**CRICKET APP**

**TEAM MEMBERS:**

**Anubhav Walia**

**Aditya Kashyap**

**Angad Sachdave**

**Debarati Ray**

**Toby George Sam**

**CONTENTS**

|  |  |
| --- | --- |
|  | Page No. |
| 1. Abstract | 3 |
| 1. Introduction | 4 |
| 1. Software Development Life Cycle | 5 |
| 1. High Level Design | 8 |
| 1. Low Level Design | 10 |
| 1. Conclusion | 11 |
|  |  |
|  |  |

**ABSTRACT**

It’s pretty evident that technology is accelerating at a rapid pace and humans are becoming further dependent on it for every purpose. With people getting busier and busier in their everyday lives the need for online applications to serve daily needs, be it booking a cab or keeping up with score updates of your favourite sports is increasing. This new online cricket info application provides you fast and accurate information on upcoming matches along with their dates and locations, updates on all the teams as well as current statistics of the players.

**Chapter 1**

**INTRODUCTION**

Cricket has 2.5 billion followers across the world. Aside from being the second most popular sport in the world after soccer, it is the most celebrated sport in countries like India, Sri Lanka, Bangladesh and other former British colonies.  For cricket enthusiasts, the best entertainment is to enjoy live cricket matches on their TV screens, but if they fail to see it on television, technology has built a pathway to access the online live cricket scores and other sports like Football, Hockey, Tennis, Kabaddi, etc.

Hence, online cricket score applications are gaining favour from the masses day by day. In this project a new online cricket application is developed that provides its users with detailed information on upcoming tournaments and matches, corresponding dates and locations, information on all the teams as well as current and updated performance statistics of all the players. In order to develop this application, a layered architecture is created using Spring MVC and the Java Persistence API (JPA) which is a Java specification for accessing, persisting, and managing data between Java objects / classes and a relational database. The frontend development process makes use of JPA, Spring MVC and CSS to make the display pages for validating user login credentials and subsequent display of desired cricket information after successful login. The backend development is also done using JPA to provide the necessary functionalities to the application. Both frontend and backend programs are developed as dynamic web projects on Eclipse Oxygen IDE and integrated together in the end to get the complete functioning application.

The subsequent chapters describe the software development life cycle of this online application, its high-level and low-level designs and conclude with the applications and advantages of this application.

**Chapter 2**

**SOFTWARE DEVELOPMENT LIFE CYCLE**

 The process of software development services in India goes through a series of stages in step wise fashion that almost every developing company follows. It may be required to choose the right SDLC model according to the specific concerns and requirements of the project to ensure its success.

In this project the waterfall model is followed. It is a linear sequential flow in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of software implementation. This means that any phase in the development process begins only if the previous phase is complete. The steps involved in this model of SDLC are clearly described by the following diagram:

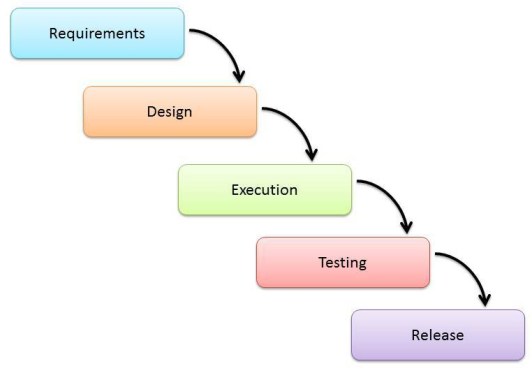


Figure 1: The Waterfall Model

1. **Requirements**

The primary requirements that this online application is expected to fulfil are listed below:

* View all upcoming cricket matches scheduled with date and location.
* Search option for list of players for each match.
* View current stats(runs, matches, wickets taken etc.) of each player as chosen.

1. **Design**

The system configuration used for this project is listed in the table below:

**Software Configuration:**

|  |  |
| --- | --- |
| Operating System | Windows 10 Pro |
| Web Server | Tomcat 8.0.53 |
| Front End | HTML, JSP |
| Scripts | Javascript |
| Server Side | Spring MVC |
| Database | SQL Plus |
| DB Connectivity | Hibernate |

The programming language used in the project is java so that configuration can be changed easily if necessary.

The design of the project primarily consists of two broad subsections:

1. Frontend design
2. Backend design

The frontend of the project involves the parts of the project that the user is going to interact with. On accessing the project url, the first page that appears is the one that asks the user to sign up or else enter the login credentials if already signed up. On successful login the user is redirected to a match schedule page that contains the list of upcoming matches along with the location and date. If the user wants information on the teams and its players he/she will have to enter the match number as displayed on the schedule. Player stats including wickets taken, runs and matches played are also displayed.

The backend of the project deals with the functionalities that do the work but the user cannot see or is unaware of. In the back end development process, a controller class named client is created that has functions to access information from the database. Separate classes for Matches, Players, player stats and Team details are created. A layered architecture is made by connecting the client class with a service package and then to a Dao package which in turn fetches the data from the database. A RowMapper class is used to map the class path instead of using an xml file to do so.

1. **Execution**

Once the front end and back end parts of the project are ready and functioning they are integrated together to create the entire project. Then comes the execution stage where the project is put through a pilot study to check if it is functioning properly.

1. **Testing**

The testing stage assesses the software for errors and documents bugs if there are any.

1. **Release**

After the necessary testing and debugging processes are complete the application is released for use by the client.

**Chapter 3**

**HIGH LEVEL DESIGN**

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

A brief description of each module involved in the design of the project is given below:

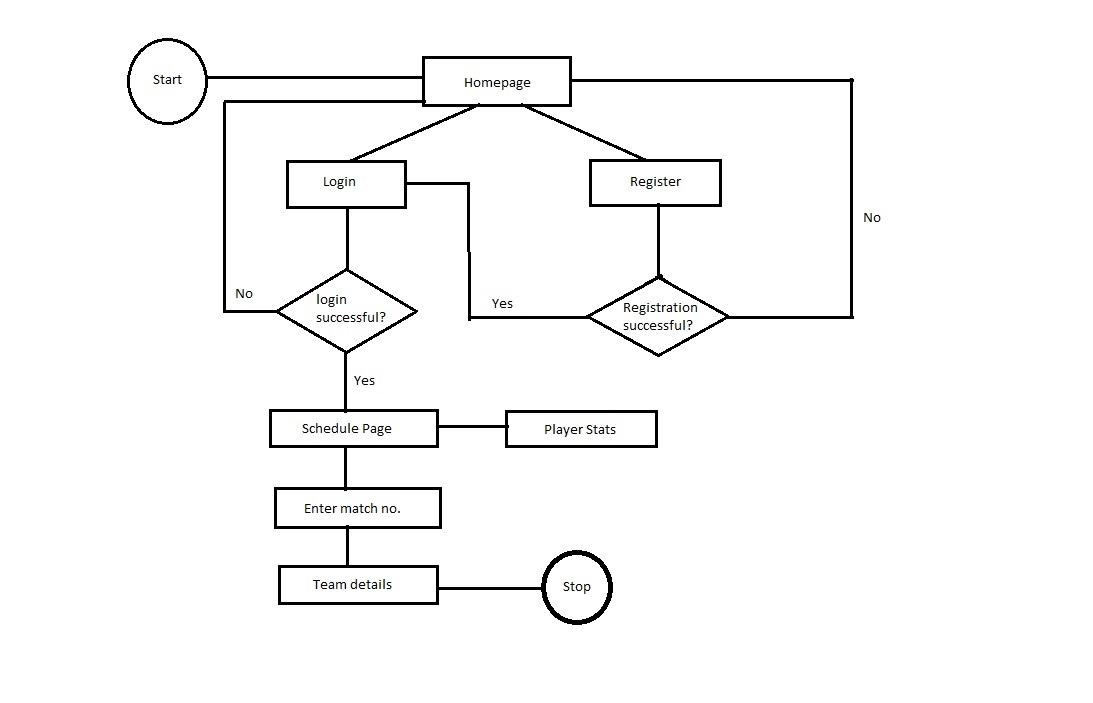
****

Figure 2: Flow diagram of front-end design

**1. Home Page**

This module will display an introductory message and validate login credentials of the user. A new user will be asked to register or else provide login credentials if already signed up. On successful login the user will be redirected to the schedule page. In case of unsuccessful login or registration failure, the user is taken back to the homepage.

**2. Match Schedule**

This page displays the schedule of upcoming matches along with their dates and allotted locations. In case the user wants more information regarding a team or player he/she will have to enter the match number as displayed on the match schedule in the search box at the bottom of the page.

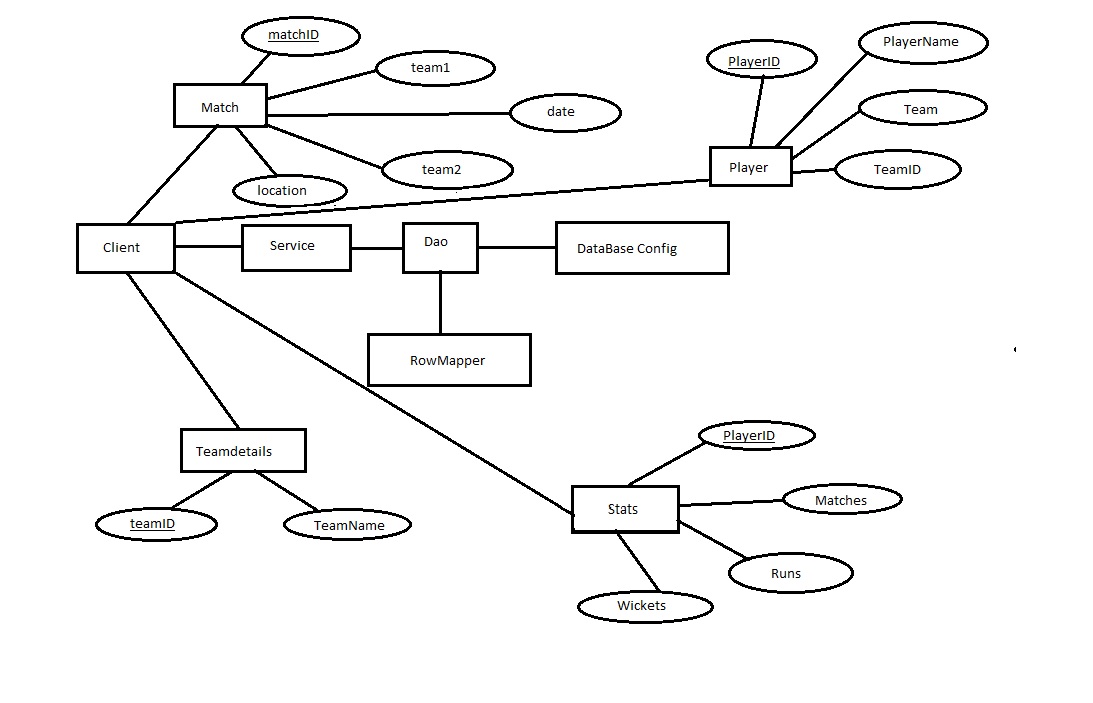
**3. Teams**

When the user enters a match number in the search box, information regarding the teams playing against each other in that match is displayed.

1. **Players**

Player statistics like runs, matches played and wickets taken associated with each player are displayed.

A class diagram of all the entity classes involved in the back end development process is given below:

****Figure 3: Class diagram of backend design.

**Chapter 4**

**LOW LEVEL DESIGN**

The Low Level Design (LLD) phase is the stage where the actual software components are designed.

The classes along with user defined methods and variables are described by the diagram below:

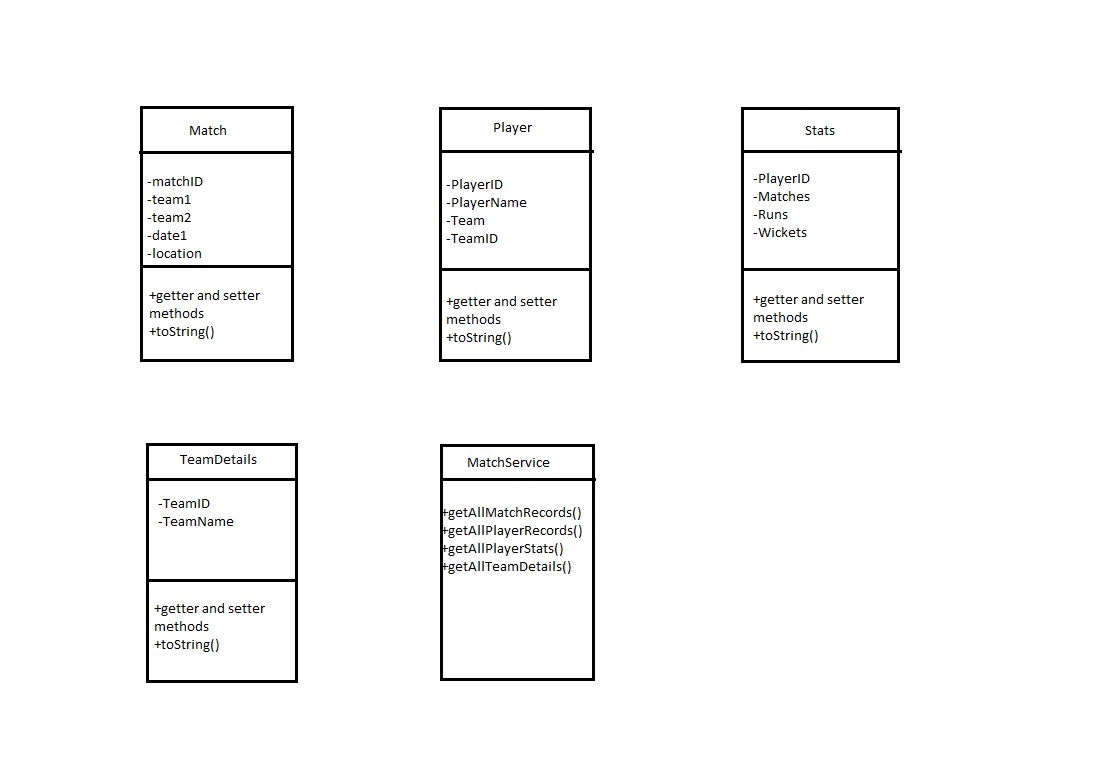


Figure 4: Class UML diagram of backend design

**ASSOCIATED CODE**

**Database creation:**

create table matchentity

( MatchID number(3) PRIMARY KEY,

Team1 varchar2(15),

Team2 varchar2(15),

date1 varchar2(15),

location varchar2(10));

create table teamdetails

( TeamID varchar2(10),

TeamName varchar2(15));

create table player(

PlayerID varchar2(10),

PlayerName varchar2(25),

Team varchar2(15),

TeamID varchar2(10));

create table stats(

PlayerID varchar2(10),

Matches number(5),

Runs number(5),

Wickets number(5));

**Chapter 5**

**CONCLUSION**

This project has been developed by the collaboration of five members namely Anubhav Walia, Aditya Kashyap, Angad Sachdave, Debarati Ray and Toby George Sam. The front end development team consisted of Anubhav, Aditya and Toby. The backend development team consisted of Angad and Debarati.

Individual contributions of the team members are listed below:

1. Backend(layered architecture using Spring JPA) - Angad
2. Frontend(layered architecture using Spring MVC) - Aditya
3. CSS for styling HTML pages - Toby
4. Database structure and command lines - Anubhav
5. Final project report consisting of HLD and LLD - Debarati