FOOTBALL LEAGUE DATABASE SYSTEM

1. **TEAM MEMBERS**

1. Ajay Hegde (Roll No. 29 and Reg. No. 150905126)

2. Shashank Kunikullaya (Roll No. 32 and Reg. No. 150905132)

**2. ABSTRACT**

We aim to create a Football League Database Management System to manage the

Statistics of a football league. It will include two primary users, Administrators and followers(fans) of the league.

The Administrator will have complete access and control over the Database. He/She will have the authority to manipulate and modify the data of the database.

The other users i.e. the followers will only be authorized to view the data regarding the various teams they are following.

The league DBMS will be modeled on statistics based on a real-world football league.

Team names, Fixtures, Players, Number of games played, Goals for and against etc are some of the attributes used in the model. These attributes will be updated by the administrator as the league progresses.

**3. PROBLEM STATEMENT**

We designed a football league database system that will allow the fans to view the statistics of a football league and compare the performance of various teams playing in the league. The administrator of the DBS can delete, insert and update the league info using an easy to use Swings GUI.

**FUNCTIONAL REQUIREMENTS**

Java Environment - Java SE Development Kit 8

JDBC connectivity for MySQL

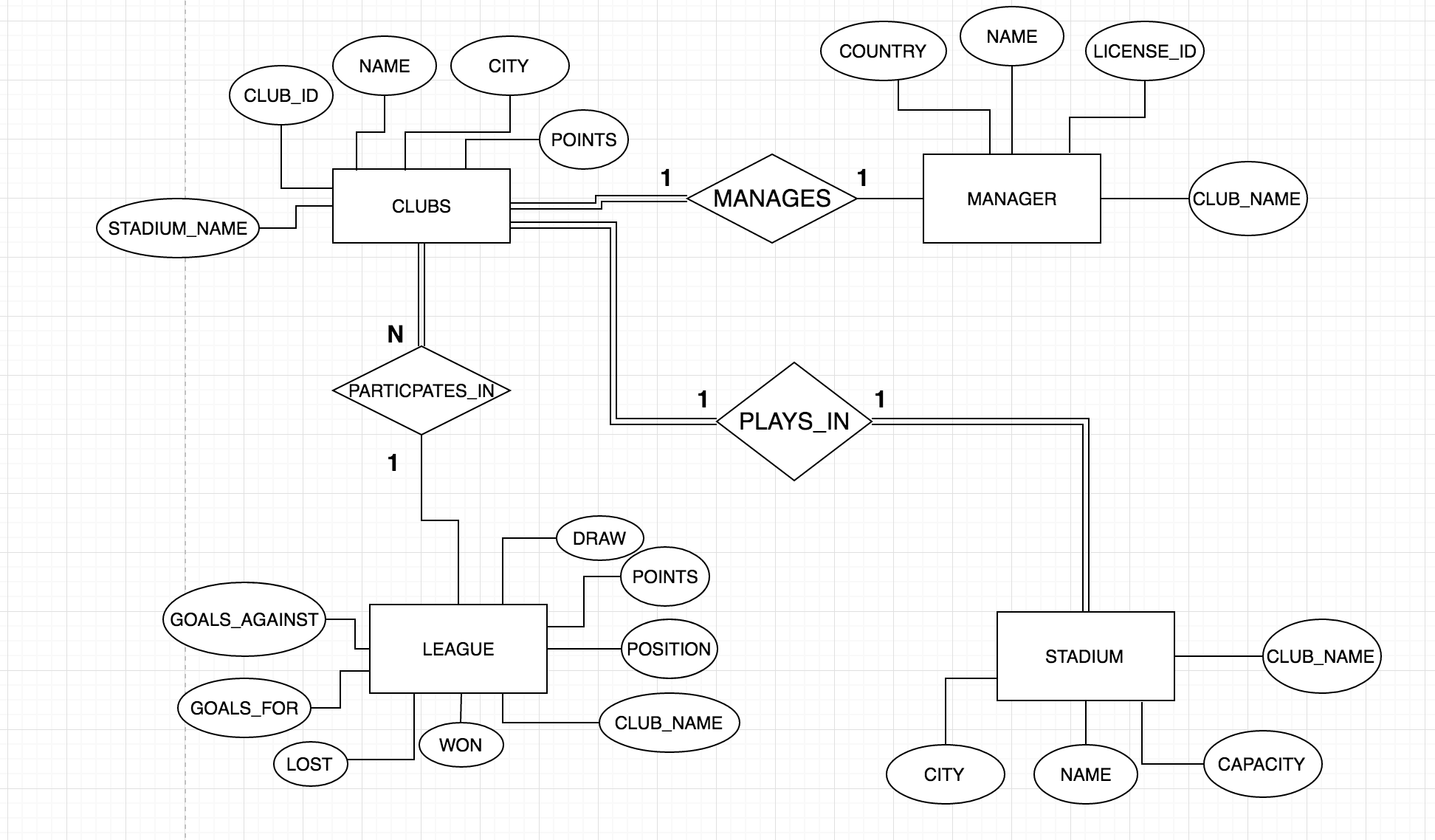
PhpMyAdmin for easy UI for tables

**System Requirements**:   
Operating System: Win 7 or higher

RAM- 4GB or higher

Processor- Intel Core i3 or higher

**4. ER DIAGRAM**

****

**5. RELATIONAL TABLES**

**5.1.CLUBS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CLUB\_ID | NAME | CITY | STADIUM\_NAME | POSITION |
| 1000 | SOUTHHALL FC | LONDON | TRAFALGAR PARK | 1 |
| 1001 | DORTMUND FC | BERLIN | HANS ARENA | 2 |
| 1002 | JUVENTUS | MILAN | EZIO ARENA | 3 |
| 1003 | LA GALAXY | LOS ANGELES | TAPLES CENTRE | 4 |
| 1004 | LIVERPOOL FC | LIVERPOOL | STEINSGATE | 5 |
| 1005 | CRYSTAL PALACE | NORWOOD | SELHURST PARK | 6 |
| 1006 | BLACKBURN FC | BLACKFRIARS | KENSINGTON | 7 |
| 1007 | WELLINGTON FC | NEW HAMPSHIRE | REDMUND PARK | 8 |
| 1008 | GONZAGA UNITED | SPOKANE | CASTLE ARENA | 9 |
| 1009 | COMPTON UNITED | COMPTON | MADISON PARK | 10 |

**5.2.LEAGUE**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| POSITION | CLUB\_NAME | WON | LOST | DRAW | GOALS\_FOR | GOALS\_AGAINST | POINTS |
| 1 | SOUTHHALL FC | 9 | 1 | 0 | 34 | 14 | 27 |
| 2 | DORTMUND FC | 8 | 1 | 1 | 32 | 15 | 25 |
| 3 | JUVENTUS | 8 | 2 | 0 | 31 | 16 | 24 |
| 4 | LA GALAXY | 7 | 2 | 1 | 29 | 16 | 22 |
| 5 | LIVERPOOL FC | 6 | 2 | 2 | 27 | 18 | 20 |
| 6 | CRYSTAL PALACE | 5 | 2 | 3 | 25 | 17 | 18 |
| 7 | BLACKBURN FC | 5 | 3 | 2 | 24 | 19 | 17 |
| 8 | WELLINGTON FC | 4 | 2 | 4 | 22 | 18 | 16 |
| 9 | GONZAGA UNITED | 3 | 6 | 1 | 20 | 22 | 10 |
| 10 | COMPTON UNITED | 2 | 6 | 2 | 18 | 24 | 8 |

**5.3.LOGIN**

|  |  |  |
| --- | --- | --- |
| ID | USERNAME | PASSWORD |
| 1000 | ajayhegde | root |
| 1001 | shashank | root |
| 1002 | Admin | root |

**5.4.MANAGER**

|  |  |  |  |
| --- | --- | --- | --- |
| LICENSE\_ID | NAME | COUNTRY | CLUB\_NAME |
| 1000 | EDWARD | AMERICA | SOUTHHALL FC |
| 1001 | ROBERTSON | ENGLAND | DORTMUND FC |
| 1002 | MOURINHO | SPAIN | JUVENTUS |
| 1003 | FERGUSON | SCOTLAND | LA GALAXY |
| 1004 | RIVERS | ITALY | LIVERPOOL FC |
| 1005 | KERR | FRANCE | CRYSTAL PALACE |
| 1006 | KLOPP | GERMANY | BLACKBURN FC |
| 1007 | RASHID | DUBAI | WELLINGTON FC |
| 1008 | GOLOVKIN | TURKEY | GONZAGA UNITED |
| 1009 | MCGREGOR | IRELAND | COMPTON UNITED |

**5.5.STADIUM**

|  |  |  |  |
| --- | --- | --- | --- |
| CLUB\_NAME | NAME | CAPACITY | CITY |
| SOUTHHALL FC | TRAFALGAR PARK | 68000 | LONDON |
| DORTMUND FC | HANS ARENA | 72000 | BERLIN |
| JUVENTUS | EZIO ARENA | 70000 | MILAN |
| LA GALAXY | TAPLES CENTRE | 65000 | LOS ANGELES |
| LIVERPOOL FC | STEINSGATE | 66000 | LIVERPOOL |
| CRYSTAL PALACE | SELHURST PARK | 62000 | NORWOOD |
| BLACKBURN FC | KENSINGTON | 58000 | BLACKFRIARS |
| WELLINGTON FC | REDMUND PARK | 55000 | NEW HAMPSHIRE |
| GONZAGA UNITED | CASTLE ARENA | 52000 | SPOKANE |
| COMPTON UNITED | MADISON PARK | 48000 | COMPTON |

**6. DCL COMMANDS**

**6.1.CREATION**

* CREATE TABLE `clubs` (

`CLUB\_ID` varchar(5) NOT NULL,

`NAME` varchar(15) NOT NULL,

`CITY` varchar(15) NOT NULL,

`STADIUM\_NAME` varchar(15) NOT NULL,

`POSITION` int(2) NOT NULL

)

* CREATE TABLE `league` (

`POSITION` int(2) NOT NULL,

`CLUB\_NAME` varchar(15) NOT NULL,

`WON` int(2) NOT NULL,

`LOST` int(2) NOT NULL,

`DRAW` int(2) NOT NULL,

`GOALS\_FOR` int(2) NOT NULL,

`GOALS\_AGAINST` int(2) NOT NULL,

`POINTS` int(2) NOT NULL

)

* CREATE TABLE `login` (

`ID` varchar(4) NOT NULL,

`USERNAME` varchar(10) NOT NULL,

`PASSWORD` varchar(10) NOT NULL

)

* CREATE TABLE `manager` (

`LICENSE\_ID` varchar(5) NOT NULL,

`NAME` varchar(10) NOT NULL,

`COUNTRY` varchar(10) NOT NULL,

`CLUB\_NAME` varchar(15) NOT NULL

)

* CREATE TABLE `stadium` (

`CLUB\_NAME` varchar(15) NOT NULL,

`NAME` varchar(15) NOT NULL,

`CAPACITY` int(6) NOT NULL,

`CITY` varchar(15) NOT NULL

)

**6.2.KEYS**

* ALTER TABLE `clubs`

ADD PRIMARY KEY (`CLUB\_ID`),

ADD KEY `NAME` (`NAME`),

ADD KEY `POSITION` (`POSITION`),

ADD KEY `STADIUM\_NAME` (`STADIUM\_NAME`),

ADD KEY `CITY` (`CITY`);

* ALTER TABLE `league`

ADD PRIMARY KEY (`CLUB\_NAME`),

ADD KEY `POSITION` (`POSITION`);

* ALTER TABLE `login`

ADD PRIMARY KEY (`ID`);

* ALTER TABLE `manager`

ADD PRIMARY KEY (`LICENSE\_ID`),

ADD KEY `CLUB\_NAME` (`CLUB\_NAME`);

* ALTER TABLE `stadium`

ADD PRIMARY KEY (`NAME`),

ADD KEY `CLUB\_NAME` (`CLUB\_NAME`),

ADD KEY `CITY` (`CITY`);

**6.3.CONSTRAINTS**

* ALTER TABLE `clubs`

ADD CONSTRAINT `fk2` FOREIGN KEY (`NAME`) REFERENCES `league` (`CLUB\_NAME`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `fk3` FOREIGN KEY (`POSITION`) REFERENCES `league` (`POSITION`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `fk4` FOREIGN KEY (`CITY`) REFERENCES `stadium` (`CITY`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `fk5` FOREIGN KEY (`STADIUM\_NAME`) REFERENCES `stadium` (`NAME`) ON DELETE CASCADE ON UPDATE CASCADE;

* ALTER TABLE `stadium`

ADD CONSTRAINT `FK1` FOREIGN KEY (`CLUB\_NAME`) REFERENCES `league` (`CLUB\_NAME`) ON DELETE CASCADE ON UPDATE CASCADE;

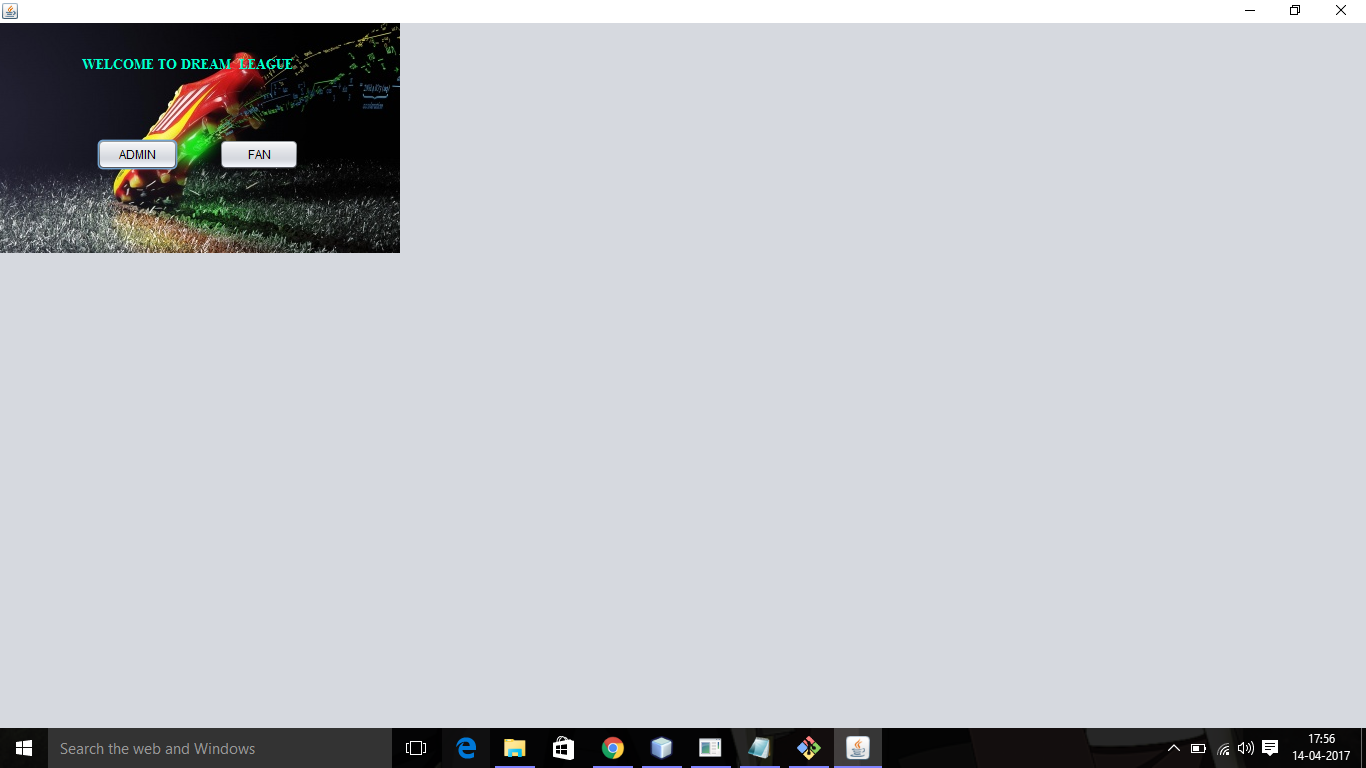
**6.4.QUERIES**

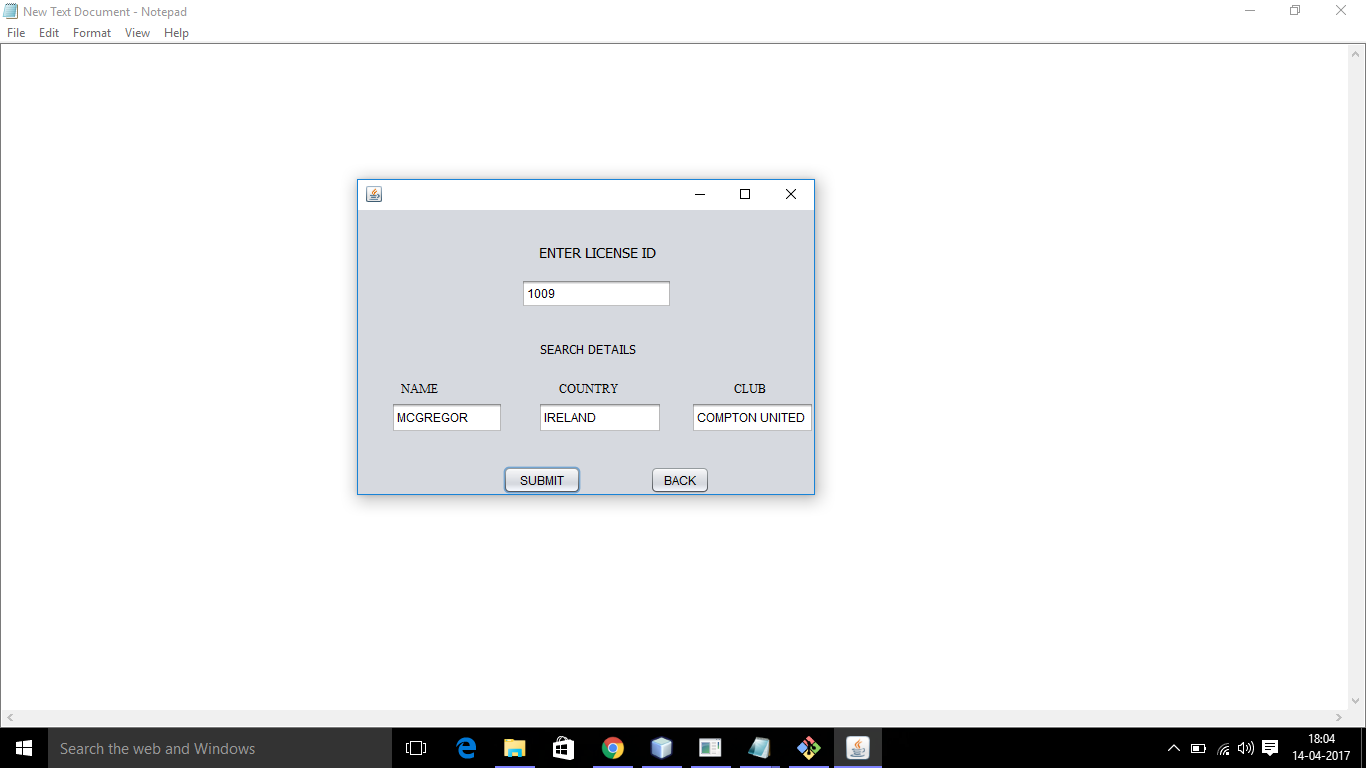
* select \* from league;
* select \* from clubs;
* select \* from stadium;
* select \* from manager;
* select \* from login;
* select \* from league order by position;
* insert into login values('1002','admin','root');
* insert into league values(1,'SOUTHHALL FC',9,1,0,34,14,27);
* insert into stadium values('SOUTHHALL FC','TRAFALGAR PARK',68000,'LONDON');
* insert into manager values('1000','EDWARD','AMERICA','SOUTHHALL FC');
* insert into clubs values('1000','SOUTHHALL FC','LONDON','TRAFALGAR PARK',1);
* select \* from manager where license\_id="1001";
* select \* from league where club\_name="MANCHESTER";
* update league set position=1,won=9,lost=1,draw=0,goals\_for=34,goals\_against=14,points=27 where club\_name="SOUTHHALL FC";
* update clubs set position=2 where name="DORTMUND FC";
* select \* from clubs c,stadium s where c.name=s.club\_name and c.club\_id="+tf.getText();
* delete from manager where license\_id="1001";
* update login set password='"+pw.getText()+"'where username=‘"+tf1.getText()+"'";

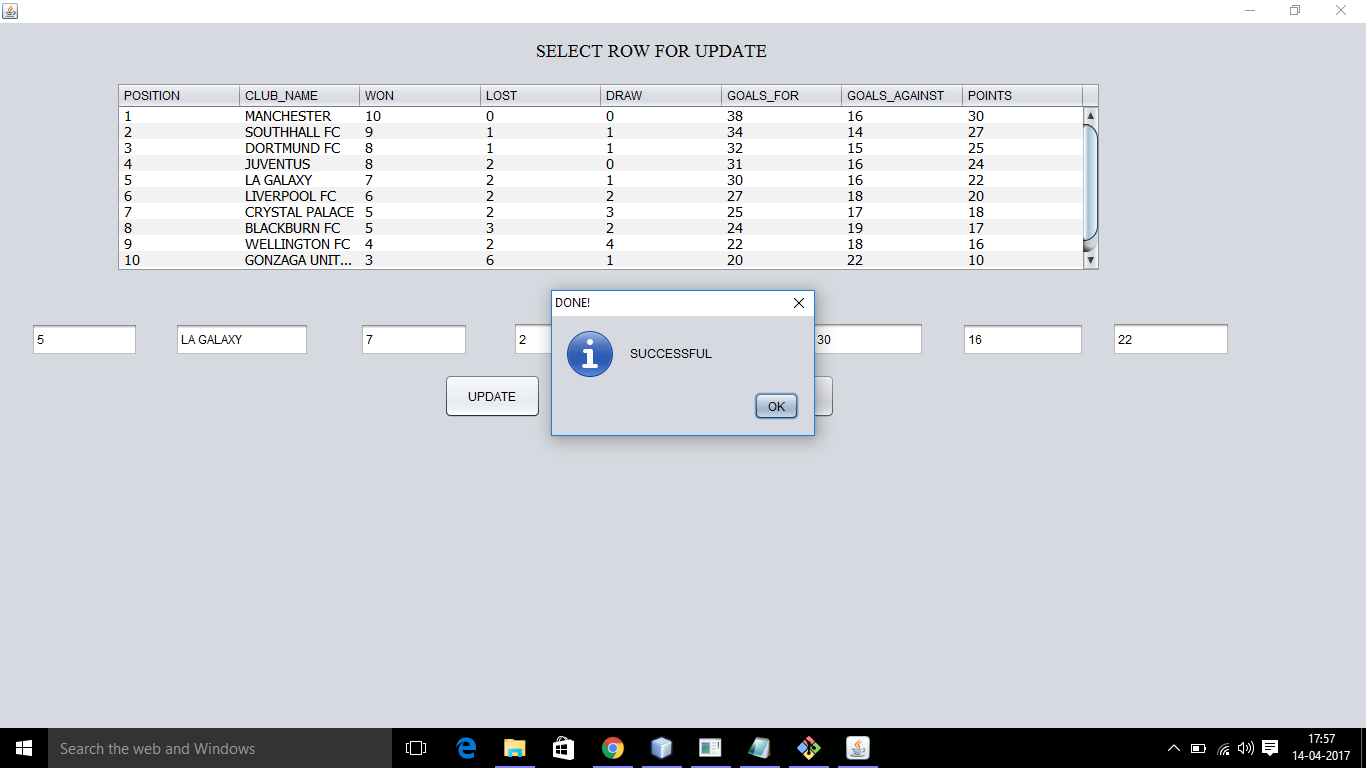
**7. UI DESIGN**

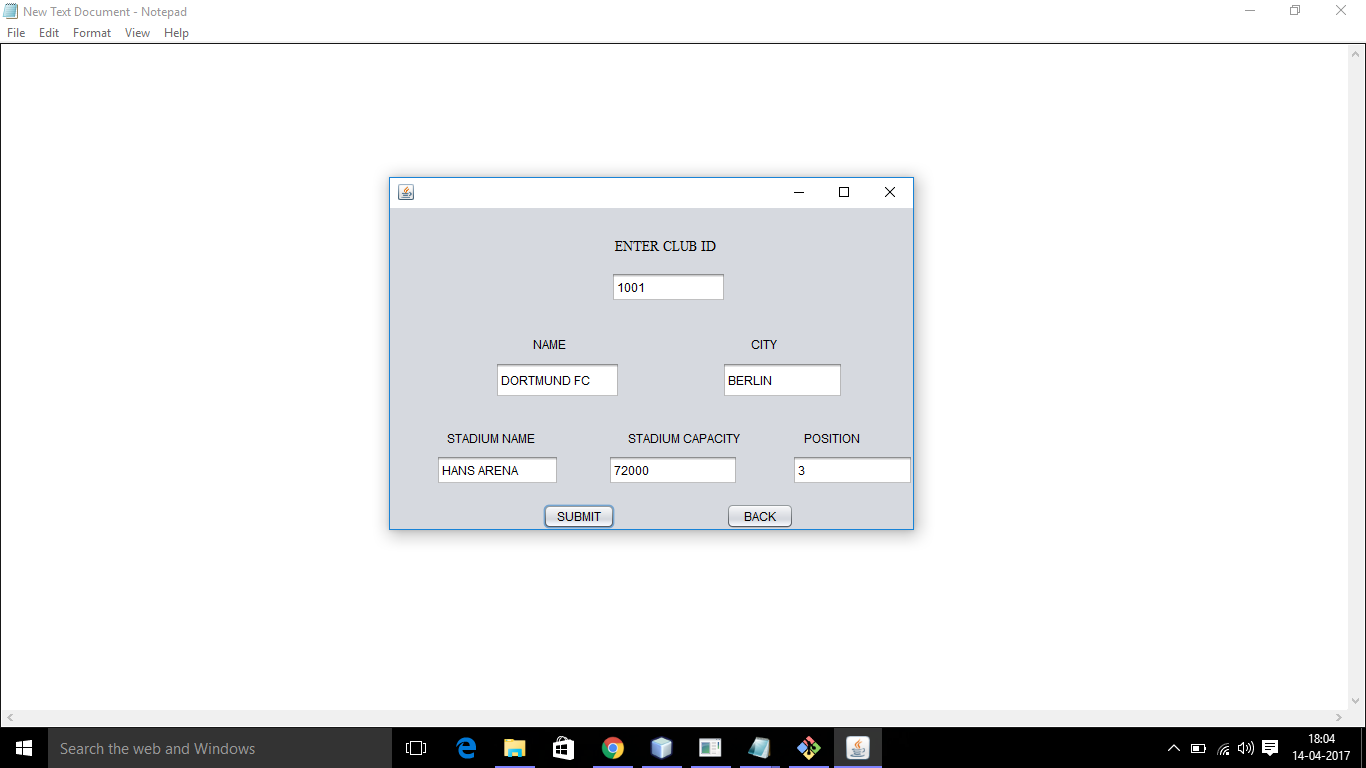
The UI of the project will be designed using Java. The swing feature in Java is an important tool for creating Graphical User Interfaces (GUI).

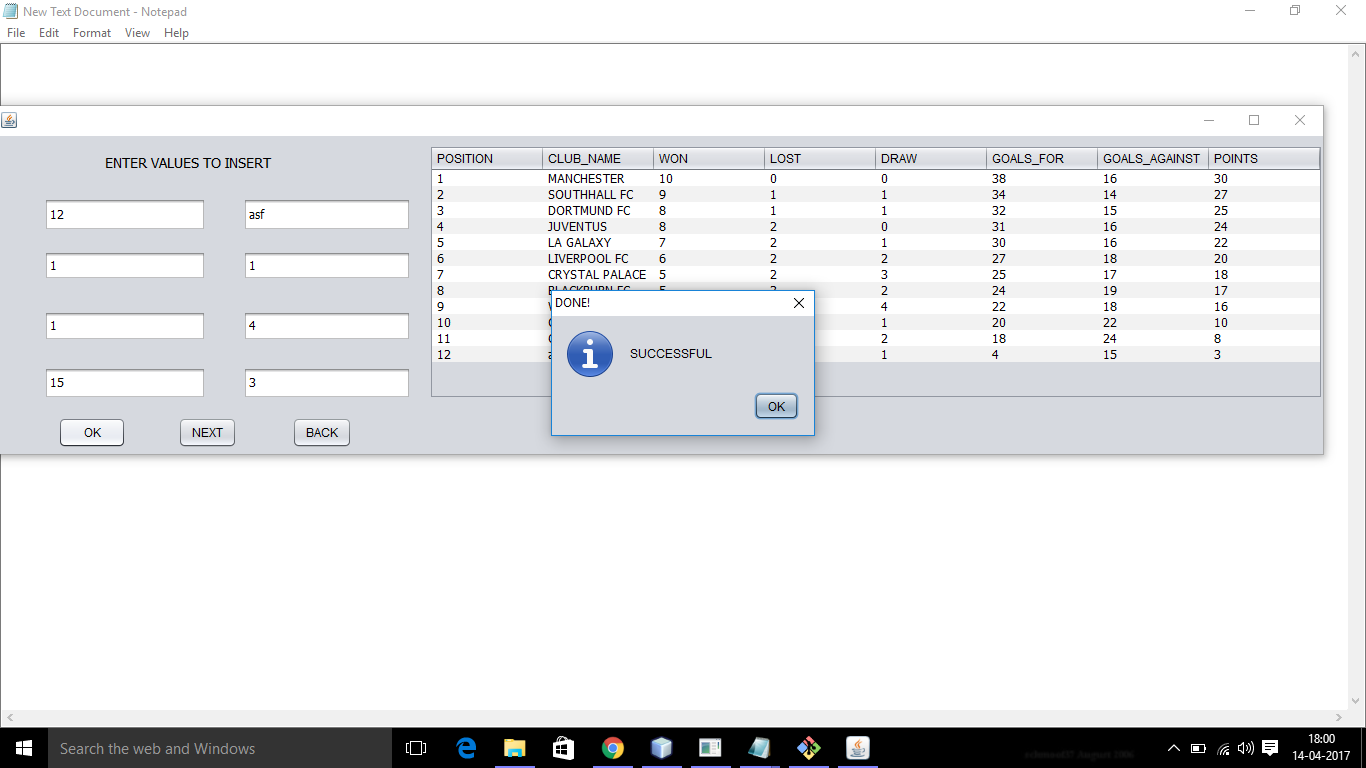
**Screenshots:**

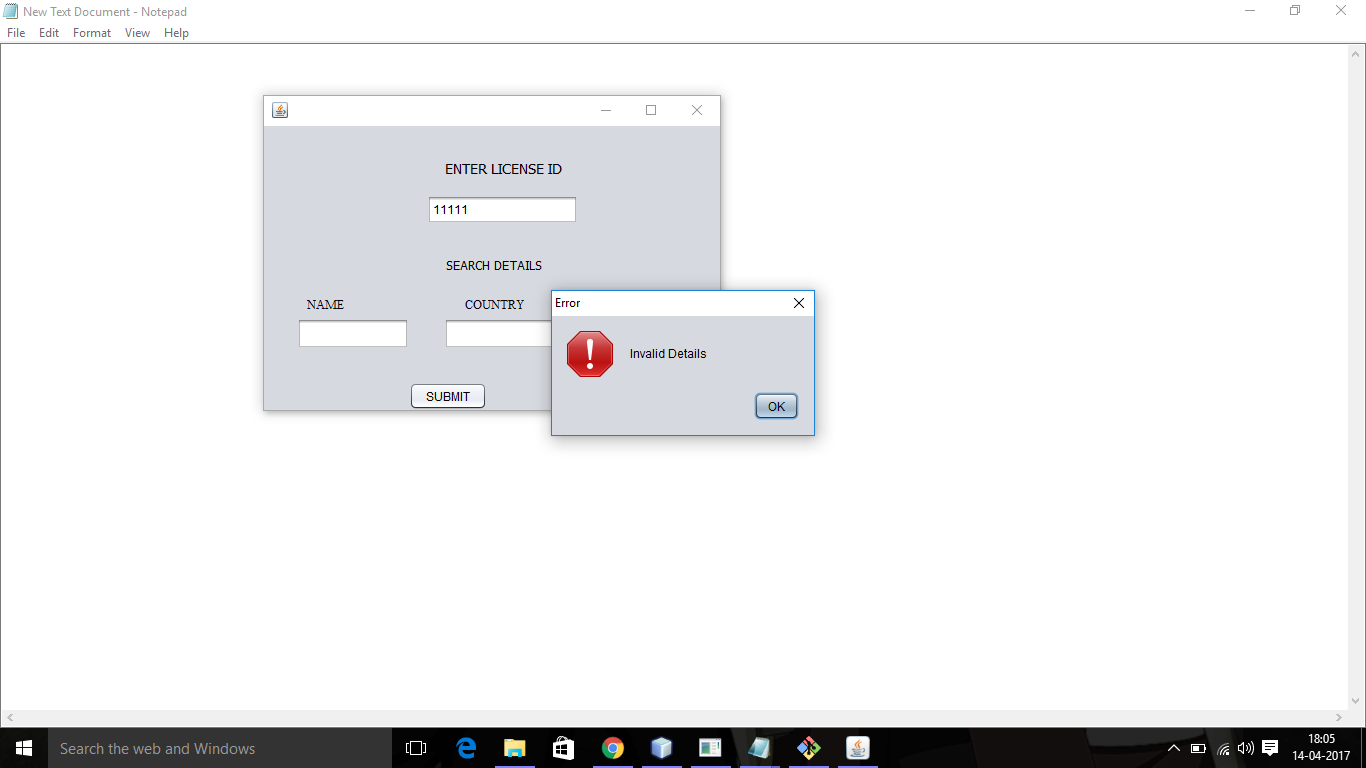
****



****

****

****

****

**8. SAMPLE CODE**

package football\_league;

import java.sql.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

import static javafx.application.Platform.exit;

import javax.swing.JOptionPane;

public class loginup extends javax.swing.JFrame {

public String s2;

Connection con=null;

Statement s=null;

ResultSet rs=null;

public loginup() {

initComponents();

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

try {

con=DriverManager.getConnection("jdbc:mysql://localhost:3306/football\_league","ajayhegde","root");

s=con.createStatement();

s.executeUpdate("update login set password='"+pw.getText()+"'where username='"+tf1.getText()+"'");

Admin a=new Admin();

this.setVisible(false);

a.setVisible(true);

} catch (SQLException ex) {

Logger.getLogger(loginup.class.getName()).log(Level.SEVERE, null, ex);

exit();

}

JOptionPane.showMessageDialog(null,"SUCCESSFUL","DONE!",JOptionPane.INFORMATION\_MESSAGE);

}

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(loginup.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(loginup.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(loginup.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(loginup.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new loginup().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPasswordField pw;

private javax.swing.JTextField tf1;

**9. REFERENCES**

1. Silberschatz, North, Sudarshan., “Database System Concepts”, McGrawHill, 6th Edition, 2011

2. [www.w3schools.com/sql](http://www.w3schools.com/sql)

3. [www.tutorialspoint.com/sql](http://www.tutorialspoint.com/sql)

4. <https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html>

5. <https://netbeans.org/kb/>