

Cricket Management System Project Report

Kamal Acharya¹

¹Tribhuvan University

August 05, 2024

AN
INTERNSHIP REPORT
ON
CRICKET MANAGEMENT SYSTEM
PROJECT
BY
KAMAL ACHARYA
(Tribhuvan University)

Date: 2022/07/07

TABLE OF CONTENT

Table of Contents

1. SOFTWARE REQUIREMENT SPECIFICATION	1
1.1 INTRODUCTION	1
1.2 FEATURES	1
1.3 PURPOSE	1
1.4 SCOPE	2
1.5 OVERALL DESCRIPTION	2
1.6 PRODUCT PERSPECTIVE	2
1.7 CONSTRAINTS, ASSUMPTIONS AND DEPENDENCIES	3
1.8 SPECIFIC REQUIREMENTS	3
1.8.1 INTERFACE REQUIREMENTS.....	3
1.8.2 FUNCTIONAL REQUIREMENTS	3
1.8.3 NON-FUNCTIONAL REQUIREMENTS.....	4
1.8.4 DESIGN CONSTRAINTS	4
1.9 DATA FLOW DIAGRAMS	5
1.9.1 DFD LEVEL 0.....	5
1.9.1 DFD LEVEL 1	6
1.9.1 DFD LEVEL 2	7
1.10 DATA DICTIONARY.....	8
2. SIZE ESTIMATION AND SCHEDULING	9
2.1 FUNCTION POINTS ESTIMATION	10
2.2 SIZE ESTIMATION.....	11
2.3 COST ESTIMATION	11
2.4 SCHEDULING.....	12
3. ARCHITECTURAL DESIGN.....	13
3.1 VIEW.....	14
3.2 RETRIEVE	14
4. RISK ANALYSIS	15
4.1 NEW UNPROVEN TECHNOLOGY.....	15
4.2 ORGANISATIONAL RISK	15
4.3 PROJECT DELAY	16
4.4 RISK OF STORAGE AND DATABASE SPACE.....	19
5.2 SCREENSHOT OF THE SOFTWARE.....	24
6. TESTING.....	29
7. USERS MANUAL.....	34
7.1 LOGIN	34
7.2 RETRIEVAL.....	35
8. CONCLUSION.....	37

LIST OF FIGURES

Figure 1: DATA FLOW DIAGRAM LEVEL-0	5
Figure 2: DATA FLOW DIAGRAM LEVEL-1	6
Figure 3: DATA FLOW DIAGRAM LEVEL-2	7
Figure 4: GANTT CHART	12
Figure 5: ARCHITECTURAL DESIGN	13

LIST OF TABLES

Table 1- DATA DICTIONARY	8
Table 2- PROJECT ESTIMATORS	9
Table 3 - FUNCTIONAL POINT ANALYSIS.....	10
Table 4 – RISK ASSESSMENT	16
Table 5 – RMMM1	17
Table 6 – RMMM2	18

1. Software Requirement Specification

1.1 Introduction

The aim of this project is to provide the complete information of the National and International statistics. The information is available country wise and player wise. By entering the data of each match, we can get all type of reports instantly, which will be useful to call back history of each player. Also the team performance in each match can be obtained. We can get a report on number of matches, wins and lost.

1.2 Features

The project will enable the user to view:

- Every Cricketer's Information
- Every Cricketer's Personal Achievements
- International Travel News
- Upcoming Series
- Every Team's Information
- ODI and test ranking
- Increment of each ball ,over, run is done automatically
- Commentary of each ball

1.3 Purpose

This document aims to give a brief description about the Cricket Management System Project. This project is very use for Cricket match broadcasters to get information quickly. Also for Cricket lovers who are very much interested in Cricket Statistics. In other words this document will provide a basis for validation and verification.

1.4 Scope

The project is designed very user friendly such that even people who know only the basic operation of the computer can use this software. So no technical expertise is required to use this software.

This software is also functional to find out the application of cricket management system and to provide information and manage the system regarding cricket matches and team training. It also helps cricket teams to register new members.

1.5 Overall Description

The overall description of our project can be stated as creating and managing the database, developing a friendly user interface to manipulate the database, provide an authentication mechanism to safely accomplish tasks mentioned above.

1.6 Product Perspective

With the CMS (Cricket Management System) we will provide them with capabilities and properties organized neatly. CMS which is an online intranet System will be used by anyone who wishes to find information for the players and teams.

Admin will basically query and edit the database via CMS. It will also calculate the rank of each player depending on their Strike Rate and no. of matches played.

1.7 Constraints, Assumptions and Dependencies

Regularity Policies:

Each user must login first to have access to information of players. In other words, each user has an account created by the Admin.

Hardware Limitations:

There is no limitation in the operating system in which CMS will work. However, the CMS system and the database will work on a server that needs to be always online.

1.8 Specific Requirements

1.8.1 Interface Requirements

All the users will see the same page when they enter CMS (Cricket Management System). There will be 2 options: One for the user and one for the admin. The admin is required to login through an ID and a password so as to make changes in the database or to add new information to it. On the other hand users can retrieve the information about the different players, along with their personal details and team details as well after getting logged in to their respective accounts.

1.8.2 Functional Requirements

Requirements which are related to functional aspects of software fall into this category. According to the CMS we have:

- View option is given to the user to view the details of their favorite players along with their personal details.
- Retrieve function lets the user to offline download the details of the players and the teams.

1.8.3 Non-Functional Requirements

Requirements which are not related to functional aspects of software fall into this category. They are implicit or expected characteristics of the software, which users make assumption of. The Non-Functional Requirements of CMS are:

- **Logging-** Both the Admin and the User need to log in to access data from CMS.
- **Storage-** The information regarding the players and their teams is stored on a database. The information includes the players' personal details along with the achievements in their cricketing career.
- **Performance-** The CMS satisfies the user requirements and can handle multiple users at a time, resulting in better performance.
- **Cost-** The overall cost of the project is minimal which is easily affordable by one and all.
- **Flexibility-** The CMS is highly flexible as it can be accessed by any person or the organization without any difficulty.

1.8.4 Design Constraints

- **Hardware Requirements**

The CMS application will be storing 500 players' personal data which will roughly require 500mb of database storage space.

- **Software Requirements**

1. Since the CMS is an android based application so the user will require mobile handsets for the deployment of the software.
2. To store and fetch data from the database online i.e. cricdb, the user also requires an active internet connection for smooth working of the project.
3. The CMS will work properly on any mobile set with android version 4.4 or higher.
4. Application will run on 256mb or higher of RAM.
5. It requires a space of 2.5mb to deploy onto the machine.

1.9 Data Flow Diagram

Data flow diagrams (also called data flow graphs) are commonly used during problem analysis. A DFD shows the flow of data through a system. It views a system as a function that transforms the inputs into desired outputs. The agent that performs the transformation of data from one state to another is called a process (or a bubble). Thus, a DFD shows the movement of data through the different transformations or processes in the system.

1.9.1 DFD Level 0

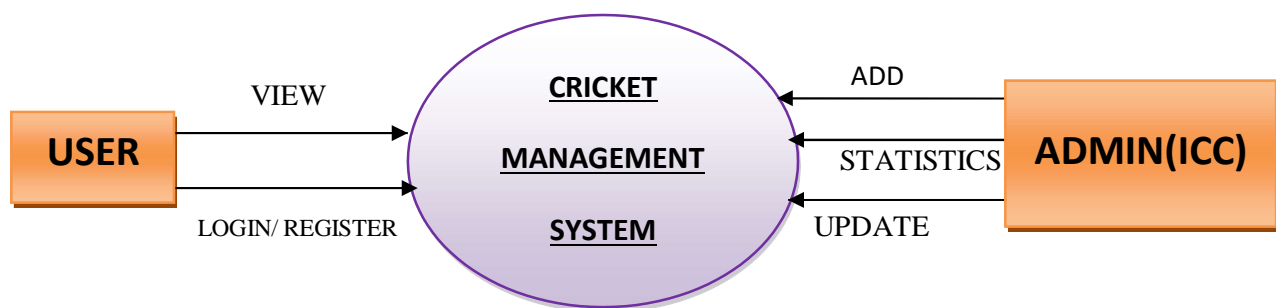


Figure 1: Data Flow Diagram Level 0

1.9.2DFD Level 1

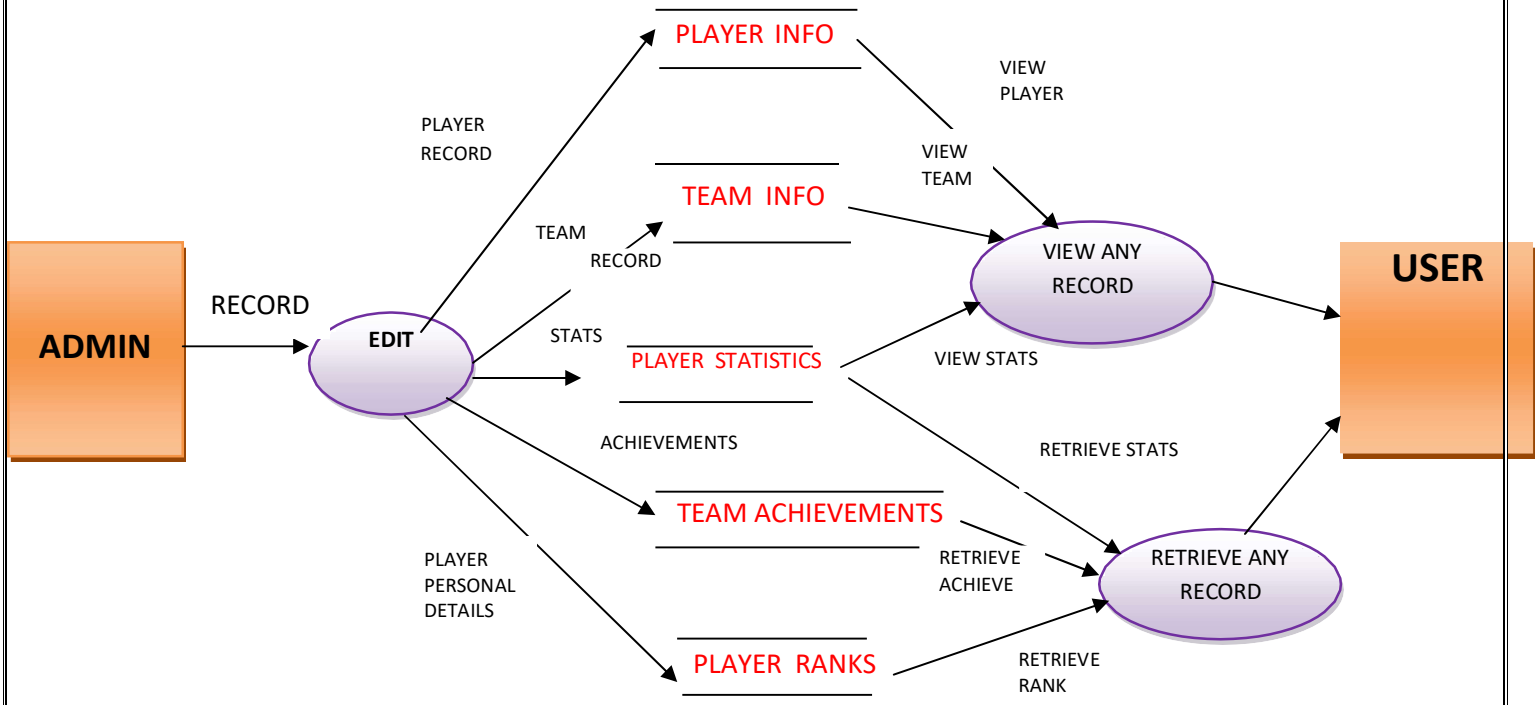


Figure 2: Data Flow Diagram Level 1

1.9.3 DFD Level 2

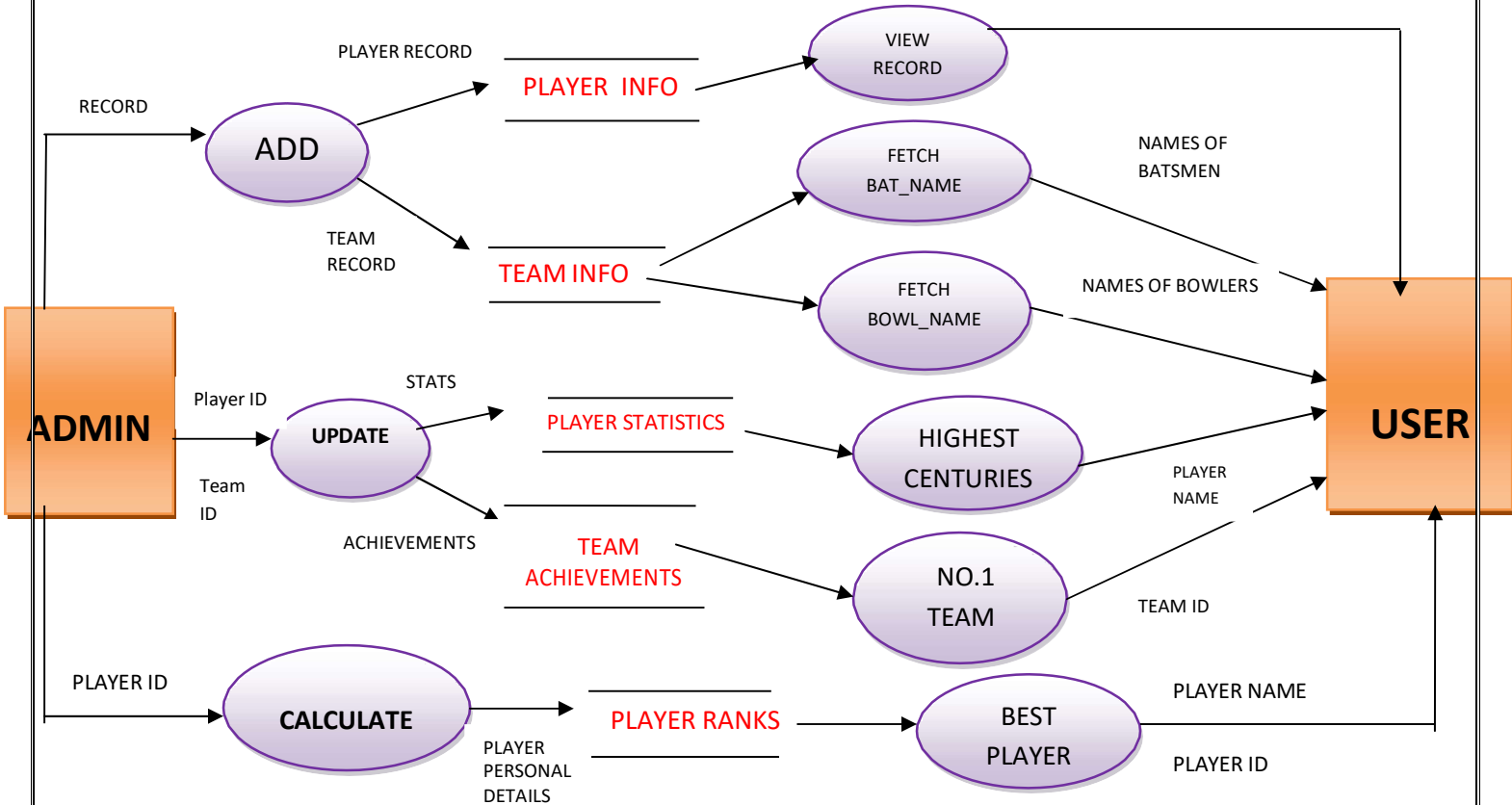


Figure 3: Data Flow Diagram Level 2

1.10 Data Dictionary

<u>DATA</u>	<u>DESCRIPTION</u>
Record	Pid + Pname + DOB + Team + Bat_Style + Bowl_Style + Strike_Rate + No_Of_Centuries + Tid + Tname + Captain + No_Of_Batsmen + No_Of_Bowlers + Team_Colour + Coach_Name +...
Player Record	Pid+ Pname+ DOB+ Team+ Bat_Style + Bowl_Style+ Strike_Rate+ No_Of_Centuries +...
Team Record	Tid + Tname+ Captain + No_Of_Batsmen+ No_Of_Bowlers + Team_Colour + Coach_Name+...
Stats	Man_Match + Man_Series + Highest_Centuries
Achievements	Won_matches + Lost_matches + Tie_matches
Player Personal Details	Pid + Pname + DOB

Table 1: Data Dictionary

2 Size Estimation And Scheduling

		GRADE VALUE
1.	Does the system require reliable backup and recovery?	5
2.	Are specialized data communications required to transfer Information to or from the application?	3
3.	Are there distributed processing functions?	4
4.	Is performance critical?	0
5.	Will the system run in an existing, heavily utilized Operational environment?	5
6.	Does the system require on-line data entry?	5
7.	Does the on-line data entry require the input transaction to be built over multiple screens or operations?	3
8.	Are the Master files updated online?	5
9.	Are the inputs, outputs, files, or inquiries complex?	2
10.	Is the internal processing complex?	1
11.	Is the code designed to be reusable?	1
12.	Are conversions and installations included in the design?	0
13.	Is the system designed for multiple installations in different Organizations?	5
14.	Is the application design to facilitate change and for ease of use by the user?	5

Table 2: Project Estimators

2.1 Function Points Estimation

Considering the following assumptions:

3-4 INPUTS AS LOW

4-7 INPUTS AS AVERAGE

>7 INPUTS AS HIGH

	COUNT			WEIGHING FACTOR			TOTAL COUNT
	Simple	average	Complex	Simple	Average	Complex	
Number of internal inputs	2	2	1	3	4	6	20
Number of external outputs	3	2	1	4	5	7	29
Number of inquiries	2	2	2	3	4	6	26
Number of internal logical files	0	0	3	7	10	15	45
Number of external interfaces	1	1	1	5	7	10	22
			Total FPs				142

Table 3: Function Point Analysis

Thus, the UNADJUSTED FUNCTIONAL POINTS (UFP) ARE:

$$\begin{aligned}\mathbf{UFP} &= \text{Total I/P} + \text{Total O/P} + \text{Total ILF} + \text{Total EIF} + \text{Total Enquiries} \\ &= 20 + 29 + 45 + 22 + 26 \\ &= \mathbf{142}\end{aligned}$$

Now, COMPLEXITY ADJUSTMENT FACTOR (CAF) IS:

$$\begin{aligned}\mathbf{CAF} &= 0.65 + (0.01 * \sum fi) \\ &= 0.65 + (0.01 * 3 * 14) \\ &= \mathbf{1.07}\end{aligned}$$

So the Functional Points (FP) is:

$$\begin{aligned}\mathbf{FP} &= \mathbf{UFP * CAF} \\ &= \mathbf{151.94}\end{aligned}$$

2.2 Size Estimation

Assuming that average productivity for such a system is 6.5 FP/pm i.e. 6.5 functional points per person month, then the estimates are:

$$\begin{aligned}\text{Size} &= \text{Total FP's} / \text{Average productivity} \\ &= 151.94 / 6.5 \\ &= 23.37 \text{ pm i.e. 24 person month.}\end{aligned}$$

2.3 Cost Estimation

Assuming that the cost is Rs.5000/pm, i.e. 5000 per person month.

$$\begin{aligned}\text{Cost} &= \text{cost per person month} * \text{effort} \\ &= \text{Rs. } (5000 * 14) = \text{Rs.70,000}\end{aligned}$$

2.4 Scheduling

Scheduling is the culmination of a planning activity that is a primary component of Software project management. When combined with estimation methods and risk analysis, scheduling establishes a road map for the project manager. The characteristics of the project are used to adapt an appropriate task set for the work to be done. A task network depicts each engineering task, its dependency on other tasks, and its projected duration.

GANTT CHART

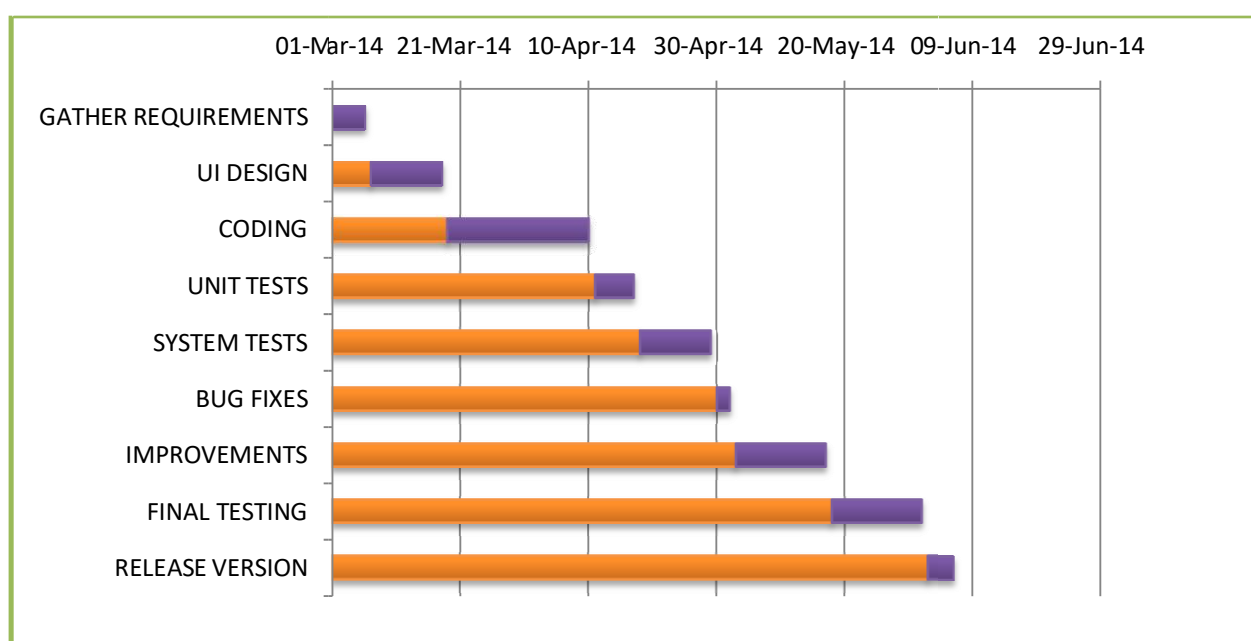


Figure 4: Gantt Chart

3. ARCHITECTURAL DESIGN

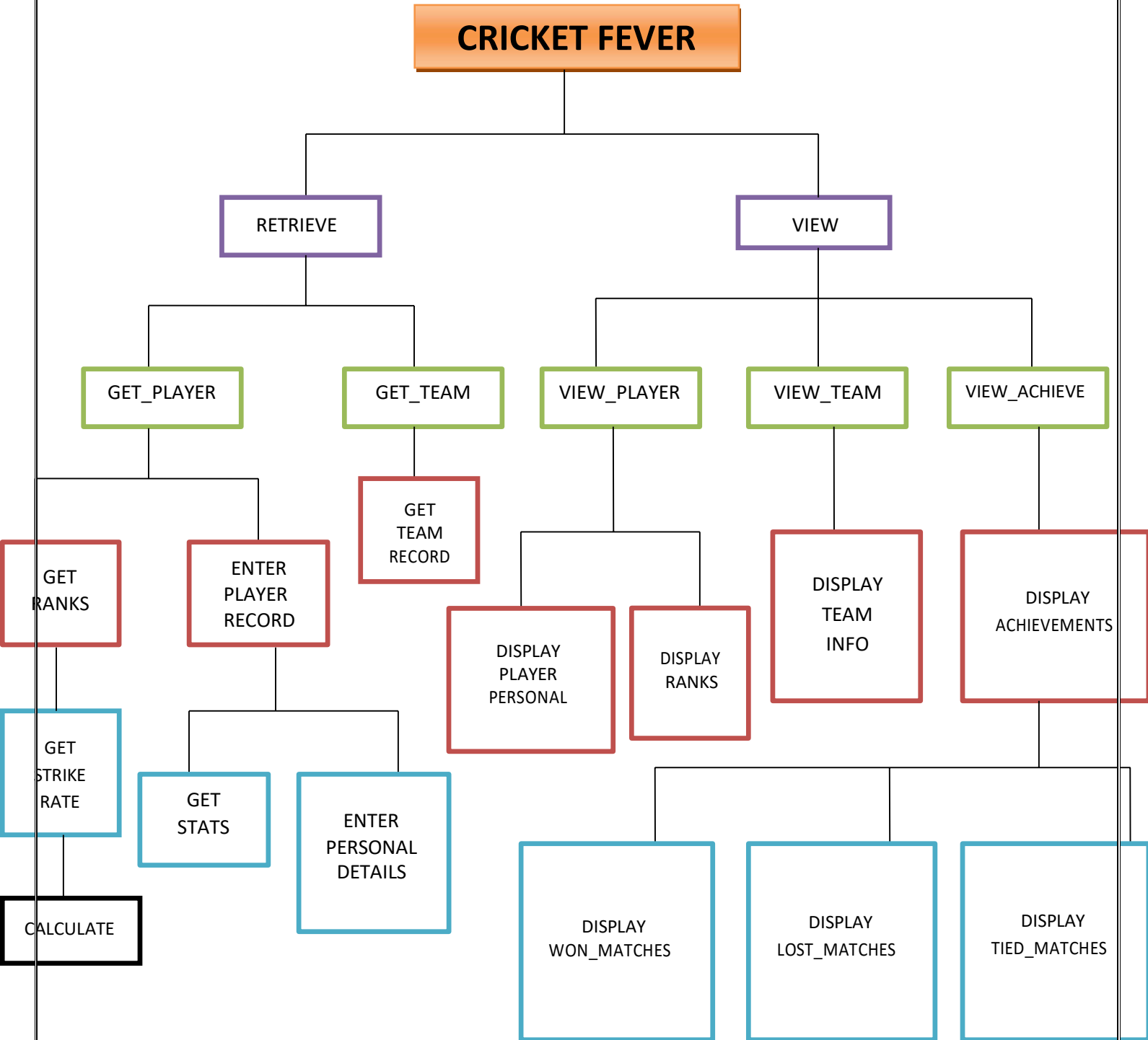


Figure 5: Architectural design

The architectural design of the software defines every module of the application in detail in a hierarchical way. The CMS application is mainly divided into two modules:

3.1 VIEW

The VIEW MODULE can view information about the PLAYER RECORD , TEAMRECORD and ACHIEVEMENTS of the Team.

It will display the personal information about the player and the ranks. Also , It will also display the won, lost and tied matches.

3.2 RETRIEVE

The RETRIEVE MODULE will retrieve details of the player and the team. Here the user will enter personal details and statistics of the player by the help of which STRIKE RATE will be calculated .

The approach used is a top to bottom approach.

PROJECT REPORT

Team Information

The administrator Can Add/Update the teams With the details and the user can view the overall information of the

the team detail module allows the user to browser The details of the team profile and management.

Team module allows user to browse different teams

where only one team can be selected at a time and get the detail

when selected on a particular team.

This module gives the detail of team such as team logo, team name, team information, team captain, address, contact no show match details

Player Details

this will come under team information which gives the basic details about the players.

allow the user to have a glimpse of the player's profile.

it give the personal details of a player with their batting style, bowling style and his position in the team along with their achievements

Batting Performance

It gives the batsman profile, statistic, and summary of his batting performance in his current and previous matches that he had played.

Bowling Performance

It gives the bowling profile, statistics, and summary of his bowling performance in his current and previous matches that he had played.

Results

This module gives the summary that was played.

It also gives the overall detailed result of that particular match that was played.

Detail such as scores scored by both the teams Over's completed, loss of wickets.

Gallery

it shows the best captured moments, pictures, video etc... That was taken during the course of the matches.

It will dynamic.

Here the photo is uploaded which was clicked/recorded by the fans during the matches.

The user cannot upload the image here due to censorship issues.

Instead, the user is supposed to e-mail the image to admin and then admin uploads it to the website.

Administrator

The administrator is allowed to access all the services in the system. The main objective of the administrator is to add/update the summary of the matches and also to add/update the schedules of the matches.

Users/Visitors

These users will get the full access of the website but they cannot do any changes on the Online Score System website, they can enjoy the services provides by the administrator.

```

#include <stdio.h>
#include <stdlib.h>

void req_rate(float a, float b, float c, float d, float e);
void runs_needed(int a, int b, int c, int d, int e);
void projected_score(int a, int b, int c, int d, float e);
void result(int a, int b, int e, int g, char f, char c[], char d[]);
float curr_rate(float a, float b, float c);
void ball_played(int *a, int *b, int *c, int *d, int *e, int *f, int *g, int *h, int *i, int *j);
int string_comp(char a[], char b[], char c[]);
float curr_rate(float a, float b, float c);
int main(int argc, char *argv[])
{
    int wd = 0, nb = 0, lb = 0, b = 0, extras = 0, overs = 0, T_overs = 0, k = 0, balls = 0, T = 0, score = 0,
wickets = 0, target = 0, q = 0, c[11], j = 1, f = 0, s = 0;

    float c_rate;

    char toss[15], team_1[15], team_2[15], choice, runs;
    printf("Match between:\n");
    gets(team_1);
    printf("versus\n");
    gets(team_2);
    printf("Toss won by: ");
    gets(toss);
    printf("Enter B to choose bat first and F to field first: ");
    scanf("%c", &choice);
    printf("Enter overs to be played by each side: ");
    scanf("%d", &T_overs);
    system("cls");
    T = string_comp(team_1, team_2, toss);
    while (overs <= T_overs || overs >= T_overs)
    {

```

```

do
{
    if (wickets > q)
    {
        *(c + wickets) = score;
    }
    q = wickets;
    system("cls");
    printf("\t\t\t%s vs %s (%d-overs match)\n", team_1, team_2, T_overs);
    if (T == 1)
    {
        if (choice == 'B' || choice == 'b')
        {
            printf("%s won the toss and elected to bat first\n", team_1);
            printf("\n\n%s %d-%d\t\tovers %d.%d\t\t", team_1, score, wickets, overs, balls);
        }
        else if (choice == 'F' || choice == 'f')
        {
            printf("%s won the toss and elected to field first\n", team_1);
            printf("\n\n%s %d-%d\t\tovers %d.%d\t\t", team_2, score, wickets, overs, balls);
        }
    }
    else if (T == 2)
    {
        if (choice == 'B' || choice == 'b')
        {
            printf("%s won the toss and elected to bat first\n", team_2);
            printf("\n\n%s %d-%d\t\tovers %d.%d\t\t", team_2, score, wickets, overs, balls);
        }
        else if (choice == 'F' || choice == 'f')
        {

```

```

printf("%s won the toss and elected to field first\n", team_2);
printf("\n\n%s %d-%d\t\tovers %d.%d\t\t", team_1, score, wickets, overs, balls);
}
}
if (overs > 0 && score > 20)
{
if (overs == T_overs || wickets == 10)
{
printf("\n");
k++;
break;
}
c_rate = curr_rate(score, overs, balls);
printf("run rate %0.2f\n", c_rate);
}
if (balls == 0)
{
printf("\n");
if (extras > 0)
{
printf("extras: %d(wd %d,nb %d,lb %d,b %d)\t", extras, wd, nb, lb, b);
}
if (f > 0 || s > 0)
{
printf("boundaries: %d fours,%d sixes\n", f, s);
}
else
printf("\n");
}
if (overs > 0 && score > 20 && balls == 0)
{

```



```

        projected_score(score, overs, T_overs, balls, c_rate);
    }
    printf("\n");
    if (overs == T_overs || wickets == 10)
    {
        k++;
        break;
    }
    ball_played(&balls, &score, &wickets, &extras, &wd, &nb, &lb, &b, &f, &s);
} while (balls < 6);
if (balls == 6)
{
    balls = 0;
}
if (overs < T_overs)
{
    overs++;
}
if (k > 0)
{
    break;
}
}
printf("\nexttras: %d(wd %d,nb %d,lb %d,b %d)\tboundaries: %d fours,%d sixes\n", extras, wd, nb,
lb, b, f, s);
if (wickets > 0)
{
    printf("\nfall of wickets: ");
    for (j = 1; j <= wickets; j++)
    {
        if (j == 1 || j == 6)

```

```

        printf("\n");
    else
        printf("\t");
    printf("%d-%d", j, c[j]);
}
}
printf("\n\ninnings completed.");
sleep(10);
for (k = 10; k >= 0; k--)
{
    system("cls");
    printf("\n\n\n\n\t\tINNINGS BREAK...\n\t\t\t\t%d", k);
    sleep(1);
}
target = score + 1, score = 0, overs = 0, k = 0, wd = 0, nb = 0, lb = 0, b = 0, balls = 0, wickets = 0,
extras = 0, j = 1, q = 0, f = 0, s = 0;
while (overs <= T_overs || overs >= T_overs)
{
    do
    {
        if (wickets > q)
        {
            *(c + wickets) = score;
        }
        q = wickets;
        system("cls");
        printf("\t\t\t%s vs %s (%d-overs match)\n", team_1, team_2, T_overs);
        if (T == 1)
        {
            if (choice == 'B' || choice == 'b')
            {

```

```

printf("%s gives target of %d runs to %s to win\n", team_1, target, team_2);
printf("\n\n%s %d-%d\t overs %d.%d\t", team_2, score, wickets, overs, balls);
}
else if (choice == 'F' || choice == 'f')
{
printf("%s gives target of %d runs to %s to win\n", team_2, target, team_1);
printf("\n\n%s %d-%d\t overs %d.%d\t", team_1, score, wickets, overs, balls);
}
}
else if (T == 2)
{
if (choice == 'B' || choice == 'b')
{
printf("%s gives target of %d runs to %s to win\n", team_2, target, team_1);
printf("\n\n%s %d-%d\t overs %d.%d\t", team_1, score, wickets, overs, balls);
}
else if (choice == 'F' || choice == 'f')
{
printf("%s gives target of %d runs to %s to win\n", team_1, target, team_2);
printf("\n\n%s %d-%d\t overs %d.%d\t", team_2, score, wickets, overs, balls);
}
}
if (overs > 0 && score > 20)
{
if (overs == T_overs || wickets == 10 || target <= score)
{
printf("\n");
k++;
break;
}
}
c_rate = curr_rate(score, overs, balls);

```

```

    printf("run rate %0.2f", c_rate);
    req_rate(score, overs, T_overs, balls, target);
}
if (balls == 0)
{
    printf("\n");
    if (extras > 0)
    {
        printf("extras: %d(wd %d,nb %d,lb %d,b %d)\t", extras, wd, nb, lb, b);
    }
    if (f > 0 || s > 0)
    {
        printf("boundaries: %d fours,%d sixes\n", f, s);
    }
    else
        printf("\n");
}
if (overs > 0 && score > 20)
{
    runs_needed(score, overs, T_overs, balls, target);
}
printf("\n");
if (wickets == 10 || target <= score || overs == T_overs)
{
    k++;
    break;
}
ball_played(&balls, &score, &wickets, &extras, &wd, &nb, &lb, &b, &f, &s);
} while (balls < 6 && overs < T_overs);
if (balls == 6)
{

```

```

        balls = 0;
    }
    if (overs < T_overs)
    {
        overs++;
    }
    if (k > 0)
    {
        break;
    }
}

printf("\nextras: %d(wd %d,nb %d,lb %d,b %d)\tboundaries: %d fours,%d sixes\n", extras, wd, nb,
lb, b, f, s);

if (wickets > 0)
{
    printf("\nfall of wickets: ");
    for (j = 1; j <= wickets; j++)
    {
        if (j == 1 || j == 6)
            printf("\n");
        else
            printf("\t");
        printf("%d-%d", j, c[j]);
    }
}

printf("\n\ninnings completed.");

sleep(10);

for (k = 0; k <= 7; k++)
{
    system("cls");
    sleep(1);
}

```

```

    printf("\n\n\n");
    result(target, score, T, wickets, choice, team_1, team_2);
    sleep(1);
}
getche();
return 0;
}
float curr_rate(float a, float b, float c)
{
    float d;
    if (c == 0)
        d = a / b;
    else
        d = a / (b + (c / 10));
    return d;
}
int string_comp(char a[], char b[], char c[])
{
    int i = 0, ab = 0, j = 0, k = 0;
   strupr(a);
   strupr(b);
   strupr(c);
    for (i = 0; i < 3; i++)
    {
        if (a[i] == c[i])
            j++;
        else if (b[i] == c[i])
            k++;
        else
            break;
    }
}

```

```

if (j == 3)
    ab = 1;
else if (k == 3)
    ab = 2;
else
    ab = 0;
return ab;
}

void ball_played(int *a, int *b, int *c, int *d, int *e, int *f, int *g, int *h, int *i, int *j)
{
    char runs;
    int t;
    scanf("%c", &runs);
    switch (runs)
    {
        case '0':
            *a = *a + 1;
            break;
        case '1':
            *b = *b + 1;
            *a = *a + 1;
            break;
        case '2':
            *b = *b + 2;
            *a = *a + 1;
            break;
        case '3':
            *b = *b + 3;
            *a = *a + 1;
            break;
        case '4':

```

```
*b = *b + 4;

*a = *a + 1;

*i = *i + 1;

break;
case '5':

    *b = *b + 5;

    *a = *a + 1;

    break;
case '6':

    *b = *b + 6;

    *a = *a + 1;

    *j = *j + 1;

    break;
case 'w':
case 'W':

    printf("enter runs on wide ball: ");

    scanf("%d", &t);

    *b = *b + 1 + t;

    *d = *d + 1 + t;

    *e = *e + 1 + t;

    break;
case 'n':
case 'N':

    printf("enter runs on no ball: ");

    scanf("%d", &t);

    *b = *b + t + 1;

    *d = *d + 1;

    *f = *f + 1;

    break;
case 'l':
case 'L':
```



```

printf("enter runs as leg byes: ");
scanf("%d", &t);

*b = *b + t;
*a = *a + 1;
*d = *d + t;
*g = *g + t;

break;
case 'b':
case 'B':

printf("enter runs as byes: ");
scanf("%d", &t);

*b = *b + t;
*a = *a + 1;
*d = *d + t;
*h = *h + t;

break;
case 'o':
case 'O':

*a = *a + 1;
*c = *c + 1;

printf("enter runs if wicket is runout: ");
scanf("%d", &t);

*b = *b + t;

break;
}
}
void result(int a, int b, int e, int g, char f, char c[], char d[])
{
int h, i;
h = (a - 1) - b;
i = 10 - g;

```

```

if ((a - 1) == b)
{
    printf("\t\tMATCH DRAWN.");
}
else if (a > b)
{
    if (e == 1)
    {
        if (f == 'B' || f == 'b')
            printf("\t\t%s won by %d runs\n\t\tCONGRATULATIONS TEAM %s!", c, h, c);
        else if (f == 'F' || f == 'f')
            printf("\t\t%s won by %d runs\n\t\tCONGRATULATIONS TEAM %s!", d, h, d);
        }
    else if (e == 2)
    {
        if (f == 'B' || f == 'b')
            printf("\t\t%s won by %d runs\n\t\tCONGRATULATIONS TEAM %s!", d, h, d);
        else if (f == 'F' || f == 'f')
            printf("\t\t%s won by %d runs\n\t\tCONGRATULATIONS TEAM %s!", c, h, c);
        }
    }
else if (a < b || a == b)
{
    if (e == 1)
    {
        if (f == 'B' || f == 'b')
            printf("\t\t%s won by %d wickets\n\t\tCONGRATULATIONS TEAM %s!", d, i, d);
        else if (f == 'F' || f == 'f')
            printf("\t\t%s won by %d wickets\n\t\tCONGRATULATIONS TEAM %s!", c, i, c);
        }
    else if (e == 2)

```

```

    {
        if (f == 'B' || f == 'b')
            printf("\t\t%s won by %d wickets\n\t\t\tCONGRATULATIONS TEAM %s!", c, i, c);
        else if (f == 'F' || f == 'f')
            printf("\t\t%s won by %d wickets\n\t\t\tCONGRATULATIONS TEAM %s!", d, i, d);
    }
}

void projected_score(int a, int b, int c, int d, float e)
{
    int ab, ba;
    ab = (c - b) + (d / 10);
    ba = (ab * e) + a;
    printf("\nPROJECTED SCORE:\n%d at %0.2f RPO(curr. rate)\n%d at 6 RPO\n%d at 8 RPO\n%d at 10 RPO\n", ba, e, (ab * 6) + a, (ab * 8) + a, (ab * 10) + a);
}

void runs_needed(int a, int b, int c, int d, int e)
{
    int f, h;
    f = (c * 6) - ((b * 6) + d);
    h = e - a;
    printf("%d runs needed to win from %d balls", h, f);
}

void req_rate(float a, float b, float c, float d, float e)
{
    float f, g, h, i;
    f = e - a;
    if (d == 0)
    {
        h = c;
    }
}

```

```
else
{
    h = (c - 1) + 0.6;
}
i = b + (d / 10);
g = f / (h - i);
printf("\t req. run rate %0.2f\n", g);
}
```

Match between:

india

versus

england

Toss won by: india

Enter B to choose bat first and F to field first:

main.c: In function 'string_comp':

INDIA vs ENGLAND (1-overs match)

INDIA won the toss and elected to bat first

INDIA 0-0

overs 0.0

INDIA vs ENGLAND (1-overs match)
INDIA won the toss and elected to bat first

INDIA 16-0 overs 1.0
boundaries: 3 fours,0 sixes

extras: 0(wd 0,nb 0,lb 0,b 0) boundaries: 3 fours,0 sixes

innings completed.█

INDIA vs ENGLAND (1-overs match)
INDIA gives target of 17 runs to ENGLAND to win

ENGLAND 0-0 overs 0.0

█

INDIA vs ENGLAND (1-overs match)
INDIA gives target of 17 runs to ENGLAND to win

ENGLAND 12-0 overs 1.0
boundaries: 0 fours,1 sixes

extras: 0(wd 0,nb 0,lb 0,b 0) boundaries: 0 fours,1 sixes

innings completed.█

INDIA won by 4 runs
CONGRATULATIONS TEAM INDIA!█

4.

RISK ANALYSIS

While deploying a project, several risks are always there with different probability of happening and different consequences. Here we are giving such risks for our project.

4.1 New Unproven Technology

The Cricket Management System is developed on Android Studio. One of the biggest risks involved with our project is the advancement in technology. As soon as new tools and better technology will come into play in the near future, the software might become outdated and hence would not be able to meet the user expectations. Thus, the software will not be able to compete with other better software available in the market and henceforth will ultimately fail.

4.2 Organizational Risk

Another risk involved with the Cricket Management System is that of the organization of player info, team info into various tables in the database. Even if there is the slightest possibility that there exists some ambiguity and inconsistency in the stored information about different players and teams, this would make it difficult for the users to fetch the information of their favorite players and would display wrong results too. Thus, the Software would lose its essence of correctness at the first place. In the longer run, this situation may lead to the complete removal of the Software from the user market.

4.3 Project Delay

One of the major risks involved with our project is in the project delay that if the project is not completed within the given deadline, then we have to increase the person hours on our project which will ultimately lead to an increase in the overall cost of the project. Furthermore, if the quality software is not delivered on time, it would be of no use to the costumers as well. Thus, The whole project will not serve any fruitful purpose to the users.

4.4 Risk of storage of database space

What if the company faces the shortage of database space for storing the application database? Then the CMS application will not work properly. To resolve this, the management has to redesign the database and some changes to the CMS application which will add cost of the application to the management. The probability of this to the management is less in short term and has a little higher risk in the long term.

RISK	CATEGORY	PROBABILITY (OUT OF 100%)	IMPACT (1-5)
1.Size estimate is significantly low	Product Size	40%	2
2.Large no of users	Product Size	30%	3
3.Less Reuse	Product Size	20%	2
4.Delivery Deadline NotMet	Business Impact	20%	4
5.Cost Exceeds	Cost Risk	50%	4
6.Custom RequirementChange	Product Size	45%	3
7.Freezing Cost	Business Impact	35%	2
8.Lack Of Planning	Technology	10%	3
9.Staff In Experience	Staff Size And Experience	10%	2

Table 4: Risk Assessment

RISK INFORMATION SHEET (RMMM1)

RISK ID: R01	DATE : 10-02-2018	PROBABILITY: 60%	IMPACT: High
------------------------	-----------------------------	----------------------------	------------------------

DESCRIPTION:

Organizational Risk (Inconsistent data into tables)

REFINEMENT / CONTEXT:

1. Inconsistency of data will lead to the formation of spurious tuples in the database.
2. Incorrect information will be passed on to the users about the players and teams.

MITIGATION / MONITORING:

1. Correct data must be inserted into the database tables by the Admin so that no spurious tuples are formed.
2. Cricket fans and other users can get accurate and precise information about their favorite players and teams along with the live scores.

MANAGEMENT / CONTINGENCY PLAN / TRIGGER:

1. Risk Estimation computed to be 5000. Allocate this amount within project contingency cost.
2. Develop, revise and schedule assuming that high capable server is to be adopted: allocate staff accordingly.

Trigger: Mitigation steps unproductive as of 20-02-2018

CURRENT STATUS:

05/03/2018: Mitigation steps initiated

Originator: Vrinda Anand	Assigned: Riya Goel
---------------------------------	----------------------------

RISK INFORMATION SHEET (RMMM2)

Risk ID: R02	DATE: 12-02-2018	PROBABILITY: 40%	IMPACT: AVERAGE
------------------------	----------------------------	----------------------------	---------------------------

DESCRIPTION:

New Unproven technology (Obsolete version of Android Studio)

REFINEMENT / CONTEXT:

1. The software will fail if Android becomes outdated and is replaced by some other Operating System.
2. The software will fail to provide information to the cricket fans about their favourite players.

MITIGATION/MONITORING:

The software will continue to rule the market if the developers continue to release newer versions of Android.

MANAGEMENT / CONTINGENCY PLAN/ TRIGGER:

1. Risk Estimation computed to be 40,000. Allocate this amount within project contingency cost.
2. Develop, revised schedule assuming that high capable server is to be adopted: allocate staff accordingly.

Trigger: Mitigation steps unproductive as of 22-02-2018

CURRENT STATUS:

07/03/2018: Mitigation steps initiated

Originator: Priya Aggarwal	Assigned: Sonakshi Garg
-----------------------------------	--------------------------------

6. TESTING

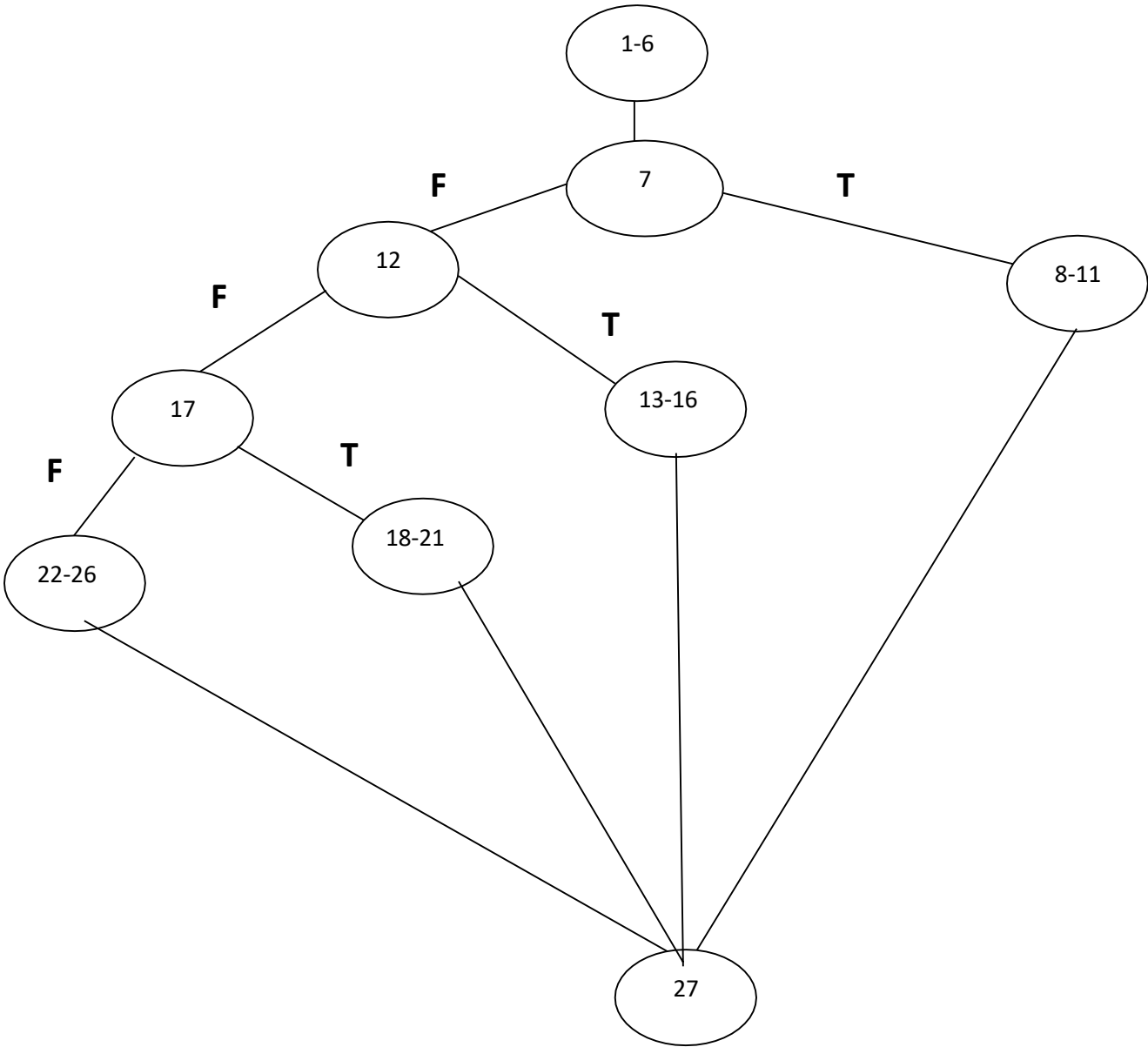
SOFTWARE TESTING

Testing is done with an objective of finding most errors with minimum amount of time and effort.

WHITE BOX testing sometimes called *glass-box testing*, is a test-case design philosophy that uses the control structure described as part of component-level design to derive test cases. Using white-box testing methods, you can derive test cases that :

- (1) Guarantee that all independent paths within a module have been exercised at least once.
- (2) Exercise all logical decisions on their true and false sides.
- (3) Execute all loops at their boundaries and within their operational bounds.
- (4) Exercise internal data structures to ensure their validity.

Basic Path Testing is a White Box testing technique that enables to derive logical complexity and defines basic test of execution paths. The test cases are prepared so that each execution path will occur at least once.



- **Cyclomatic Complexity**

Cyclomatic complexity is a software metric that provides a quantitative measure of the logical complexity of a program. When used in the context of the basis path testing method, the value computed for cyclomatic complexity defines the number of independent paths in the basis set of a program and provides you with an upper bound for the number of tests that must be conducted to ensure that all statements have been executed at least once.

$$P = 3$$

$$E = 11$$

$$N = 9$$

$$R = 4$$

WHERE P = NO OF PREDICATE NODES

E = NO. OF EDGES

N = NO. OF NODES

R = NO. OF REGIONS

$$\text{CYCLOMATIC COMPLEXITY} = E - N + 2$$

$$= 11 - 9 + 2$$

$$= 4$$

OR

$$\text{CYCLOMATIC COMPLEXITY} = P + 1$$

$$= 3 + 1$$

$$= 4$$

OR

$$\text{CYCLOMATIC COMPLEXITY} = R$$

$$= 4$$

- **To Find No. Of Independent Paths**

No. of independent paths=Cyclomatic Complexity
=4

- **To Determine Independent Paths**

1-6,7,8-11,27

1-6,7,12,13-16,27

1-6,7,12,17,18-21,27

1-6,7,12,17,22-26,27

- **Test Cases**

Independent Paths	Inputs	Expected O/P	Received O/P
1-6, 7, 8-11, 27	S.R=250,TOT_RUNS=3200	rank='H';	rank='H';
1-6, 7, 12, 13-16, 27	S.R=198,TOT_RUNS=2500	rank='A';	rank='A';
1-6, 7, 12, 17, 18-21, 27	S.R=90,TOT_RUNS=1014	rank='B';	rank='B';
1-6, 7, 12, 17, 22-26, 27	S.R=20,TOT_RUNS=500	rank='L';	rank='L';

7.

USER MANUAL

The Cricket Management System is an android application. Following are the requirement for the proper deployment of the software:

1. An android mobile phone is required.
2. The mobile phones must run on android 4.4 or higher version.
3. It has at least 250 MB of RAM and a 20 Mb of the hard disk space.
4. It must have an active internet connection for the smooth functioning of the software.

7.1 LOGIN

In this section of the project, the user has to login with the credentials of login ID and Password. Different application layout is there for the admin and other users.

For a new user, only the admin can create a new user using his password provide the login details to him/her so that no other outsider will violate the security of the application.

Users can logout from his/her id at any time using the logout home page.

7.CONCLUSION

We can conclude that the Cricket Management System can be used by people all across the globe to access the information of various players and teams to retrieve details along with the player personal details also. It can help in accessing new data and news about favourite players and teams. It is a modern approach to fetch data of players and teams, along with live scores.

It is based on advanced technology as it is compatible with all mobile sets having android version 4.4 or higher. Some of the advantages of CMS (Cricket Management System) are as follows:-

- Can ODI Tournament details on Internet.
- Players, Organizers, Selectors, Fan and Followers can access it from anywhere and anytime.
- The ODI Tournament can reach thousands of online users and gain major popularity.
- User-Friendly admin and easy to manage.
- Admin can add and edit information at any point of time and also from any location.
- Can share ODI Details through Social Media: Comments, posting, etc.
- Latest cricket news updates, ongoing ODI Events information is easy to retrieve.

Reference

- Acharya, Kamal. "STUDENT INFORMATION MANAGEMENT SYSTEM." *Authorea Preprints* (2023).
- Acharya, Kamal. "Library Management System." Available at SSRN4807104 (2019).
- ACHARYA, KAMAL, et al. "LIBRARY MANAGEMENT SYSTEM." (2019).
- Acharya, Kamal. "Online bus reservation system project report." *Authorea Preprints* (2024).
- Acharya, Kamal. "Online bus reservation system project report." (2024).
- Acharya, Kamal. "Online Bus Reservation System." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal. "Student Information Management System Project." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal. "ATTENDANCE MANAGEMENT SYSTEM." *International Research Journal of Modernization in Engineering Technology and Science* (2023): n. pag.
- Acharya, Kamal. "College Information Management System." *SSRN ElectroNIC ASIA Journal* (2024): n. pag.
- Acharya, Kamal, *Attendance Management System Project* (April 28, 2024). Available at SSRN: <https://ssrn.com/abstract=4810251> or <http://dx.doi.org/10.2139/ssrn.4810251>
- Acharya, Kamal, *Online Food Order System* (May 2, 2024). Available at SSRN: <https://ssrn.com/abstract=4814732> or <http://dx.doi.org/10.2139/ssrn.4814732>
- Acharya, Kamal, *University management system project*. (May 1, 2024). Available at SSRN: <https://ssrn.com/abstract=4814103> or <http://dx.doi.org/10.2139/ssrn.4814103>
- Acharya, Kamal, *Online banking management system*. (May 1, 2024). Available at SSRN: <https://ssrn.com/abstract=4813597> or <http://dx.doi.org/10.2139/ssrn.4813597>
- Acharya, Kamal, *Online Job Portal Management System* (May 5, 2024). Available at SSRN: <https://ssrn.com/abstract=4817534> or <http://dx.doi.org/10.2139/ssrn.4817534>
- Acharya, Kamal, *Employee leave management system*. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819626> or <http://dx.doi.org/10.2139/ssrn.4819626>
- Acharya, Kamal, *Online electricity billing project report*. (May 7, 2024). Available at SSRN: <https://ssrn.com/abstract=4819630> or <http://dx.doi.org/10.2139/ssrn.4819630>
- Acharya, Kamal, *POLICY MANAGEMENT SYSTEM PROJECT REPORT*. (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4831694> or <http://dx.doi.org/10.2139/ssrn.4831694>
- Acharya, Kamal, *Online job placement system project report*. (January 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4831638> or <http://dx.doi.org/10.2139/ssrn.4831638>

Acharya, Kamal, Software testing for project report. (May 16, 2023). Available at SSRN: <https://ssrn.com/abstract=4831028> or <http://dx.doi.org/10.2139/ssrn.4831028>

Acharya, Kamal, ONLINE CRIME REPORTING SYSTEM PROJECT. (August 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4831015> or <http://dx.doi.org/10.2139/ssrn.4831015>

Acharya, Kamal, Burber ordering system project report. (October 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4832704> or <http://dx.doi.org/10.2139/ssrn.4832704>

Acharya, Kamal, Teachers Record Management System Project Report (December 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4833821> or <http://dx.doi.org/10.2139/ssrn.4833821>

Acharya, Kamal, Dairy Management System Project Report (December 20, 2020). Available at SSRN: <https://ssrn.com/abstract=4835231> or <http://dx.doi.org/10.2139/ssrn.4835231>

Acharya, Kamal, Electrical Shop Management System Project (December 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835238> or <http://dx.doi.org/10.2139/ssrn.4835238>

Acharya, Kamal, Online book store management system project report. (February 10, 2020). Available at SSRN: <https://ssrn.com/abstract=4835277> or <http://dx.doi.org/10.2139/ssrn.4835277>

Acharya, Kamal, Paint shop management system project report. (January 10, 2019). Available at SSRN: <https://ssrn.com/abstract=4835441> or <http://dx.doi.org/10.2139/ssrn.4835441>

Acharya, Kamal, Supermarket billing system project report. (August 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4835474> or <http://dx.doi.org/10.2139/ssrn.4835474>

Acharya, Kamal, Online taxi booking system project report. (March 10, 2022). Available at SSRN: <https://ssrn.com/abstract=4837729> or <http://dx.doi.org/10.2139/ssrn.4837729>

Acharya, Kamal, Online car servicing system project report. (March 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4837832> or <http://dx.doi.org/10.2139/ssrn.4837832>

Acharya, Kamal, School management system project report. (July 10, 2021). Available at SSRN: <https://ssrn.com/abstract=4837837> or <http://dx.doi.org/10.2139/ssrn.4837837>

Acharya, Kamal, Furniture Showroom Management System Project Report (March 21, 2021). Available at SSRN: <https://ssrn.com/abstract=4839422> or <http://dx.doi.org/10.2139/ssrn.4839422>

Acharya, Kamal, Online Vehicle Rental System Project Report (March 21, 2019). Available at SSRN: <https://ssrn.com/abstract=4839429> or <http://dx.doi.org/10.2139/ssrn.4839429>

Acharya, Kamal, Fruit Shop Management System Project Report (August 10, 2023). Available at SSRN: <https://ssrn.com/abstract=4841048> or <http://dx.doi.org/10.2139/ssrn.4841048>

Acharya, Kamal, Hall Booking Management System Project Report (December 21, 2023). Available at SSRN: <https://ssrn.com/abstract=4841055> or <http://dx.doi.org/10.2139/ssrn.4841055>

Acharya, Kamal, Laundry Management System Project Report (October 21, 2023). Available at SSRN: <https://ssrn.com/abstract=4841059> or <http://dx.doi.org/10.2139/ssrn.4841059>

Acharya, Kamal, A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT (September 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4841209> or <http://dx.doi.org/10.2139/ssrn.4841209>

Acharya, Kamal, A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT (May 25, 2024). Available at SSRN: <https://ssrn.com/abstract=4841210> or <http://dx.doi.org/10.2139/ssrn.4841210>

Acharya, Kamal, ONLINE DATING MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4842066> or <http://dx.doi.org/10.2139/ssrn.4842066>

Acharya, Kamal, ONLINE RESUME BUILDER MANAGEMENT SYSTEM PROJECT REPORT. (April 25, 2021). Available at SSRN: <https://ssrn.com/abstract=4842071> or <http://dx.doi.org/10.2139/ssrn.4842071>

Acharya, Kamal, TOLL TEX MANAGEMENT SYSTEM PROJECT REPORT (August 21, 2023). Available at

SSRN: <https://ssrn.com/abstract=4842082> or <http://dx.doi.org/10.2139/ssrn.4842082>

Acharya, Kamal, *Chat Application Through Client Server Management System Project Report* (June 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4842761> or <http://dx.doi.org/10.2139/ssrn.4842761>

Acharya, Kamal, *Web Chatting Application Management System Project Report* (April 25, 2022). Available at SSRN: <https://ssrn.com/abstract=4842771> or <http://dx.doi.org/10.2139/ssrn.4842771>

Acharya, Kamal, *Automobile management system project report* (May 25, 2022). Available at SSRN: <https://ssrn.com/abstract=4846917> or <http://dx.doi.org/10.2139/ssrn.4846917>

Acharya, Kamal, *College bus management system project report* (April 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4846920> or <http://dx.doi.org/10.2139/ssrn.4846920>

Acharya, Kamal, *Courier management system project report* (May 25, 2023). Available at SSRN: <https://ssrn.com/abstract=4846922> or <http://dx.doi.org/10.2139/ssrn.4846922>

Acharya, Kamal, *Event management system project report* (April 25, 2021). Available at SSRN: <https://ssrn.com/abstract=4846927> or <http://dx.doi.org/10.2139/ssrn.4846927>

Acharya, Kamal, *Library management system project report II* (May 25, 2020). Available at SSRN: <https://ssrn.com/abstract=4848857> or <http://dx.doi.org/10.2139/ssrn.4848857>