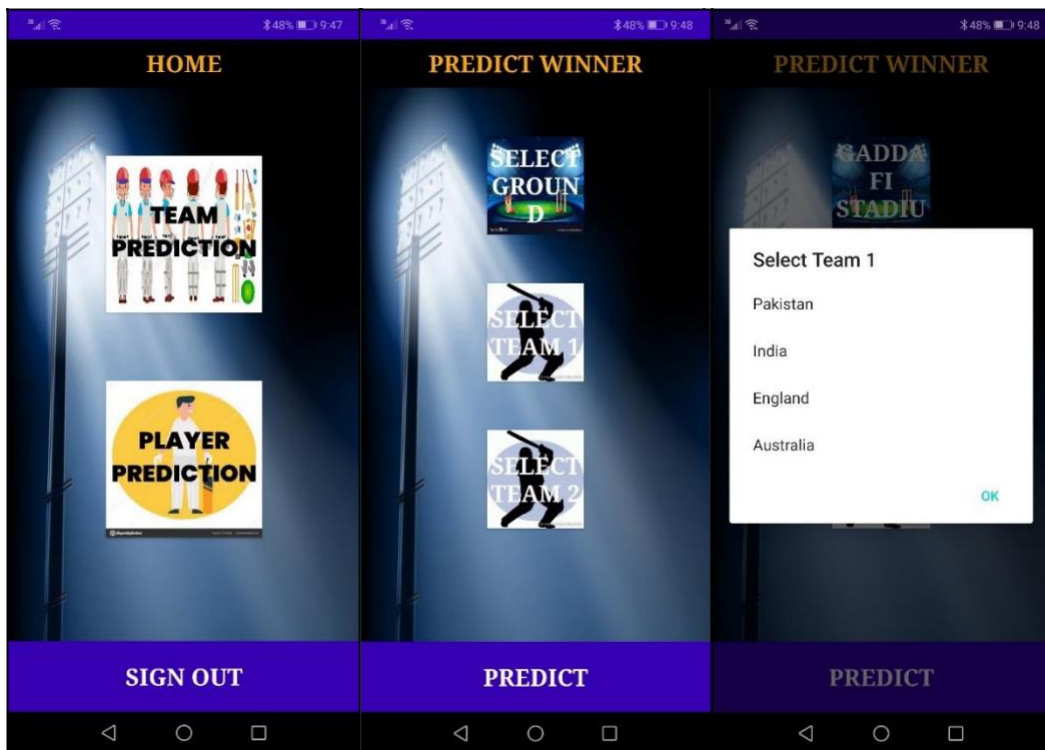


Figure 5.3: Match Prediction I

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

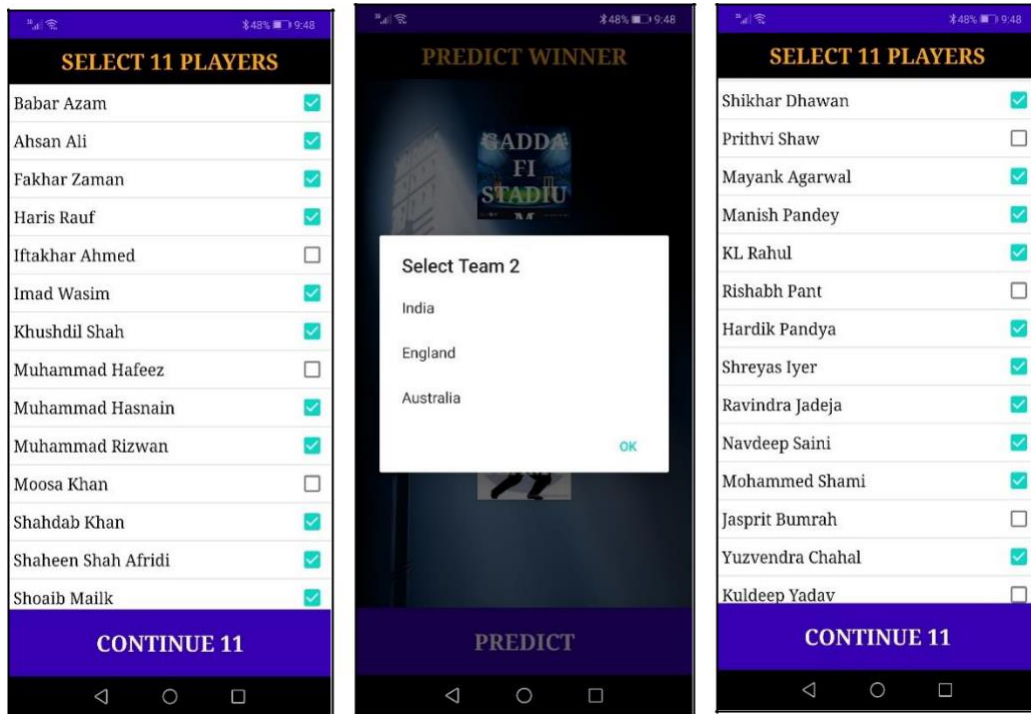


Figure 5.4: Match Prediction II



Figure 5.5: Match Prediction III

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

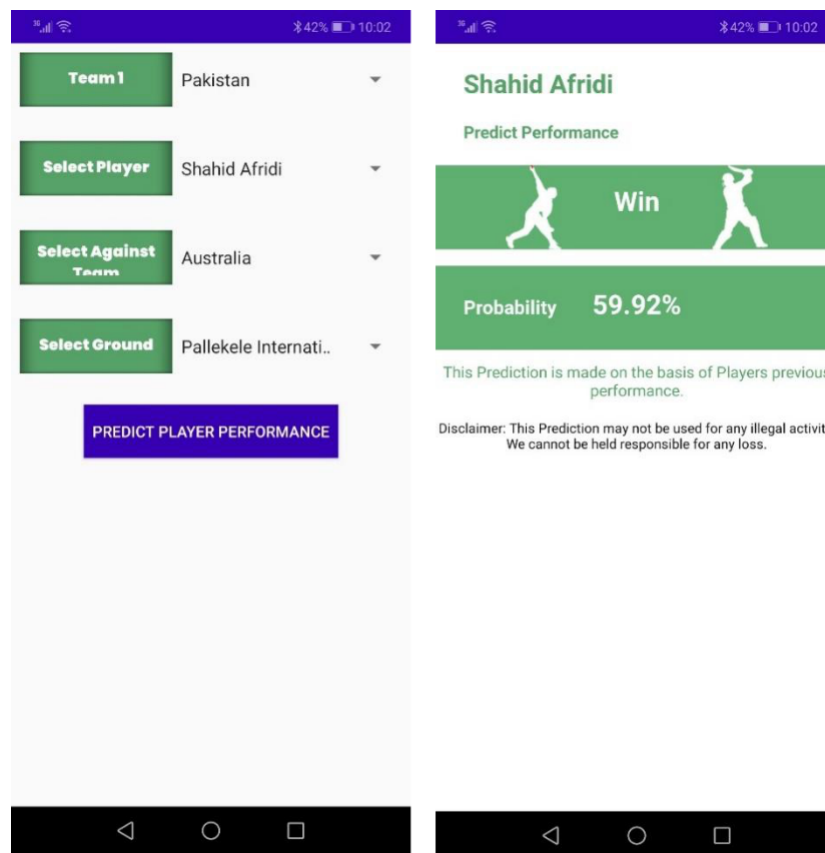


Figure 5.5: Player Prediction

Chapter 6: Testing

6.1 Introduction

Whenever new software is built the most important aspect to check is whether the system works according to the given requirements. To check if the system is following the flow that was defined in the requirements and to check that it is producing a correct result as expected from it. These factors make the testing part important because it is useful to evaluate how well the system works.

6.2 Test Cases

6.2.1 Login

Test Case ID	TC-CPASS-01	
Associated Use Case	Log-in	
Functionality to be tested	Admin successfully login to the system	
Actor	Admin	
Pre-Conditions	Admin must be register in the system	
Post Conditions	Admin Login in the system	
	Input Data/Events	Expected Output Data/Events

	<p>Actor must open login page</p> <p>Actor Enter username</p> <p>Actor Enter password</p> <p>Actor Click the Log-In Button.</p>	<p>An error message is shown if user don't enter user name, password and press the login button</p> <p>An error message is shown if user only enter the</p>
		<p>username and press the login button</p>
Test Status	System has passed.	

Table 6.1 : Login

6.2.2 Adding Data

Test Case ID	TC-CPASS-01	
Associated Use Case	Adding Data	
Functionality to be tested	Admin successfully add data into the system	
Actor	Admin	
Pre-Conditions	Admin must be register in the system	
Post Conditions	Admin Login in the system	
	Input Data/Events	Expected Output Data/Events

	Actor enter the data Actor Click the Submit Button.	An error message is shown if user enter “123” in place of name An error message is shown if user don’t enter any data and
		press submit button
Test Status	System has passed.	

Table 6.2 : Adding Data

6.2.3 Logout

Test Case ID	TC-CPASS-01	
Associated Use Case	Adding Data	
Functionality to be tested	Admin successfully add data into the system	
Actor	Admin	
Pre-Conditions	Admin must be register in the system	
Post Conditions	Admin Login in the system	
	Input Data/Events	Expected Output Data/Events

	Actor enter the data Actor Click the Submit Button.	An error message is shown if user enter “123” in place of name An error message is shown if user don’t enter any data and
		press submit button
Test Status	System has passed.	

Table 6.3 : Logout

6.2.4 Team Prediction

Test Case ID	TC-CPASS-01	
Associated Use Case	Team Prediction	
Functionality to be tested	Test successful prediction of team (winner).	
Actor	User	
Pre-Conditions	The user must run application	
Post Conditions	The user has successfully predicted the team (winner).	
	Input Data/Events	Expected Output Data/Events

CPASS - Cricket Player Performance System & Pakistan Army Sports Selection System

	<p>User run application Actor click on “Select Team 1” button. Actor selects the Team</p>	<p>An error message is shown if Internet is not connected. An error message shown if user select the team</p>
	<p>from drop down menu. 8. Actor click on “Select Team 2” button. 9. Actor selects the Team from drop down menu 10. Actor click on “Select Ground” button. 11. Actor selects the ground from the drop down menu. 12. Actor clicks on “Next” button.</p>	<p>whose data is not available in database. 5. User has successfully selected the teams.</p>
	Actual Input Data/Events	Actual Output Data/Events
	<p>1. Actor must choose eleven players from the list of team 1. 2. Actor clicks on “next” button.</p>	<p>1. An Error message shown if user select more than eleven players.</p>

	<ol style="list-style-type: none"> 1. Actor must choose eleven players from the list of team 2. 2. Actor clicks on “predicts Result” button. 	<ol style="list-style-type: none"> 1. An error message shown if user select more than eleven players.
	<ol style="list-style-type: none"> 1. Actor Select team 2 players more than team 1 players 	<ol style="list-style-type: none"> 1. An error message shown that team 1 players not equal to team 2
Test Status	System has passed.	

Table 6.4 : Team Prediction

6.2.5 Player Performance Prediction

Test Case ID	TC-CPASS-02	
Associated Use Case	Predict Player Performance	
Functionality to be tested	Test successful prediction of a player performance.	
Actor	User	
Pre-Conditions	The user must run application	
Post Conditions	The user has successfully predicted of a player performance.	
	Input Data/Events	Expected Output Data/Events

	<p>User run application</p> <p>Actor click on “Select Team” button.</p> <p>Actor selects the Team from drop down menu.</p> <p>Actor click on “Select Player” button.</p> <p>Actor selects player from drop down menu.</p>	<p>An error message is shown if Internet is not connected.</p> <p>An error message shown if user select the team whose data is not available in database.</p> <p>User has successfully selected the teams.</p>
	<p>Actor click on “Select against team” button.</p> <p>Actor selects against team from drop down menu.</p> <p>Actor click on “Select Ground” button.</p> <p>Actor select ground from drop down menu.</p> <p>Actor clicks on “Predict Player Performance” button.</p>	
Test Status	Application has passed.	

Table 6.5 : Player Performance Prediction

6.2.6 Previous Result

Test Case ID	TC-CPASS-03
Associated Use Case	View Previous Result

Functionality to be tested	Test successful View Previous Result.	
Actor	User	
Pre-Conditions	The user must run application	
Post Conditions	The user has successfully View Previous Result.	
	Input Data/Events	Expected Output Data/Events
	User run application User View list of the previous result.	9. An error message is shown if Internet is not connected.
Test Status	Application has passed.	

Table 6.6 : Previous Result

6.3 Summary

The chapter has defined the major test case scenarios that can affect the application if they have errors. The test case scenarios define all the possible inputs and outputs to and from the application. These test case scenarios can be used to define actual test cases which then can be used to check the system for any errors. Apart from the functional testing of the system the testing of actual algorithm and there working will require a sizeable dataset. The dataset can then be used to actually test the systems functionality and accuracy of the recommendations produced by the application. But due to small time period of development and non-availability of dataset the actual testing will be performed on smaller scale with a limited dataset.

Chapter 7: Conclusion and outlook

7.1 Introduction

In this chapter we conclude our project report as it has reached to its end. Here we are presenting the overall summary of the project right from the first day of its beginning till its day of successful completion. As well as, the difficulties and hurdles we faced in implementing the project successfully. This chapter will also highlight the high level discussion of the project, as well as our thoughts and feeling on its successful completion.

7.2 Achievements and Improvements

The major achievement in our eyes is the level of understanding that we as a team developed during the course of development. We as a team have learned many things related to the field of cricket prediction and have gathered a lot of knowledge on it. As a team we have looked upon many problems and faced them together and every problem taught us a new lesson academically and professionally. The project has helped us all develop individual skills in problem solving and researching. The research part of our system has given us a good habit of reading which is compulsory for any person who wants to pursue his dreams in this profession. After the development this cricket prediction application we are able to understand how these systems work and how to develop such systems. Apart from academic perspective we all have developed skills to work as a team, to communicate and discuss problems and be responsible for what is assigned to you.

7.3 Critical Review

This project is on the baseline of developing cricket prediction application. The basic and most important functionality of collaborative filtering and personalization of application has been implemented. The application will be optimized in future to develop a more efficient working. The algorithms that are implemented are also the basic prediction algorithm that can be optimized and mixed with more complex and efficient algorithms.

7.4 Future Recommendation/Outlook

The application developed at starting point and just to support a theory and give proof of concepts. So, the application developed may have missed the issues such as time efficiency due to limited duration of development.

- In future more complex and efficient prediction algorithms can be used to make the application work more efficiently and with higher quality.
- Implement the crawler that will extract data from cricinfo.com to our database that will be more efficient and accurate.
- Learn more data sciences in future that our application will predict near to accurate result.

7.5 Summary

The application developed is a predictor application by its nature. It has the capability of performing the basic prediction about player and match. The application is a first application which will use the predicting algorithm. User can open the application using the smart phone with Android operating system. The application is uploaded on the Google Play Store any user can download the application and use for predication

Chapter 8: Bibliography

- [1] Official Google Android Design Guidelines
<http://developer.android.com/design/getstarted/principles.html>
- [2] Weka Tool and Library <http://www.cs.waikato.ac.nz/ml/weka/>
- [3] Neural Network http://en.wikipedia.org/wiki/Artificial_neural_network
- [4] KNN http://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm
- [5] Logistic Regression http://en.wikipedia.org/wiki/Logistic_regression
- [6] Data Analysis http://en.wikipedia.org/wiki/Data_analysis
- [7] Big Data http://www.webopedia.com/TERM/B/big_data_analytics.html
- [8] Data Souse <http://www.espncricinfo.com/>.

FYP Report

ORIGINALITY REPORT

13%

SIMILARITY INDEX

3%

INTERNET SOURCES

0%

PUBLICATIONS

13%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	3%
2	Submitted to University of Wales, Bangor Student Paper	1%
3	Submitted to Segi University College Student Paper	1%
4	Submitted to Universiti Tenaga Nasional Student Paper	1%
5	www.coursehero.com Internet Source	1%
6	Submitted to The University of Manchester Student Paper	1%
7	Submitted to HELP UNIVERSITY Student Paper	<1%
8	Submitted to Queen's University of Belfast Student Paper	<1%
•	www.omicsonline.org	

<1%

10

Submitted to Colorado Technical University
Online

Student Paper

<1%

11

Submitted to University of Greenwich

Student Paper

<1%

12

espace.library.uq.edu.au

Internet Source

<1%

13

Submitted to Miami University of Ohio

Student Paper

<1%

14

Submitted to Leeds Beckett University

Student Paper

<1%

15

Submitted to University of Ulster

Student Paper

<1%

16

Submitted to University of Canterbury

Student Paper

<1%

17

Submitted to University of Hong Kong

Student Paper

<1%

18

Submitted to Universiti Malaysia Pahang

Student Paper

<1%

19

Submitted to British University in Egypt

Student Paper

<1%

20

Internet Source

<1%

21

www.theseus.fi
Internet Source

<1%

22

Submitted to University of Bedfordshire
Student Paper

<1%

23

Submitted to University of Central England in
Birmingham
Student Paper

<1%

24

Submitted to University of Westminster
Student Paper

<1%

25

C. Nusai, W. Chankeaw, B. Sangkaew. "Dairy
cow-vet: A mobile expert system for disease
diagnosis of dairy cow", 2015 IEEE/SICE
International Symposium on System Integration
(SII), 2015
Publication

<1%

26

Submitted to South Bank University
Student Paper

<1%

27

Submitted to Middlesex University
Student Paper

<1%

28

Submitted to University of Sheffield
Student Paper

<1%

29

Submitted to London School of Commerce
Student Paper

<1%

30

Submitted to CSU, San Jose State University

Student Paper

<1%

31

www.ru.is

Internet Source

<1%

32

Submitted to University of East London

Student Paper

<1%

33

Submitted to University of Leeds

Student Paper

<1%

34

Submitted to University of Sunderland

Student Paper

<1%

35

Submitted to University of Moratuwa

Student Paper

<1%

Exclude quotes

Off

Exclude matches

Off

Exclude
bibliography

On

