OOP PROJECT REPORT

Course Teacher: Sir Umer Farooq

Ramisha Faheem **CT-005**
Adil Aslam **CT-023**
Rija Asif Butt **CT-031**

Group Name: WinRAR

INTRODUCTION

This project is best for the ones who love cricket and those who dream of managing their own team. It provides you with the league fixtures, player's profile, live score at one place. But its main feature is when it lets you manage your own fantasy team. It gets more interesting when you get to know that you can select your own players. The points will be rewarded according to the player's performance in real matches. Climb at the top of the table by gaining points according to the rules. This fantasy team lets you to be the coach of your own team. A cricket fan will definitely go boom boom after he/she sees this.

PROBLEM AREAS

Almost every cricket fan wants a place where he can get all the cricket stuff gathered at one place. We present to you all the cricket tournament stuff gathered at a single place, where you will just have to login and the next moment you're at your desired place. When you are a die-hard cricket fan, somewhere inside, you really wish to be the part of the game too. On a small scale, our project makes that wish come true. The people who dream of managing their own teams will be able to play like their dreams.

METHODOLOGY

The project starts with the main page which provides you with different options of Play, Fixtures, Score, Players.

- The Play option starts with a login page where you are required to enter your email and password to proceed to the next page. After login you create your own team. Create your own team option first asks you about your team formation and then lets you select your team Batsman(s), Bowler(s), All-rounder(s) and Wicket Keeper. You can also view your team from My team option plus you can calculate your points using calculate points option from the menu.
- The Fixture provides you with the schedule of the whole league/tournament.
- The Score is divided into two parts or is inherited by two classes which are Live Score and Results. Live Score shows you the score of the league match that is currently live. While the Results show you the result of the recent matches played in the tournament.
- Players option provides you with the list of the players playing in the tournament with their stats.

REASON

1- All OOP Concepts
2- Pakistan Super League Craze
3- Current affairs
4- Not mainstream

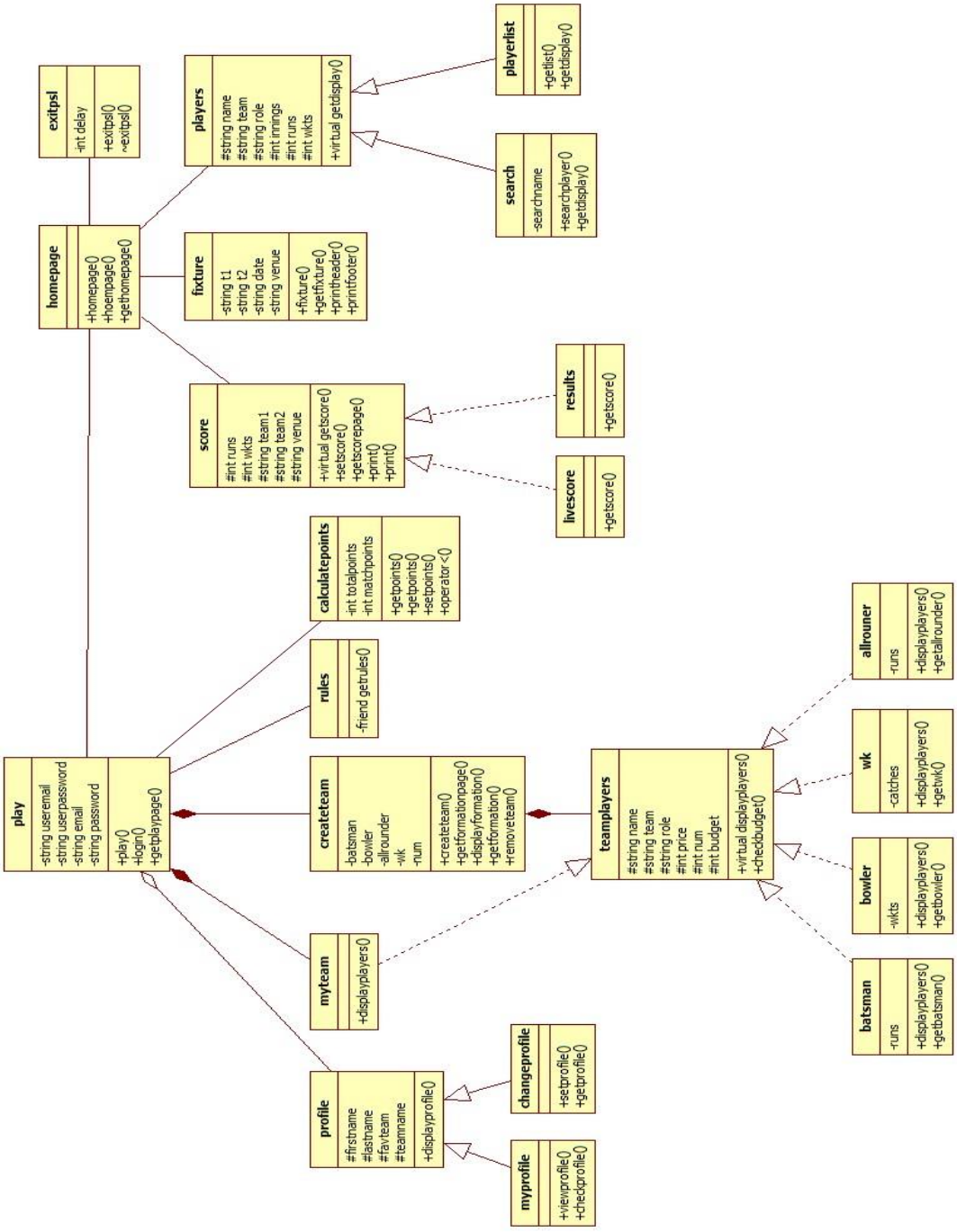
ATTRIBUTES

PLAY
-Profile
-Create Team
-My Team
-Rules
-Calculate Points
SCORE
-Live Score
-Results
FIXTURES
TEAM & PLAYERS
-Players List
-Search Players

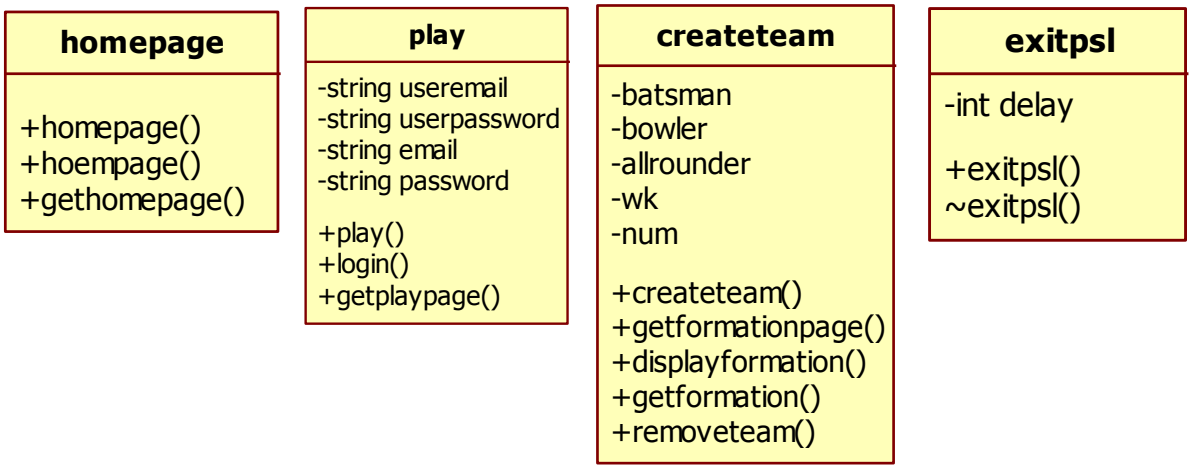
OOP CONCEPTS USED

- ✓ Constructor
- ✓ Destructor
- ✓ Encapsulation
- ✓ Abstraction
- ✓ Inheritance
- ✓ Polymorphism
- ✓ Overriding
- ✓ Interface
- ✓ Function Overloading
- ✓ Operator Overloading
- ✓ Constructor Overloading
- ✓ Friend Function
- ✓ STL Library

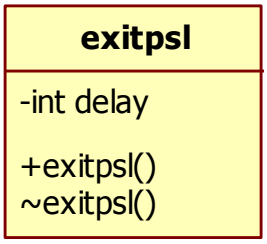
UML DIAGRAM


exitpsl
-int delay
+exitpsl()
~exitpsl()
players
#string name
#string team
#string role
#int innings
#int runs
#int wkts
+virtual getdisplay()
playerlist
+getlist()
+getdisplay()
search
-searchname
+searchplayer()
+getdisplay()
homepage
+homepage()
+hoempage()
+gethomepage()
fixture
-string t1
-string t2
-string date
-string venue
+fixture()
+getfixture()
+printheader()
+printfooter()
score
#int runs
#int wkts
#string team1
#string team2
#string venue
+virtual getscore()
+setscore()
+getscorepage()
+print()
+print()
results
+getscore()
livescore
+getscore()
calculatepoints
-int totalpoints
-int matchpoints
+getpoints()
+getpoints()
+setpoints()
+operator<()
rules
-friend getrules()
play
-string useremail
-string userpassword
-string email
-string password
+play()
+login()
+getplaypage()
createteam
-batsman
-bowler
-allrounder
-wk
-num
+createteam()
+getformationpage()
+displayformation()
+getformation()
+removeteam()
teamplayers
#string name
#string team
#string role
#int price
#int num
#int budget
+virtual displayplayers()
+checkbudget()
allrouner
-runs
+displayplayers()
+getallrounder()
wk
-catches
+displayplayers()
+getwk()
bowler
-wkts
+displayplayers()
+getbowler()
batsman
-runs
+displayplayers()
+getbatsman()
myteam
+displayplayers()
profile
#firstname
#lastname
#favteam
#teamname
+displayprofile()
changeprofile
+setprofile()
+getprofile()
myprofile
+viewprofile()
+checkprofile()

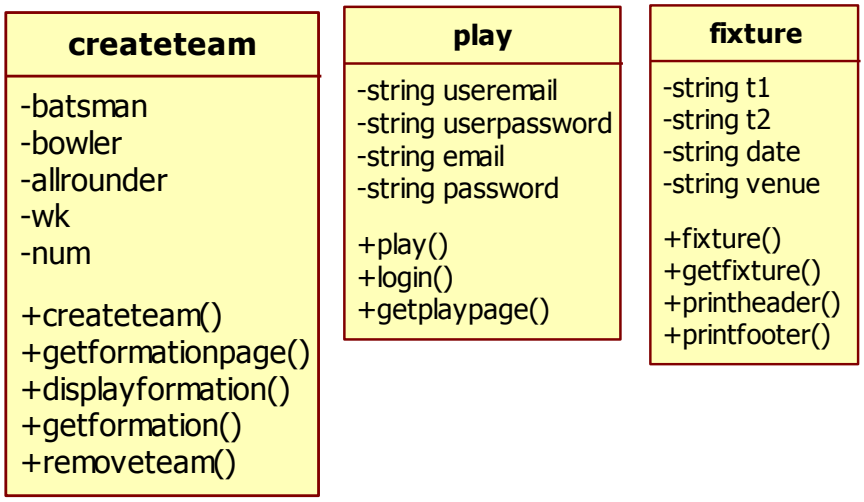
Constructor



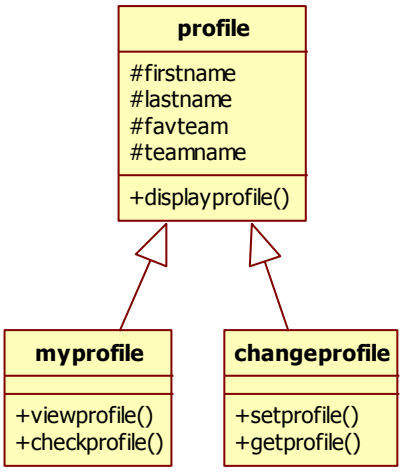
Destructor



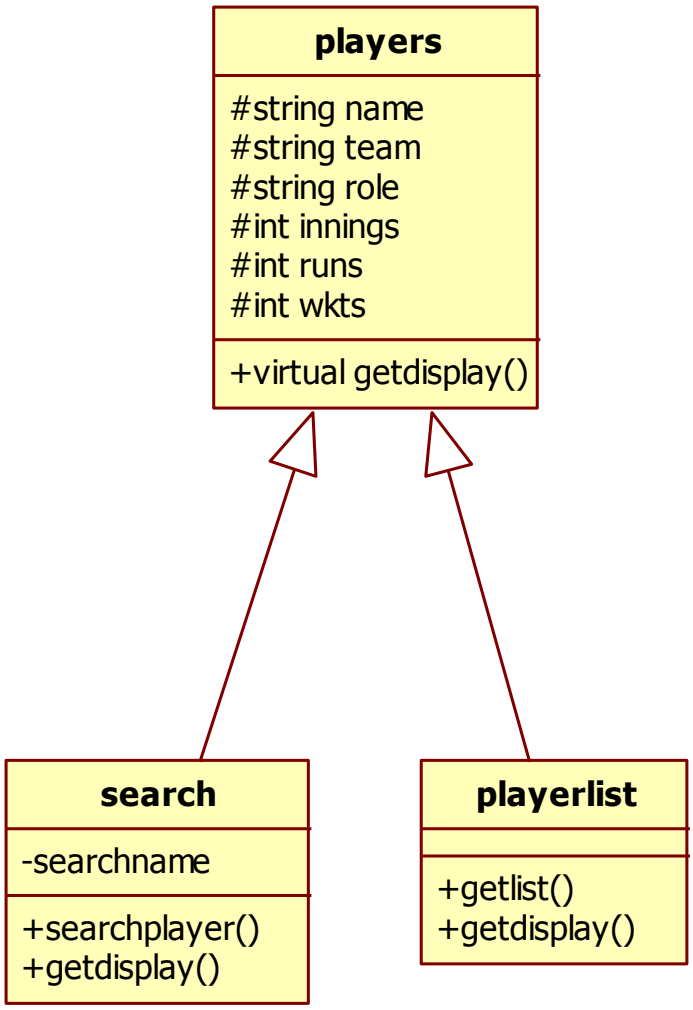
Abstraction & Encapsulation



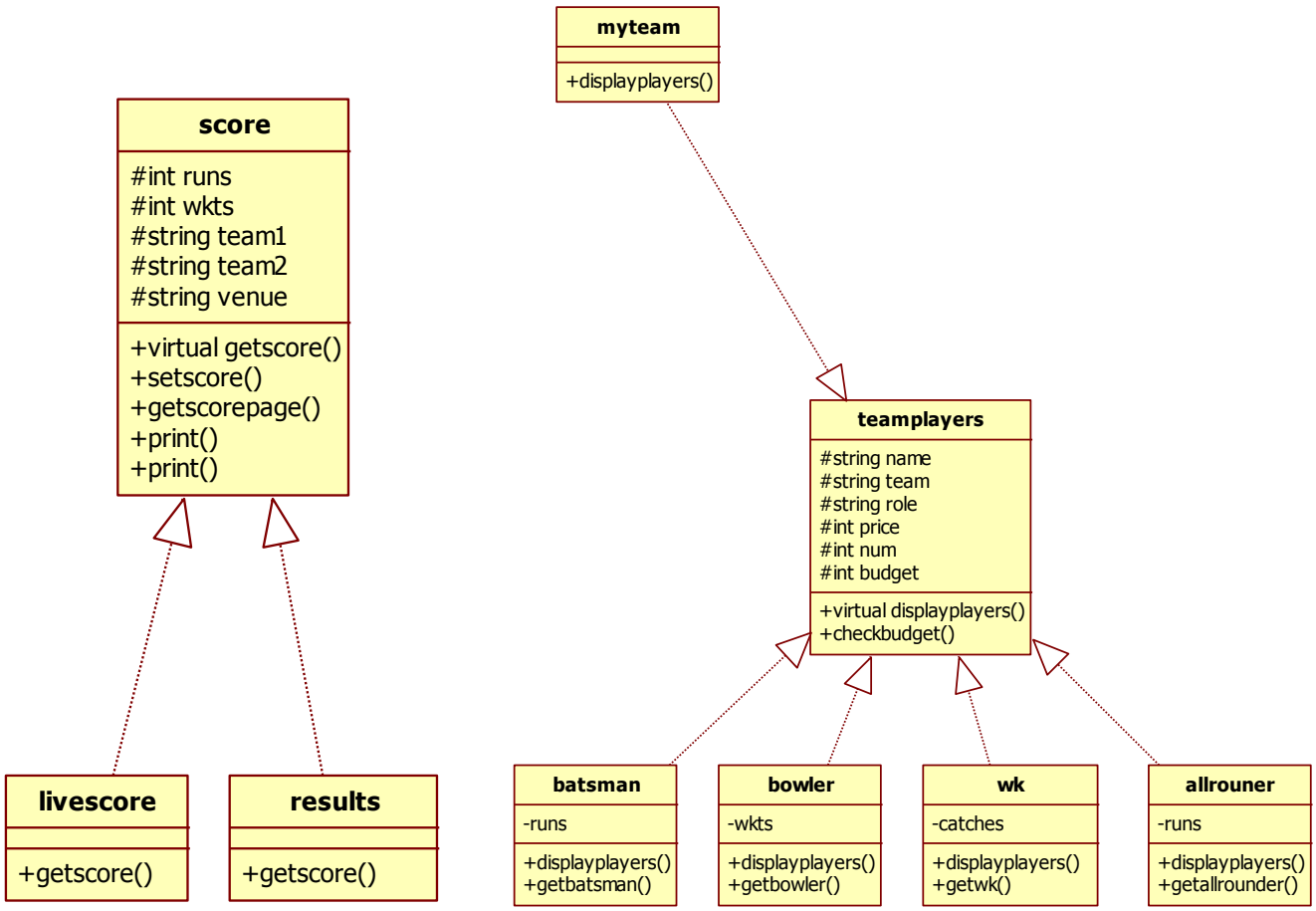
Inheritance



Polymorphism & Overriding

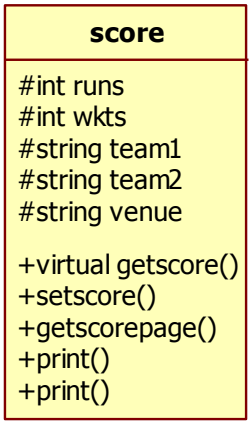


Interface


myteam
+displayplayers()
score
#int runs
#int wkts
#string team1
#string team2
#string venue
+virtual getscore()
+setscore()
+getscorepage()
+print()
+print()
teamplayers
#string name
#string team
#string role
#int price
#int num
#int budget
+virtual displayplayers()
+checkbudget()
livescore
+getscore()
results
+getscore()
batsman
-runs
+displayplayers()
+getbatsman()
bowler
-wkts
+displayplayers()
+getbowler()
wk
-catches
+displayplayers()
+getwk()
allrouner
-runs
+displayplayers()
+getallrounder()

Function Overloading

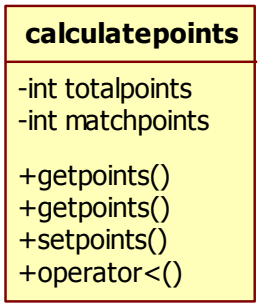
-




```
void print(string name)
{
    cout<<"\n\n\t-x-x-x-x-x-x-x-x-PSL FANTASY LEAGUE-x-x-x-x-x-x-x-x-"<<endl<<endl;
    cout<<"\t                        "<<name<<endl<<endl<<endl<<endl<<endl;
}

void print(void)
{
    cout<<"\n\n\n\n\n\n\n\tx-Exit";
    getch();
}
```

Operator Overloading

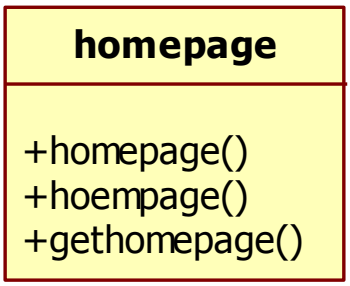
-




```
bool operator <(const calculatepoints& c)
{
if(totalpoints < c.totalpoints)
return true;

else
return false;
}
```

Constructor Overloading

-



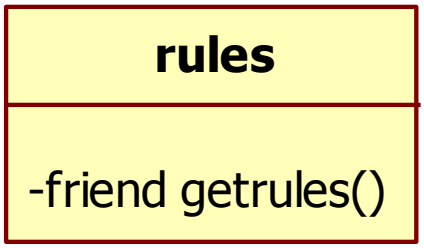

```
homepage(void)
{
system("cls");
cout<<endl<<endl<<endl<<endl<<endl;
cout<<"\t\t                        WELCOME TO          "<<endl<<endl;
}

homepage(int d)
{
delay=d;
Sleep(delay);
cout<<"\t\t**************        **************        *"<<endl;
Sleep(delay);
cout<<"\t\t*             *       *                     *"<<endl;
Sleep(delay);
cout<<"\t\t*             *       *                     *"<<endl;
Sleep(delay);
cout<<"\t\t*             *       *                     *"<<endl;
```

.

.

.

Friend Function




```
class rules
{
    friend void getrules(rules r);
};

     void getrules(rules r)
    {
        system("cls");
        cout<<"\n\n\t-x-x-x-x-x-x-x-x-PSL FANTASY LEAGUE-x-x-x-x-x-x-x-x-"<<endl<<endl;
        cout<<"\t                        Rules"<<endl<<endl<<endl<<endl<<endl;
        cout<<"\t1- You have to select your own fantasy team using the players"<<endl;
        cout<<"\t   in the CREATE TEAM option."<<endl<<endl;
        cout<<"\t2- You have to select players according to the FORMATION you"<<endl;
        cout<<"\t   have selected."<<endl<<endl;
        cout<<"\t3- You have to select players according to the BUDGET i.e 100 CR"<<endl<<endl;
        cout<<"\t4- If you fail to form team according to the BUDGET. You'll have"<<endl;
        cout<<"\t   to do another selection."<<endl<<endl;
        cout<<"\t5- There are 3 types of FORMATION to select from."<<endl<<endl<<endl<<endl;
        cout<<"\t\t\tPress any key to go back...";
        getch();
    }
```

STL Library – Vector

```
void getpoints()
{
    char selectPoints;
    int i;
    bool loop=true;

    vector<int> points;
    vector<int>::iterator pointIt;

while(loop)
{
selectPoints=getch();

switch(selectPoints)
{
case '1':
matchpoints=553;
points.push_back(matchpoints);
cout<<"\tMatch 1 added"<<endl;
break;
case '2':
matchpoints=240;
points.push_back(matchpoints);
cout<<"\tMatch 2 added"<<endl;
break;
case '3':
matchpoints=410;
points.push_back(matchpoints);
cout<<"\tMatch 3 added"<<endl;
break;
case '4':
matchpoints=545;
points.push_back(matchpoints);
cout<<"\tMatch 4 added"<<endl;
break;
case '5':
matchpoints=305;
points.push_back(matchpoints);
cout<<"\tMatch 5 added"<<endl;
break;
case 'c':
points.pop_back();
cout<<"\tLast match deleted."<<endl;
break;
case 'p':
loop=false;
totalpoints=0;

for (int i = 0; i < points.size(); i++)
{
    totalpoints=points[i]+totalpoints;
}
cout<<"\n\n\t\t-x-x-x-x-x-x- Points : "<<totalpoints;
points.clear();
break;
}
}
}
```

Conclusion

This project is designed with determination to every single little instruction, making it user-friendly and trouble-free.

The PSL Fantasy League, covering all the basic concepts of Object Oriented Programming is best for beginners who wish to learn C++ Language.

The project being interesting for the cricket lovers, plays a studious role for the future programmers as well