**ROLLBALL PLAYER MANAGEMENT SYSTEM**

**(SPORTS MANAGEMENT SYSTEM)**

**A Curriculum Project Report**

***Submitted***

***in the partial fulfilment of the requirements for the award of***

***the degree of***

**MASTER OF TECHNOLOGY**

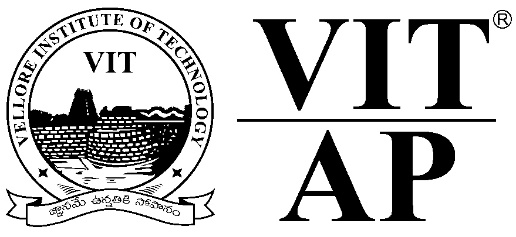
In

**SOFTWARE ENGINEERING**

By

**R V S TEJA KRISHNA**

**Regd. No. 17MIS7085**



**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**VELLORE INSTITUTE OF TECHNOLOGY-ANDHRA PRADESH**

**UNIVERSITY**

**AMARAVATI (BESIDE AP SECRETARIAT)**

**INAVOLU, GUNTUR – 522237, INDIA**

**2020**

**ROLL BALL PLAYER MANAGEMENT SYSTEM**

**(SPORTS MANAGEMENT SYSTEM)**

**CONTENTS**

|  |  |
| --- | --- |
| **TITLE** | **PAGE NO** |
| INTRODUCTION | 3 |
| TECHNOLOGIES | 5 |
| DESIGN | 7 |
| IMPLEMENTATION | 15 |
| TESTING | 22 |
| RESULTS | 27 |
| CALCULATIONS | 32 |
| REFERENCES | 36 |

**CHAPTER -1**

**INTRODUCTION**

This project Roll Ball Player management system has taken place because there are my talented players in our state for every sport, so to prove their talent they are not getting any opportunities and some people are missing them due to less information about the opportunities. To avoid such problems this project has been started to notify the players or others who wanted to prove themselves in this game, that is to maintain the information of the competitions going to be held and winners of the competitions which are already held, so that when the players or newly registered people will get notification regarding the competitions going to be held under this association. The reasons why this projects started is that previously some of the talented players have missed some events in this association because of lack of information regarding the registration and start dates, and some don’t know where to register how to get registered for the event, so to overcome this problem the project has been started.

The projects main aim is to make the registrations online, and maintain the details of the players who have registered for particular competition. The benefits of this project is whenever the organizers will add an event the players who subscribed(created an account) will get a mail notification regarding the event and for those who are not registered has to check the website to get the details of the event. The players need not go anywhere to get registered he/she can easily register from his mobile by logging into this website.

The target users of the website are players, organizers, coaches from other places.

**CHAPTER -2**

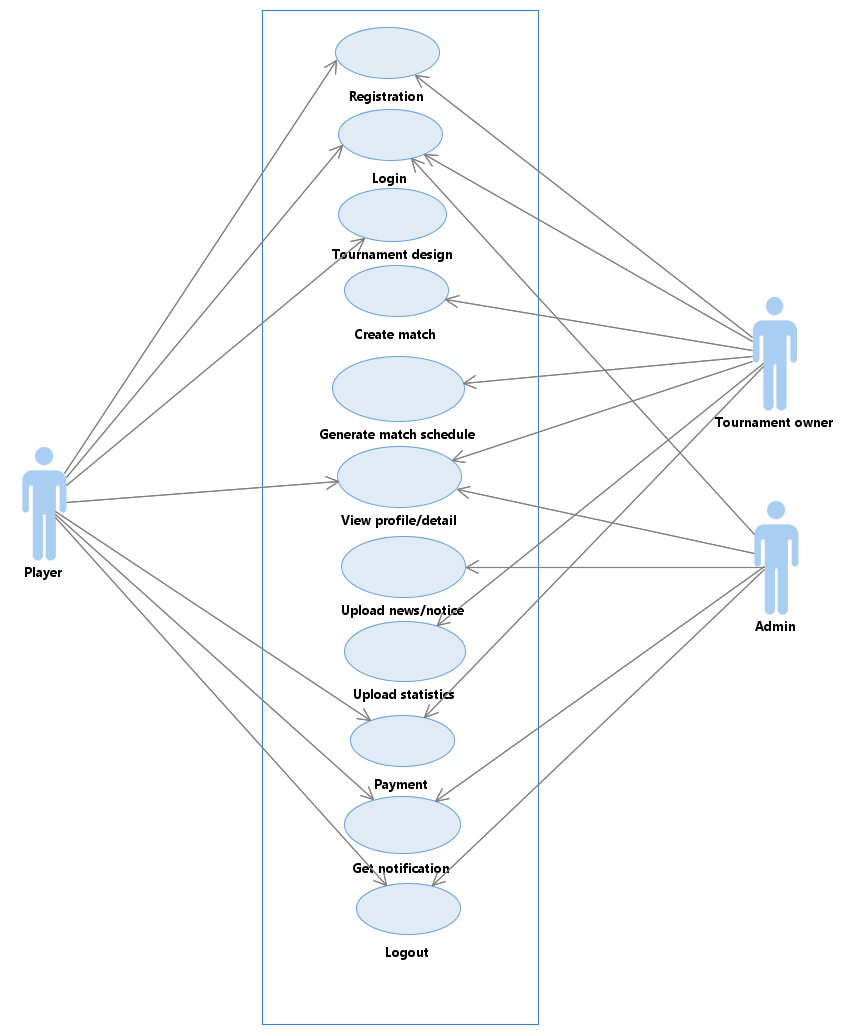
**TECHNOLOGIES**

­

­To build this project of Roll Ball Player Management System the technologies used are HTML, PHP, MYSQL, CSS, XML, BOOTSTRAP and JAVASCRIPT. The HTML is used for the user interface development to build the static and dynamic WebPages to make the website more user-friendly for the users even if they are new to the website, and PHP is used to connect the front end static and dynamic webpages to the database so that the information from the user entered in the website will be stored into the database, and the database used for connecting the website to store the details of the players entered is MYSQL from XAMPP server, and to make the website more colorful and attractive we used CSS and BOOTSTRAP for styling the websites and XML is used to view the contact details because it has the ability to describe the information in our own tag names, and for the validation of forms and validating whether actual user is logging into the website, or any wrong passwords entered by the user.

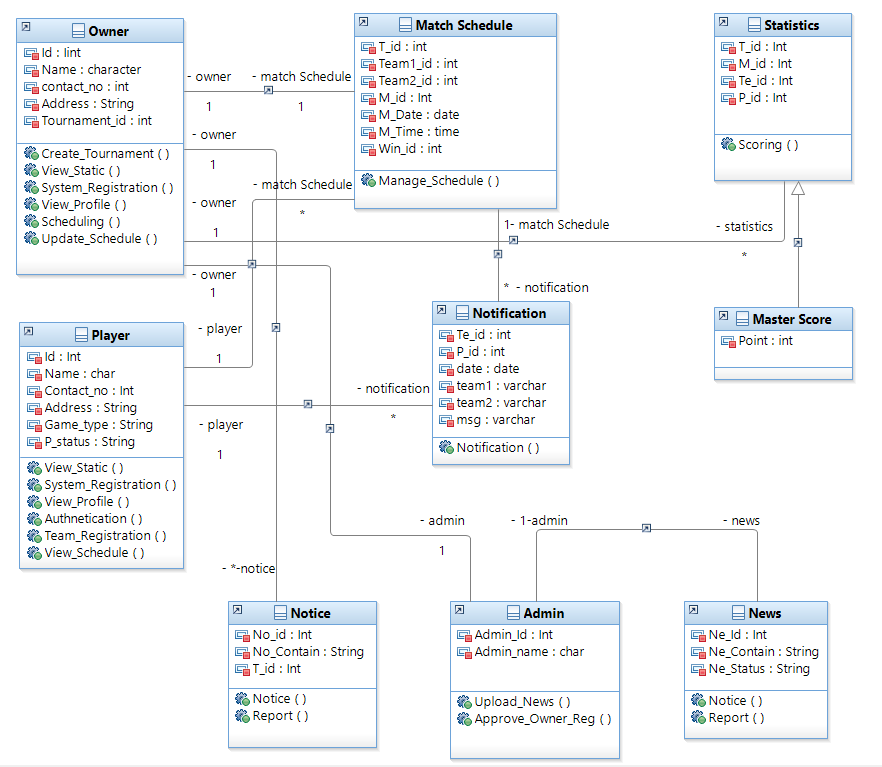
**CHAPTER-3**

**DESIGN**

**USE CASE DIAGRAM**:

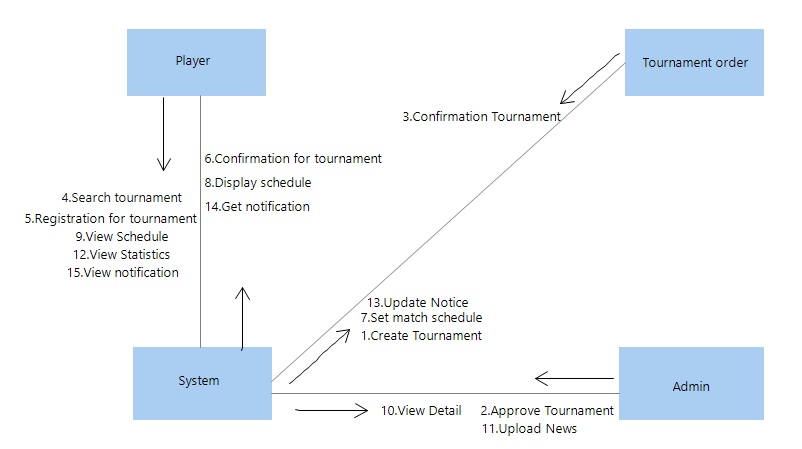
According to the above diagram the player can be able to register to the competitions and get logged to get the details of the events and tournament owner(organizer) will take care of creating a match and a schedule and managing all the works of the tournament, the admin can also do the same thing as the tournament owner and some extra functionalities for the admin to maintain the website information.

**CLASS DIAGRAM:**



In this diagram we can get the information about each activity, which are the details going to be stored in the database and the operations done by every class i.e., like organizer can create a tournament, update the schedule and such operations.

**COLLABORATION DIAGRAM:**



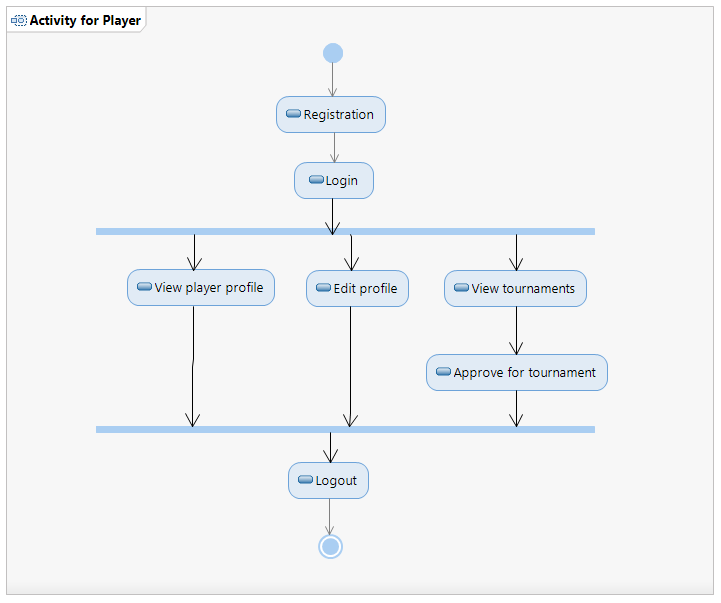
This diagram shows about the actions performed between the classes so that the access of operations can be given to the particular class to avoid the operation clashes.

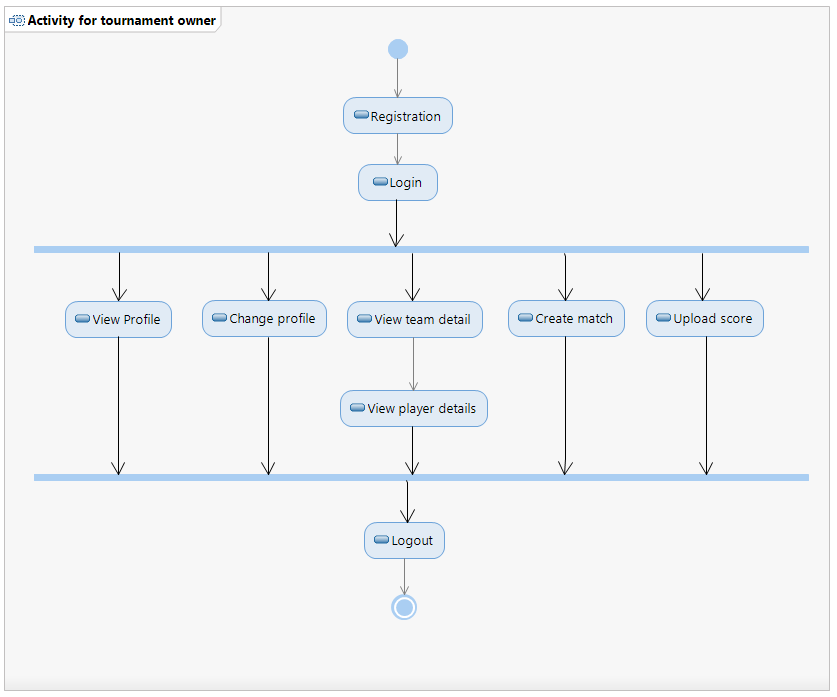
**SEQUENCE DIAGRAM:**



The sequence diagram shows the sequential order of the process of how user will interact with the system and how the system will interact with database to develop the project easily.

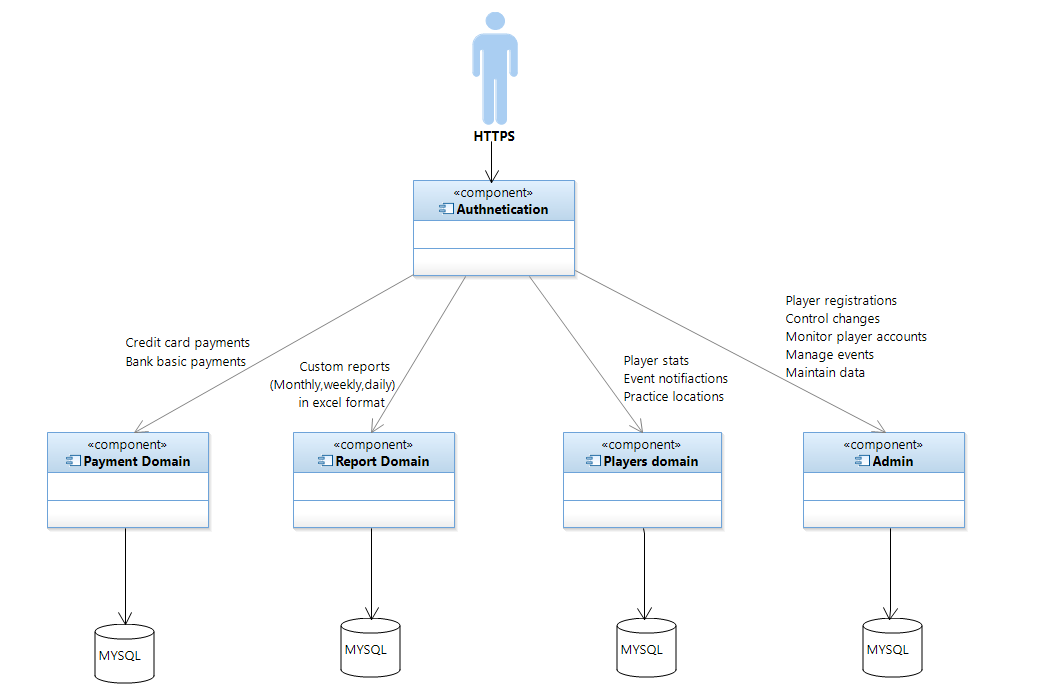
**ACTIVITY DIAGRAM:**





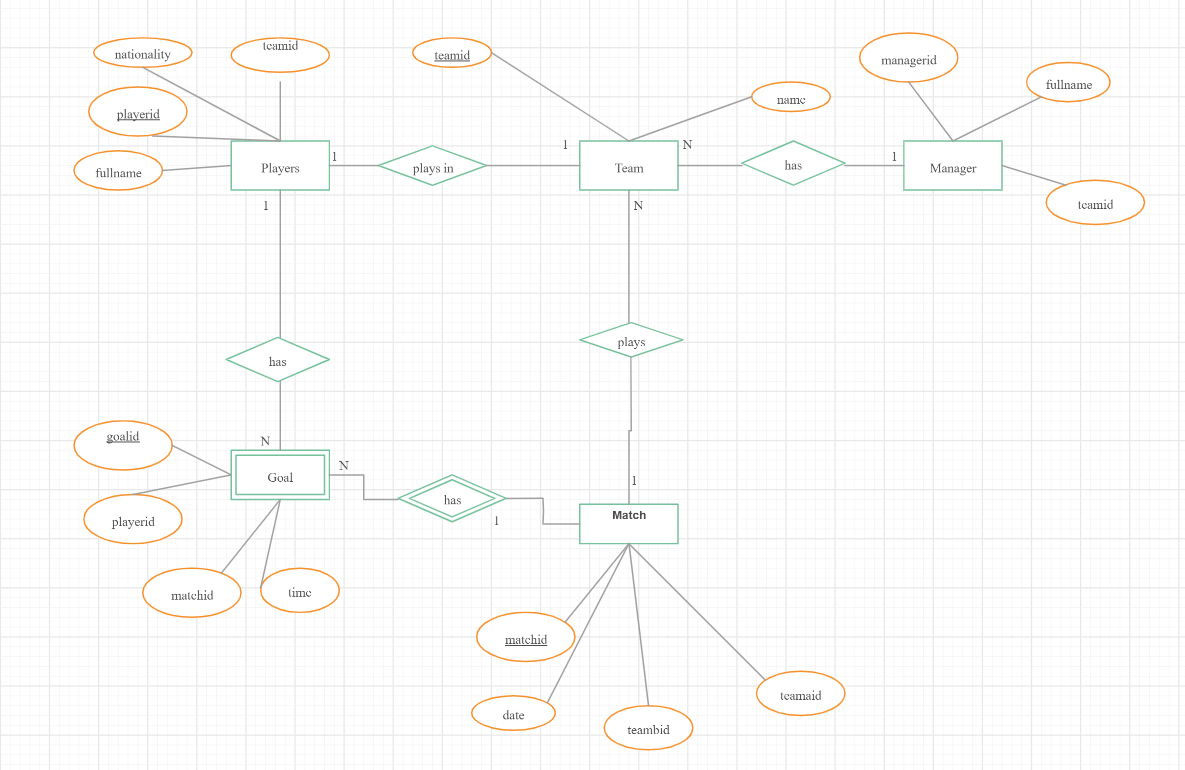
The activity diagram tells about the operations done by any user in a detailed manner. To implement the operations for the particular users it can be useful for the developer to complete the project with this activity diagram.

**ARCHITECTURE DIAGRAM**:



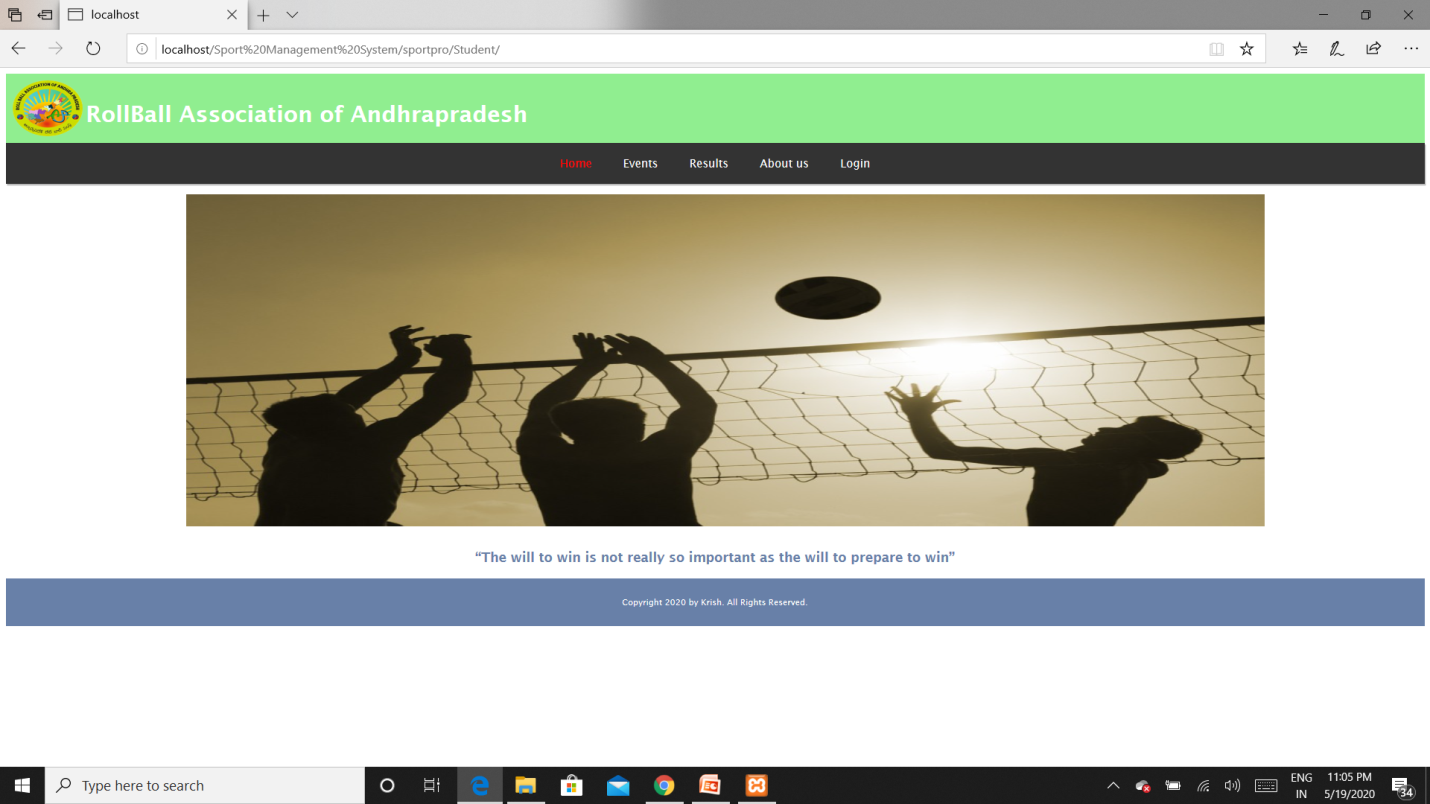
This diagram shows how the project is connected to the database and how the website is authenticated by the user. This tells the information running the backend of the website.

**ER DIAGRAM:**

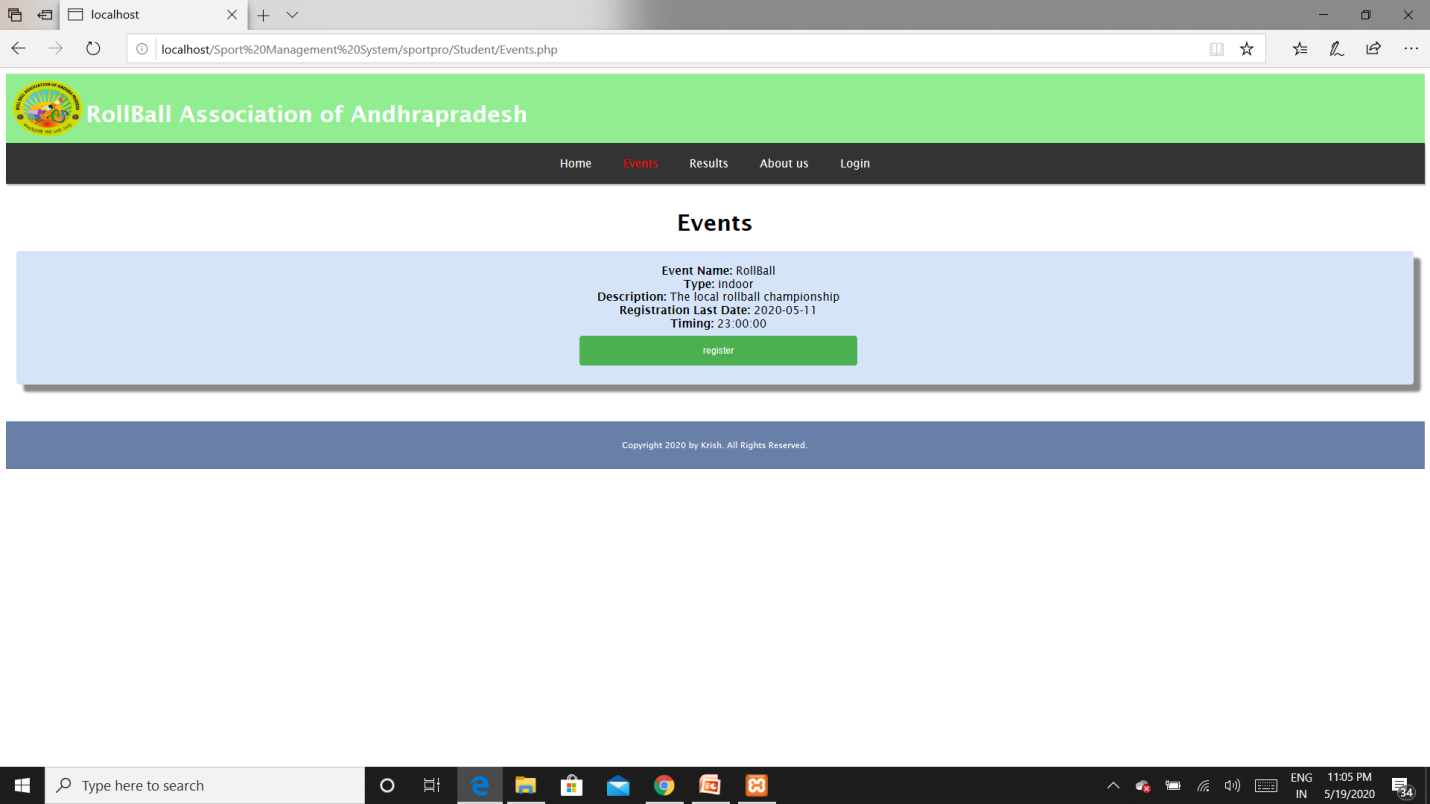


**CHAPTER -4**

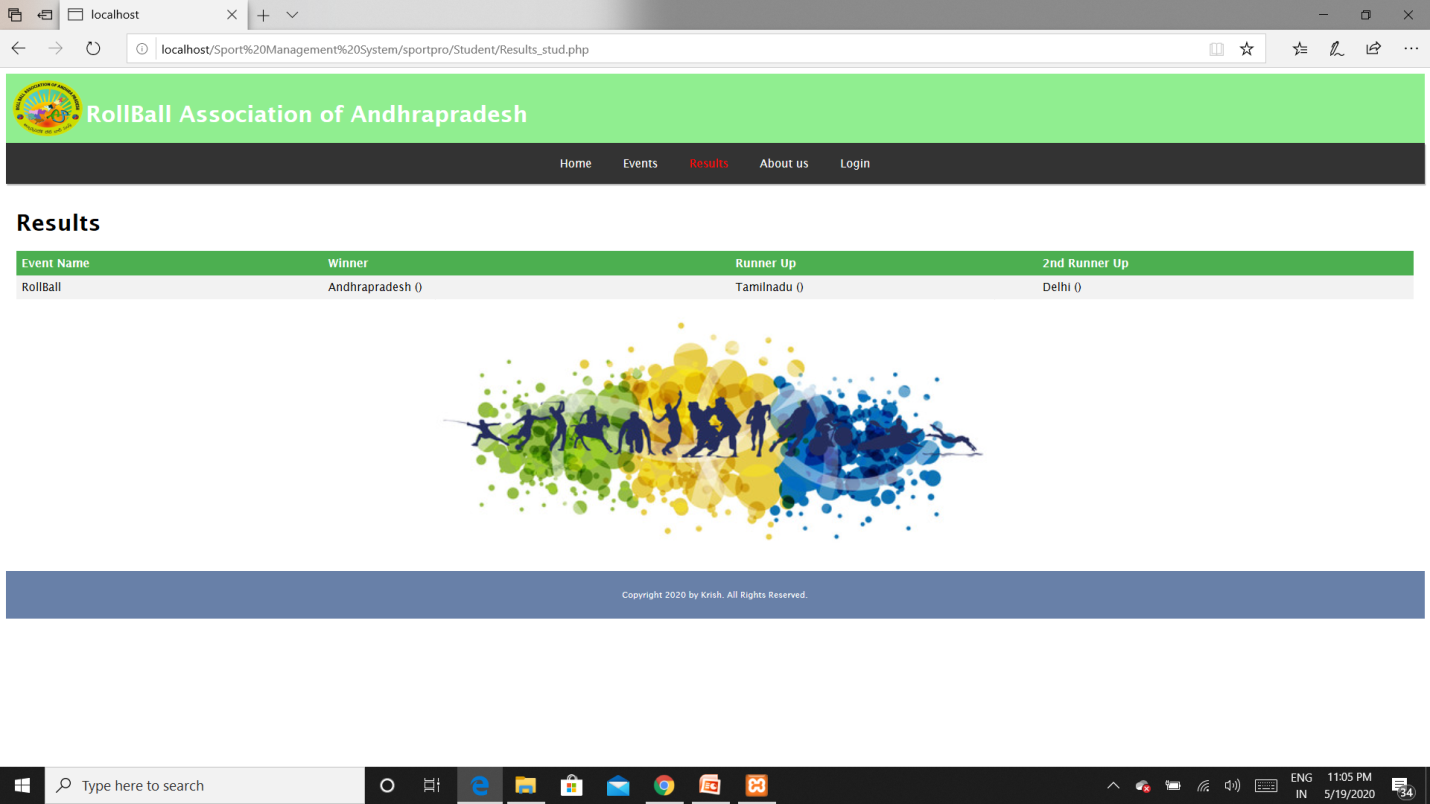
**IMPLEMENTATION**

****

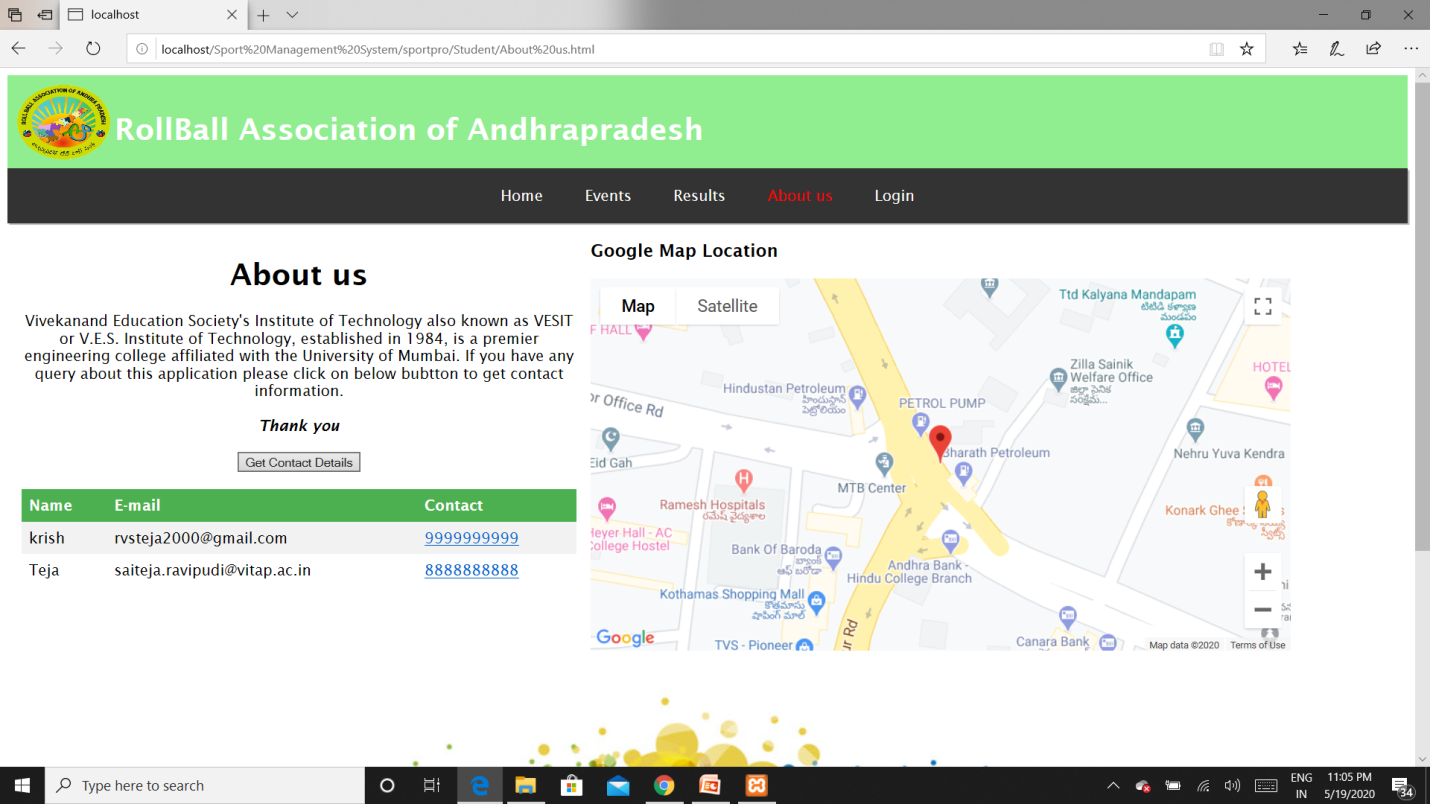
**Fig 4.1** Home page for players – This page welcomes player with sliding images on screen

****

**Fig 4.2**Events page-Shows list of upcoming events and player can register

****

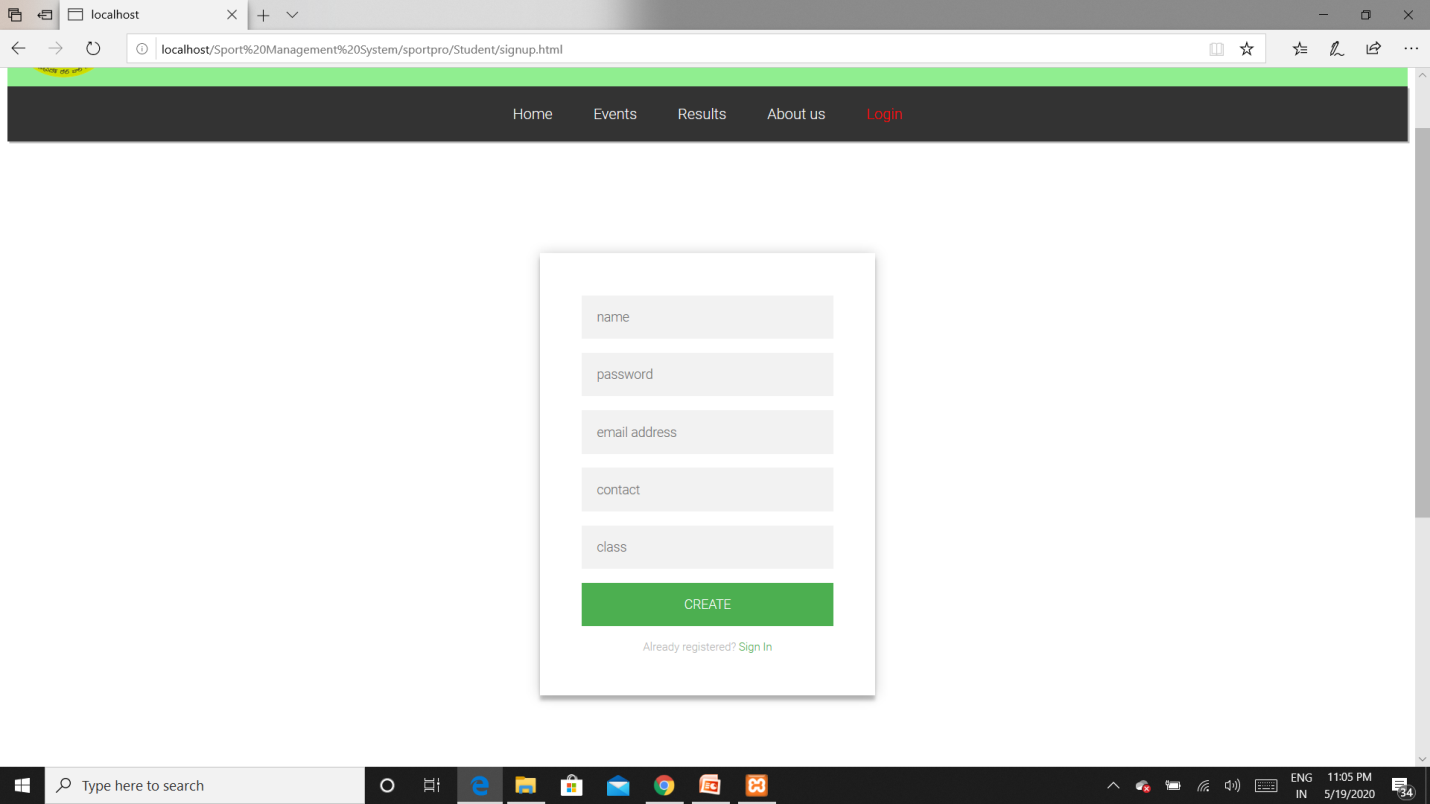
**Fig 4.3**Reports page – Shows results of competitions held

****

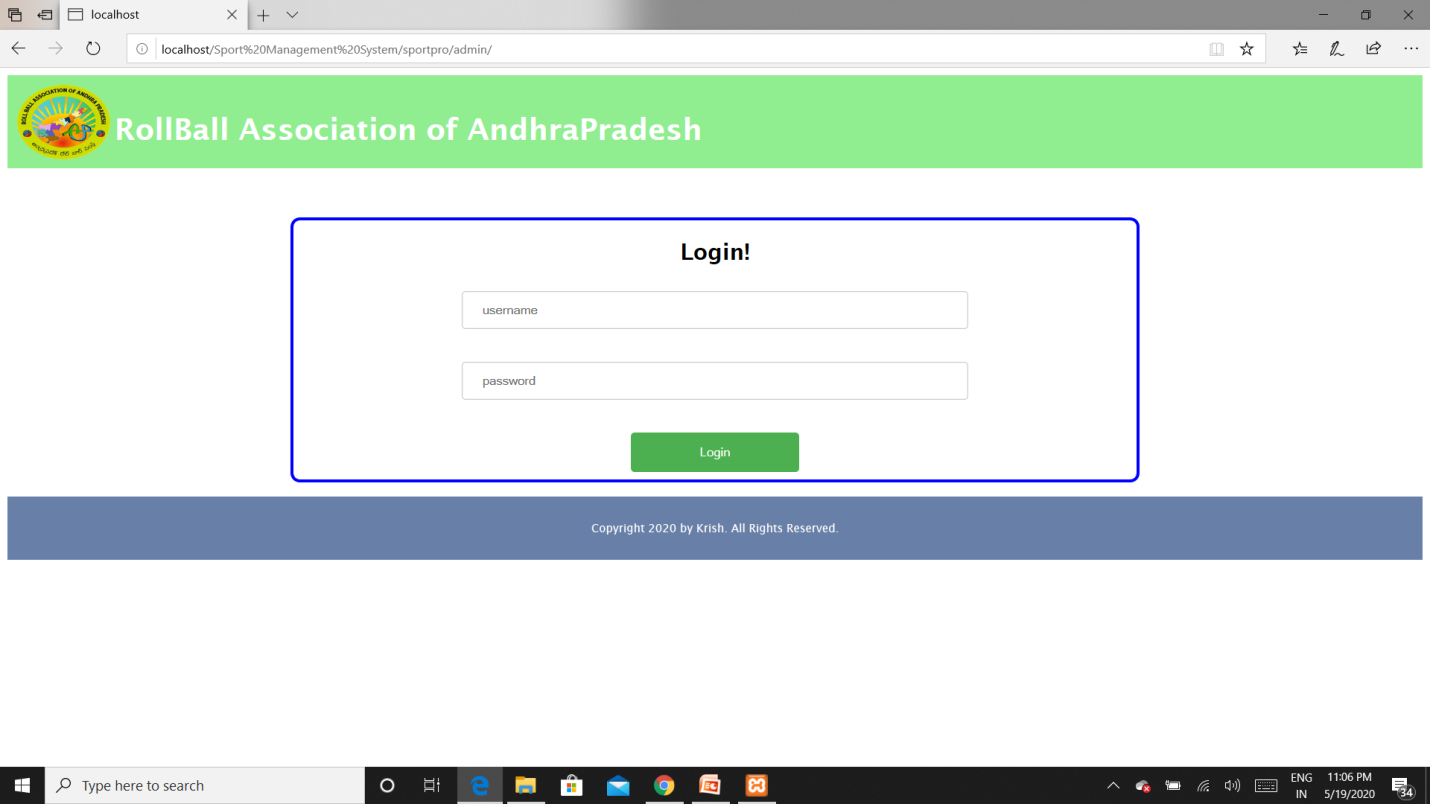
**Fig 4.4** About us page – Contains contact details and organization details

****

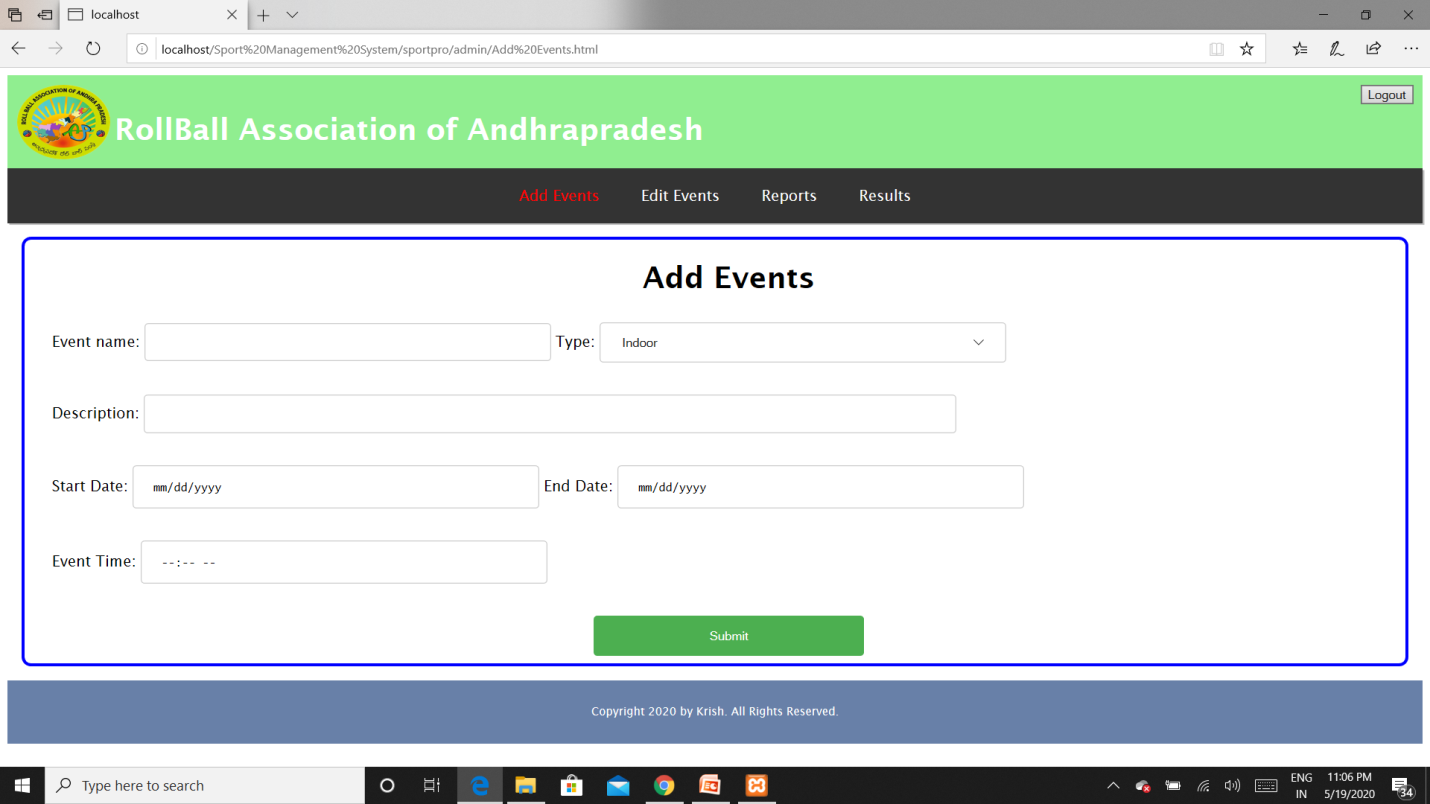
**Fig 4.5** Login page for players – Players can login here

****

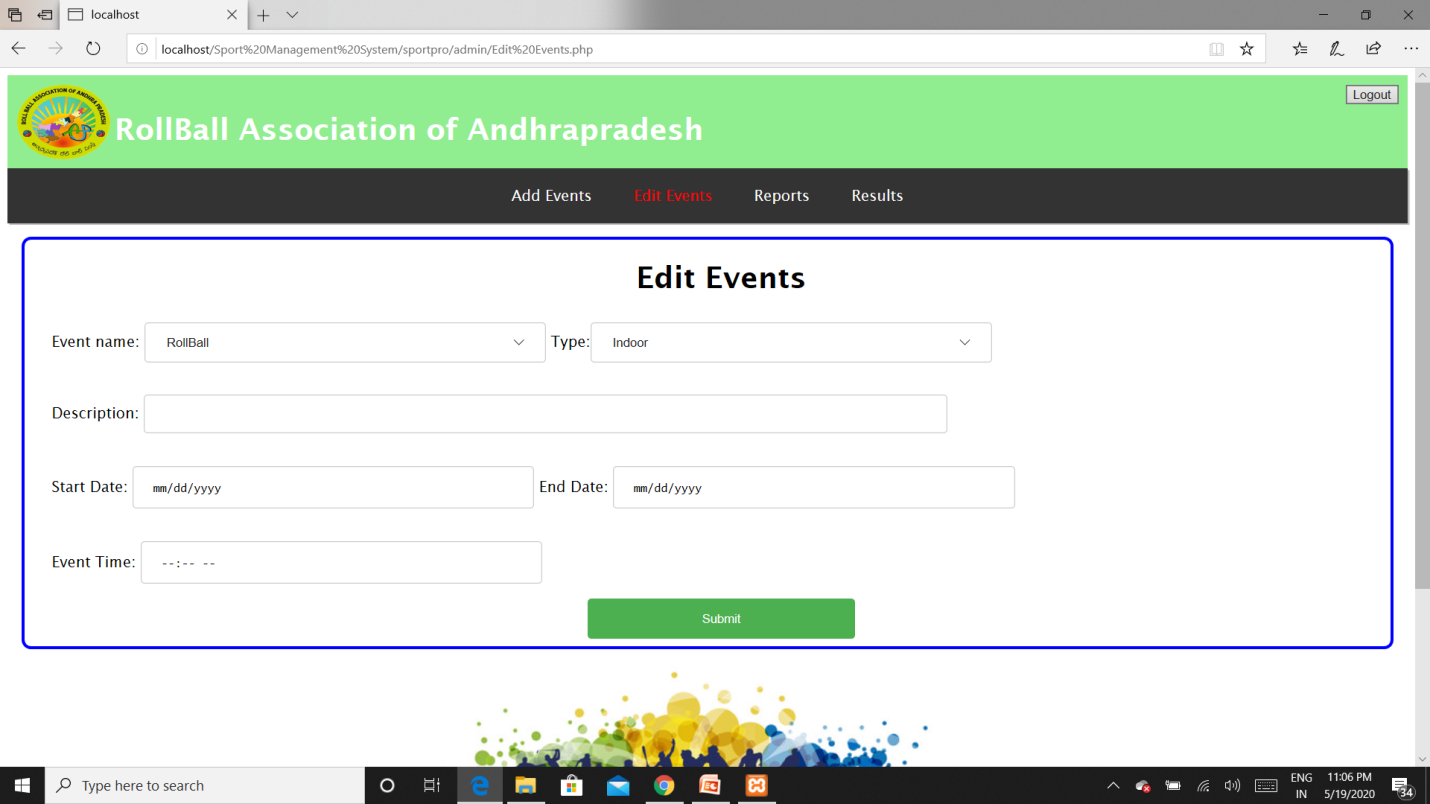
**Fig 4.6**Signup page for players – Players can create account here

****

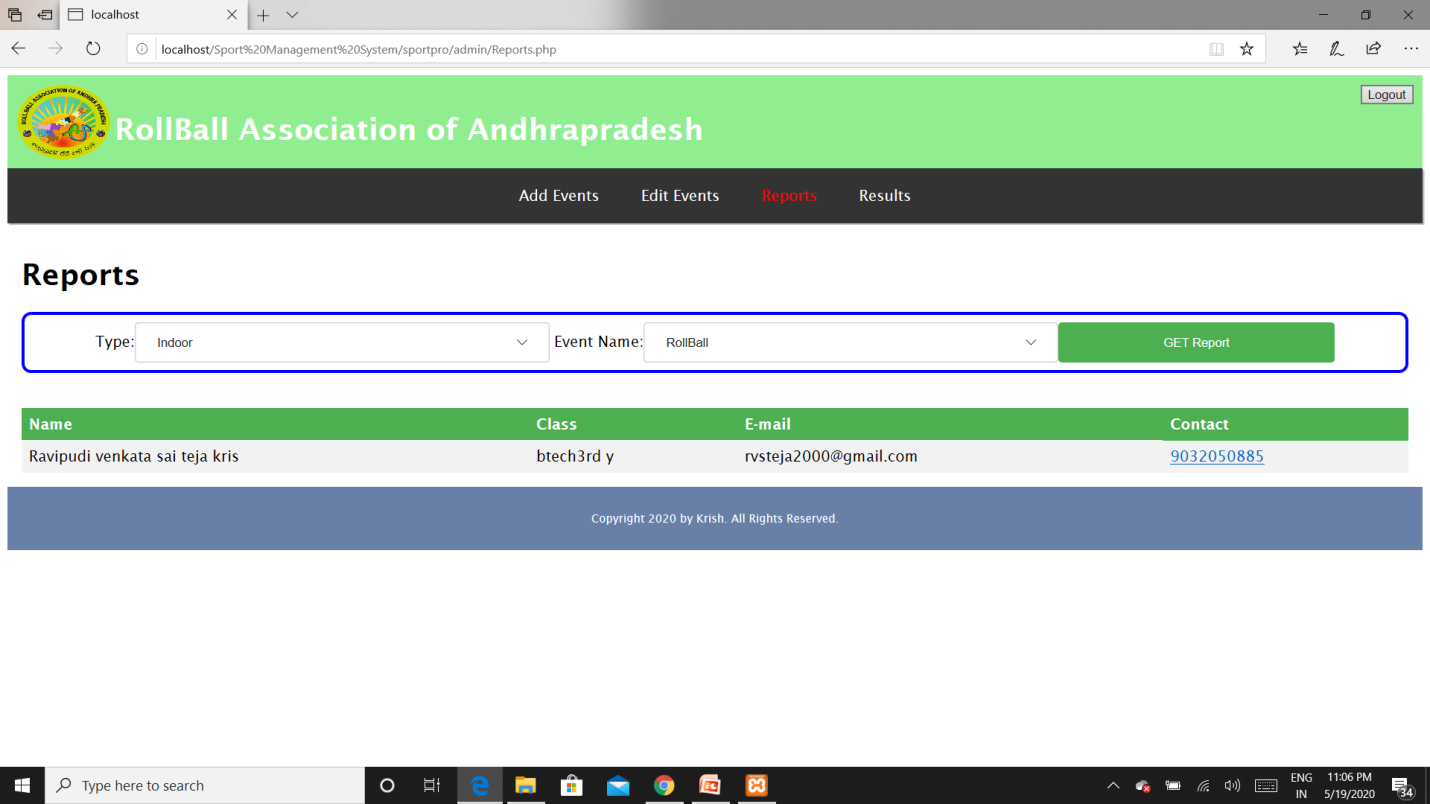
**Fig 4.7**Login page for admin – Adminwill login here

****

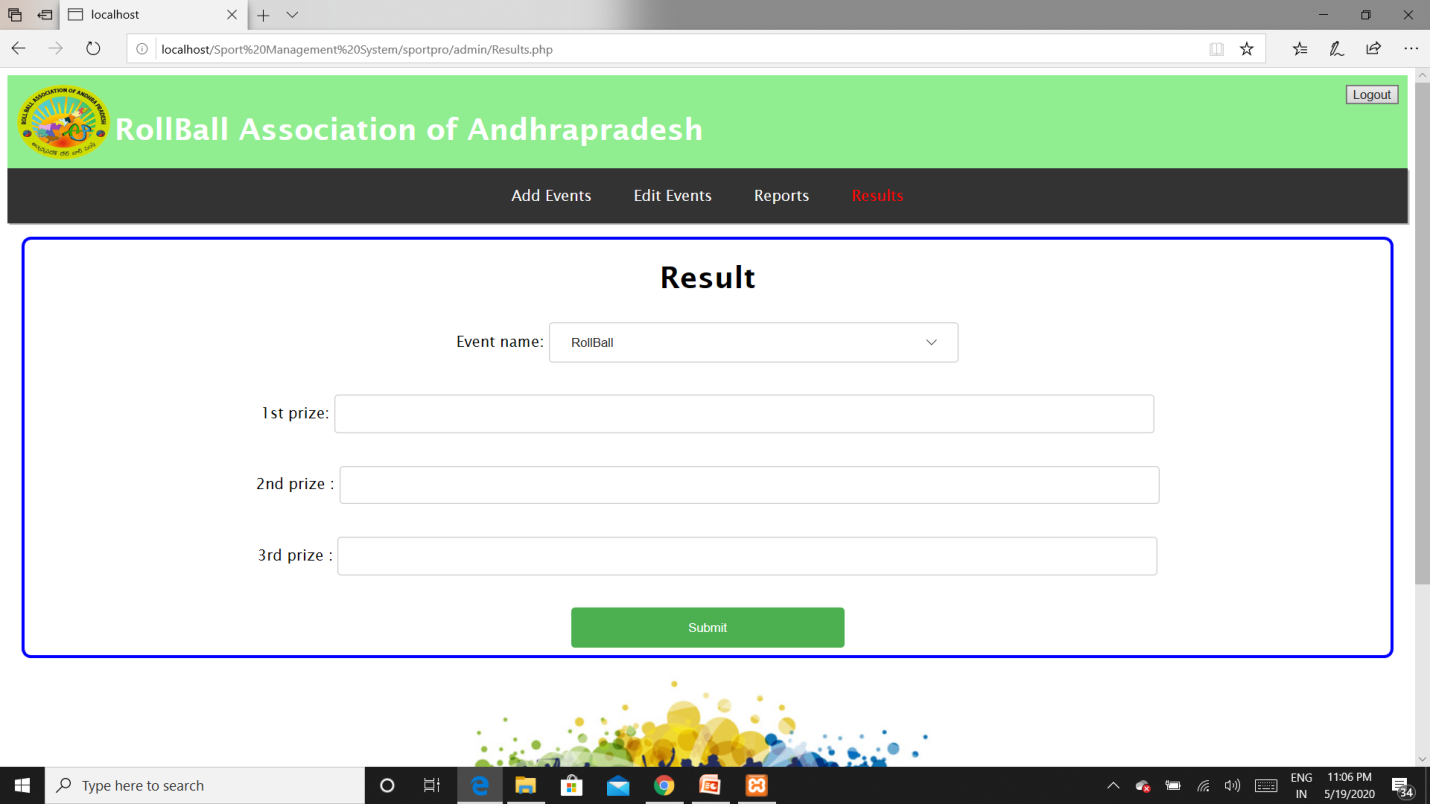
**Fig 4.8** Add events page – Admin can add new events here

****

**Fig 4.9** Edit events page – Admin can edit event details which are already added previously

****

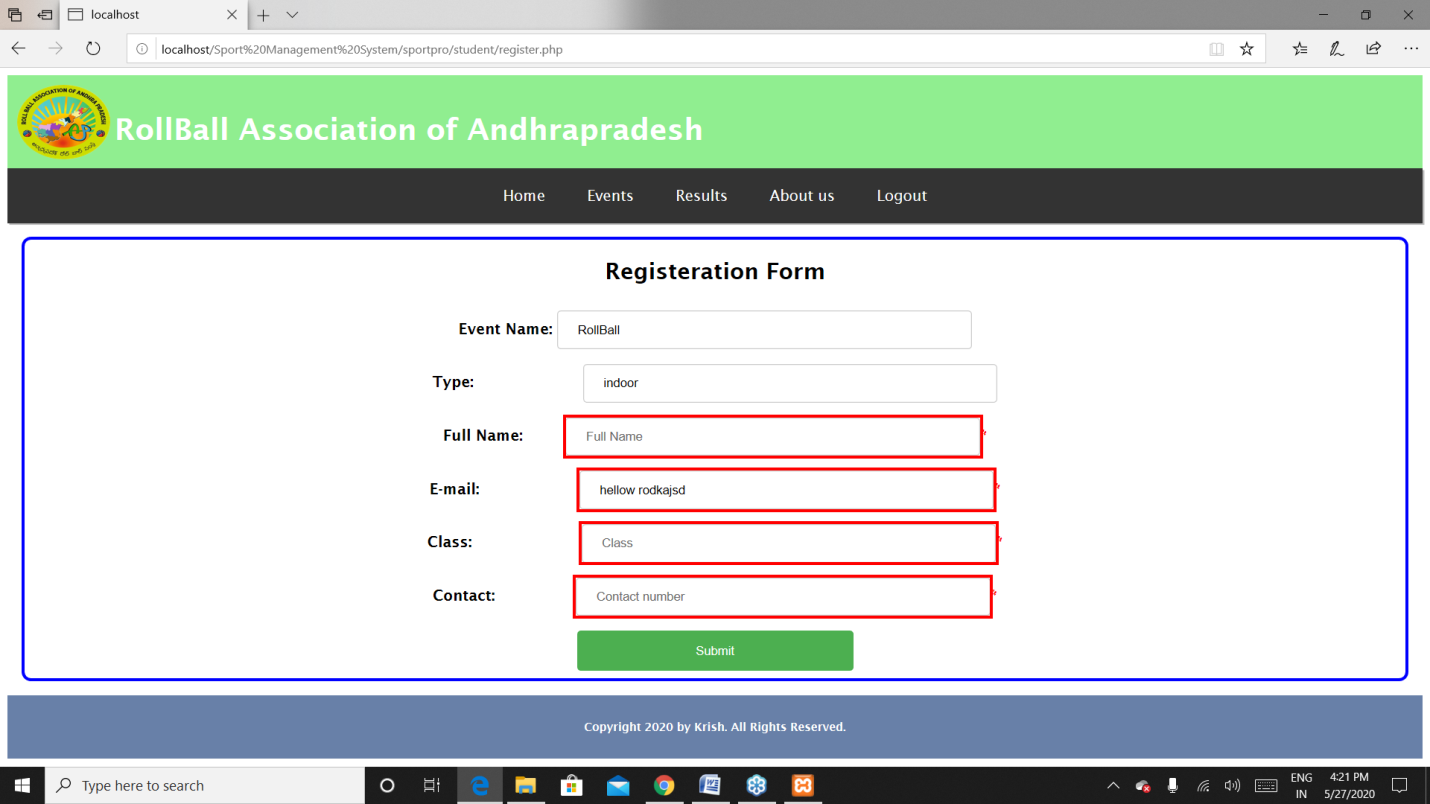
**Fig 4.10**Reports page – Registered players list is presented here in table

****

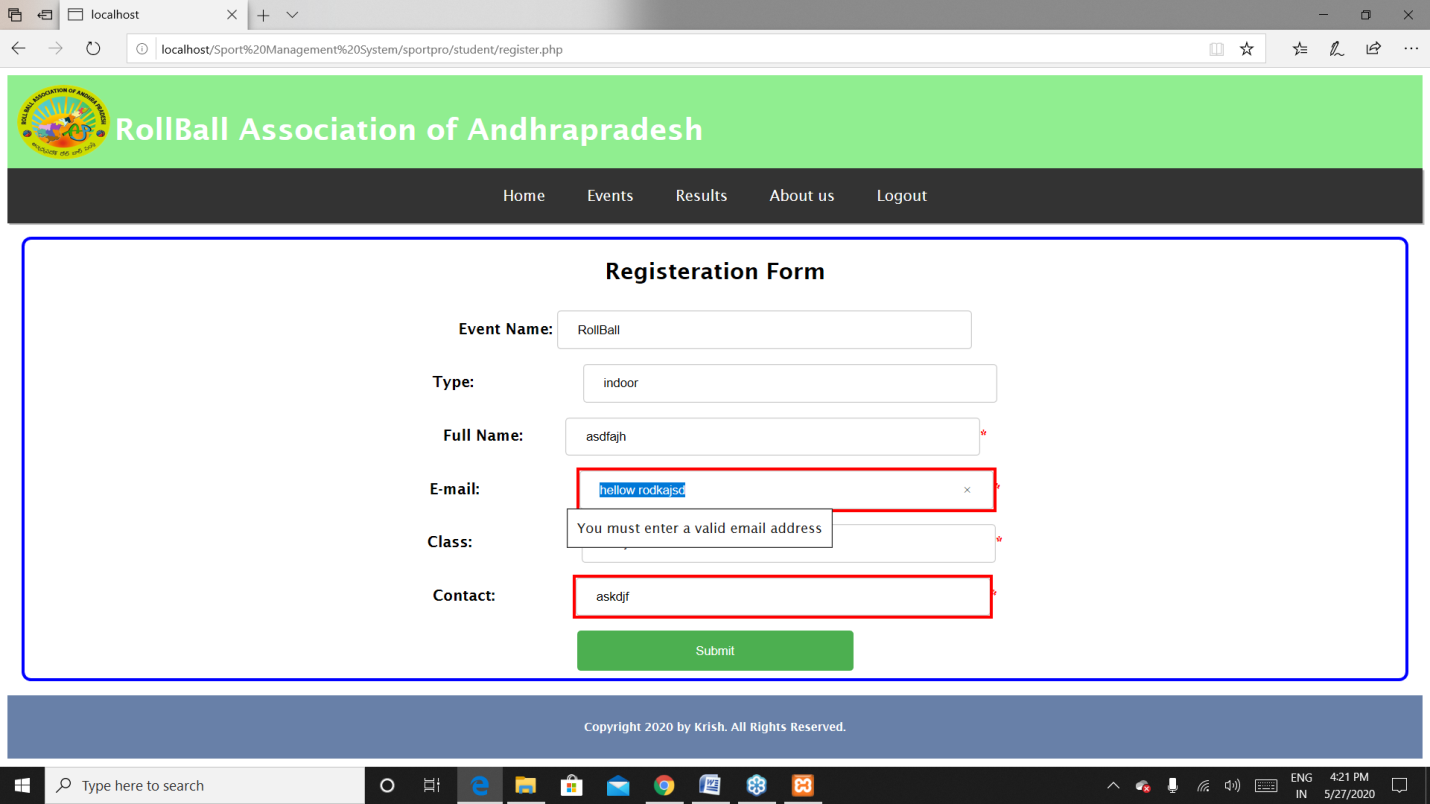
**Fig 4.11**Results page – Results of competitions held are updated here by admin

**CHAPTER -5**

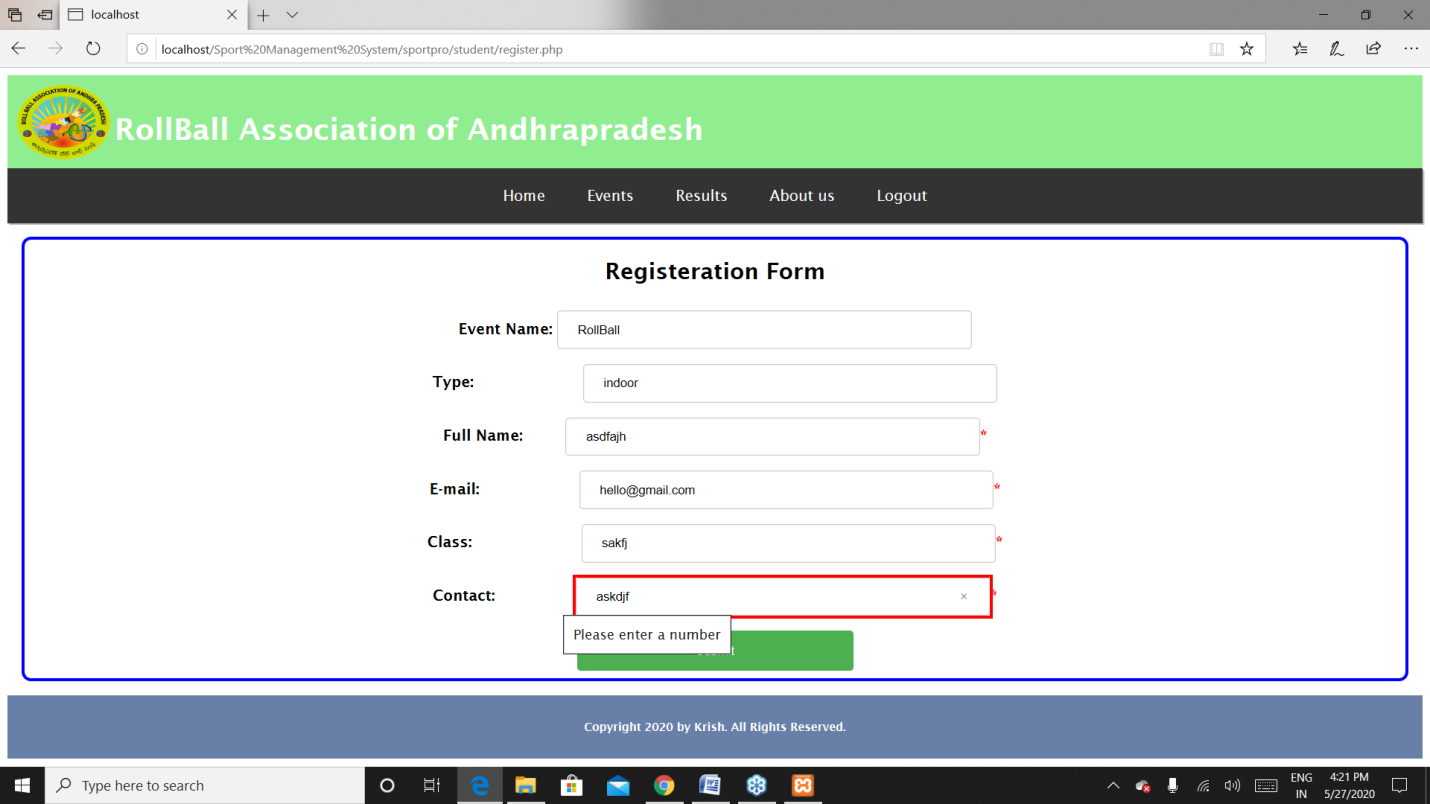
**TESTING**

****

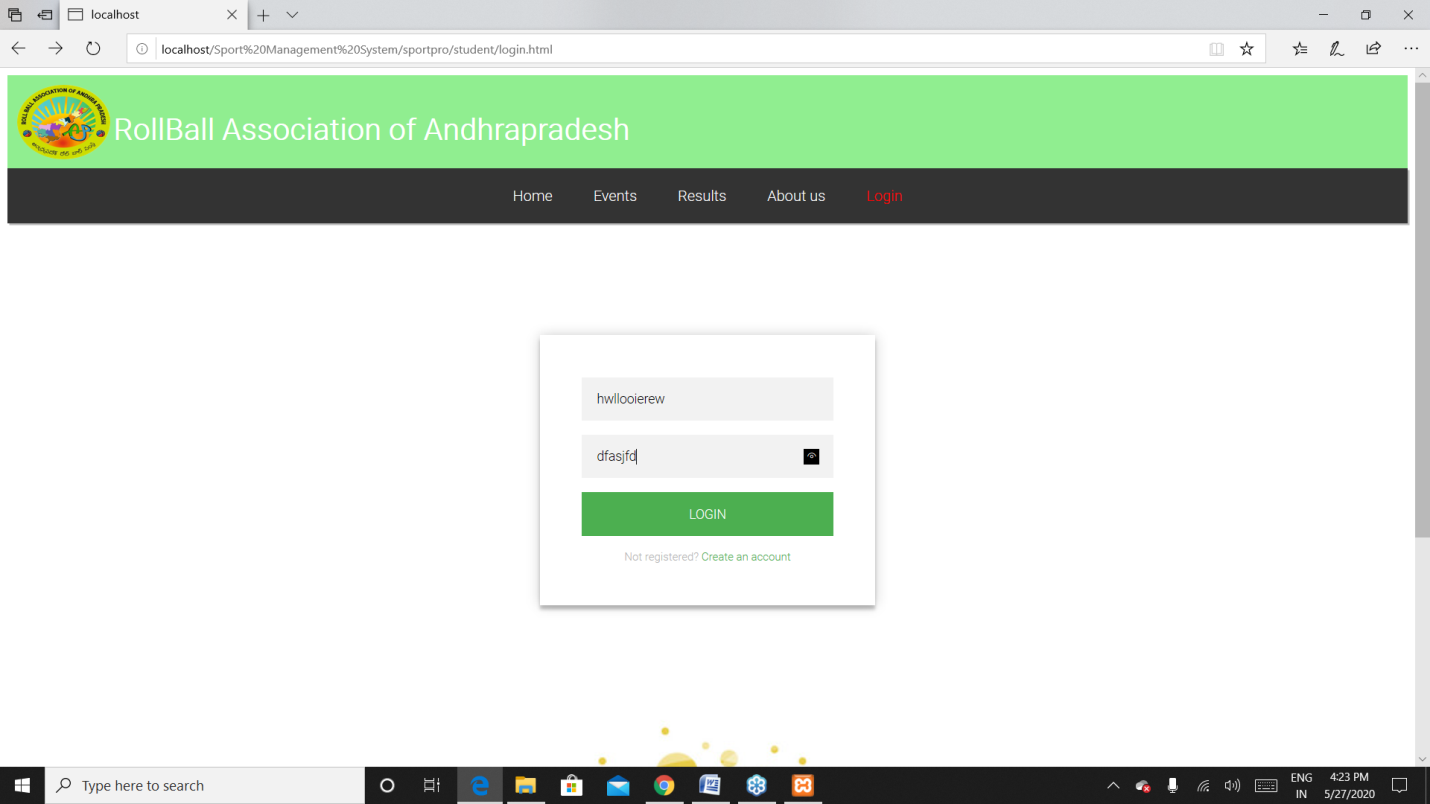
**Fig 5.1** Registrations page – if the players did not enter the details of themselves and click on submit then it highlights that they are required.

****

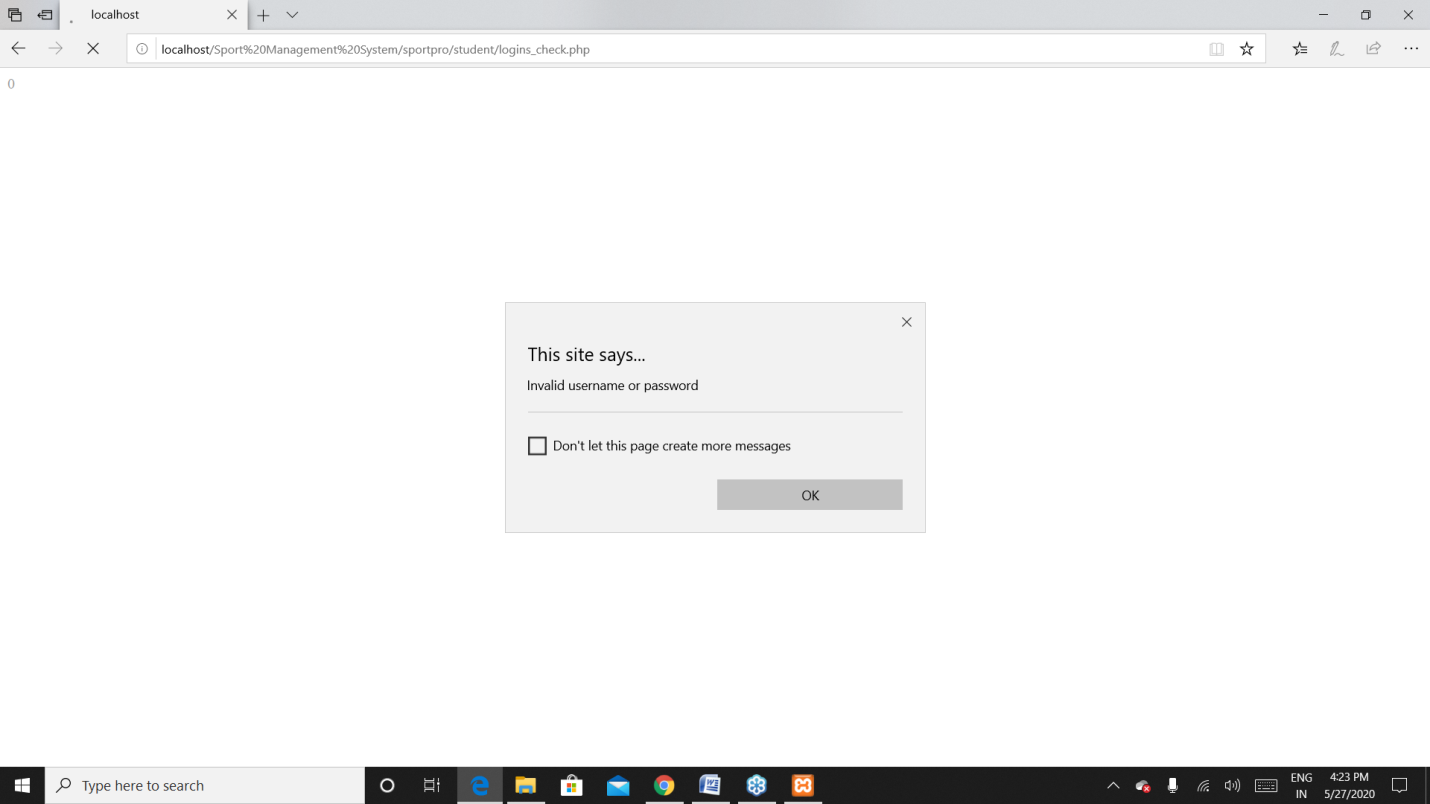
**Fig 5.2** Registrations page – if the players did not enter the email properly then it will show an error to enter the mail correctly.

****

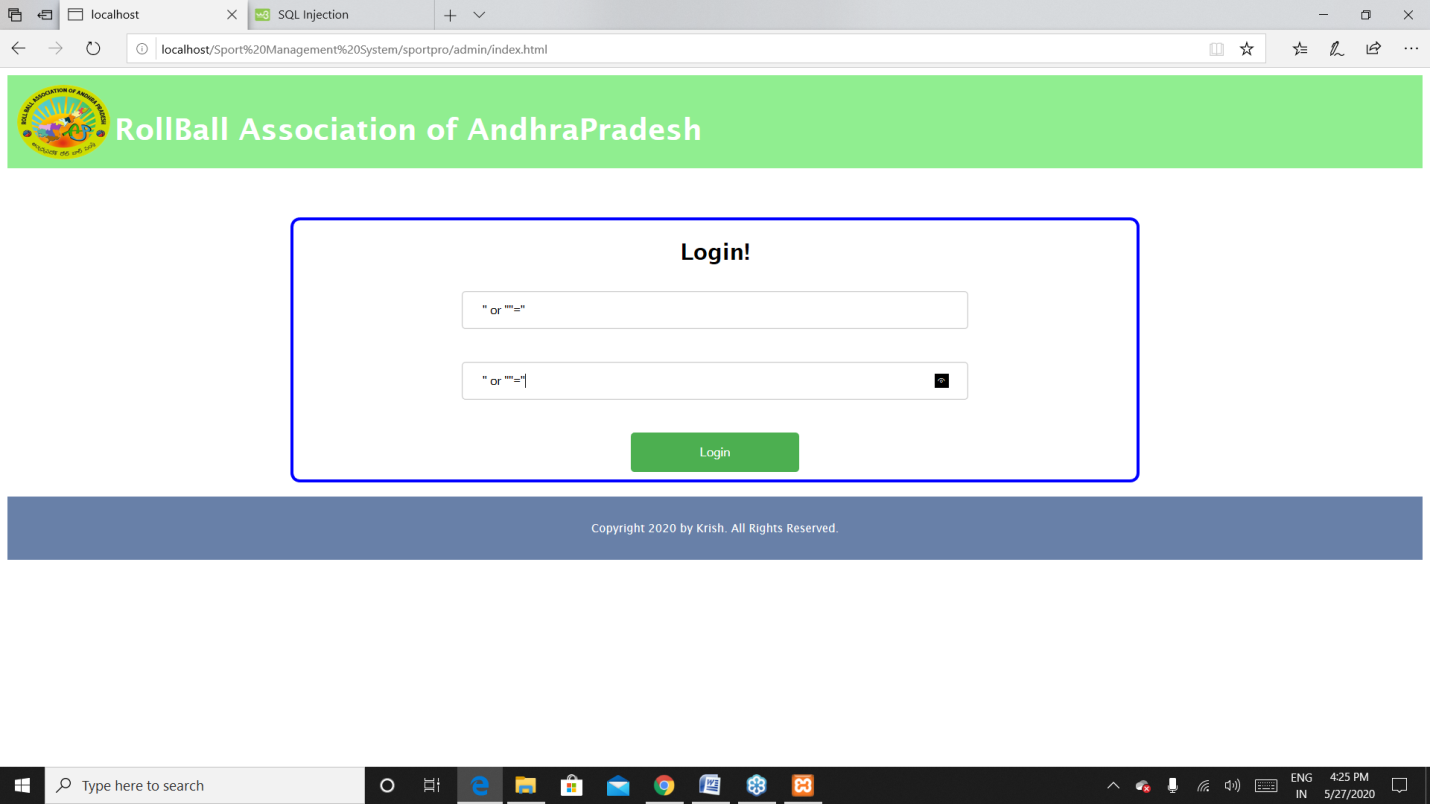
**Fig 5.3** Registrations page – if the players did not enter the mobile number correctly that is entering string type instead a number.

****

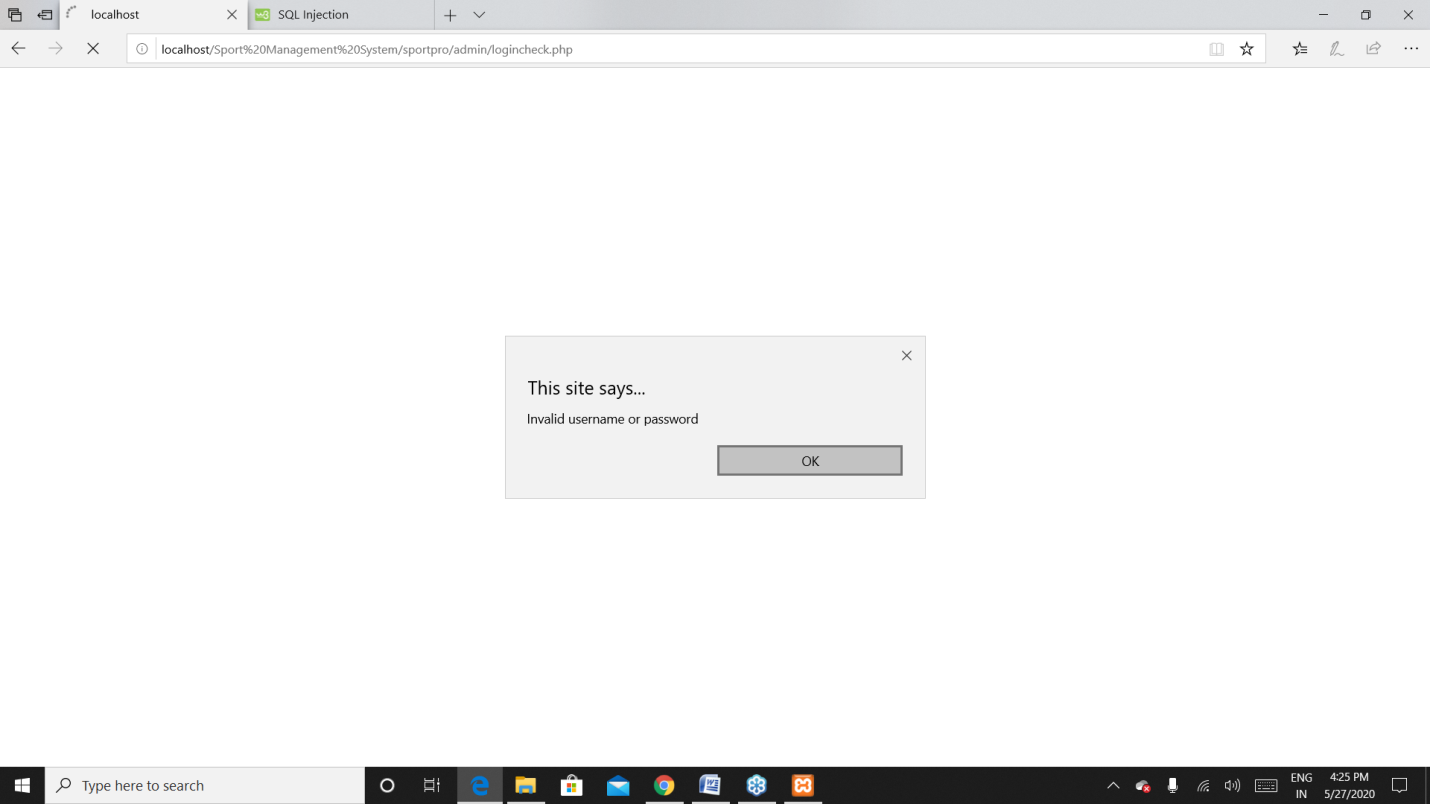
**Fig 5.4** login page – Entering the wrong credentials and clicking on login leads to next picture

****

**Fig 5.5** login page – if the player enters the wrong credentials while logging in it show a dialogue box.

****

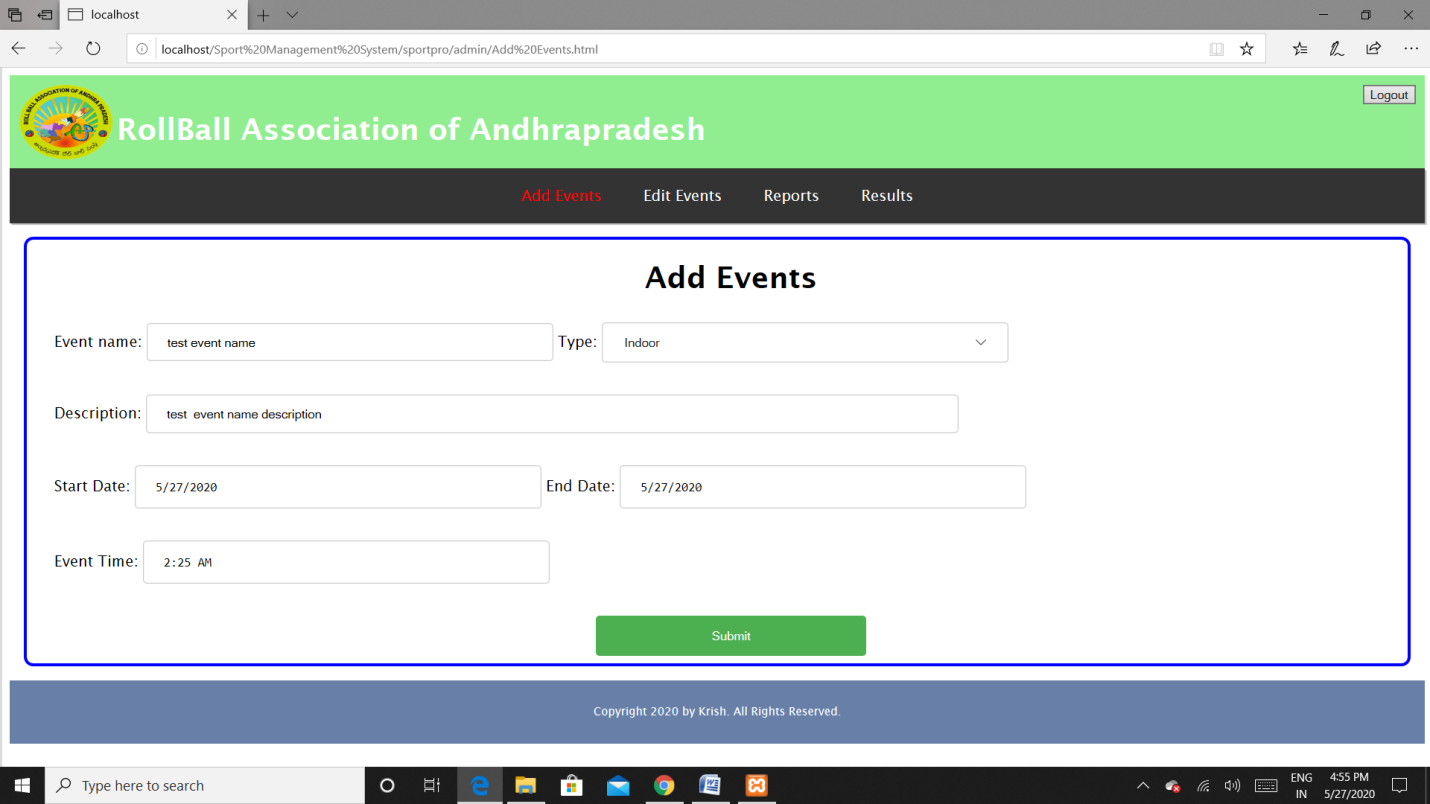
**Fig 5.6** admin login page – if any hacker wants to log into admins page using some sql injections as above it will not be possible because xampp provides defulat security for the database to avoid sql injections if so given will display a dialogue box as next picture.

****

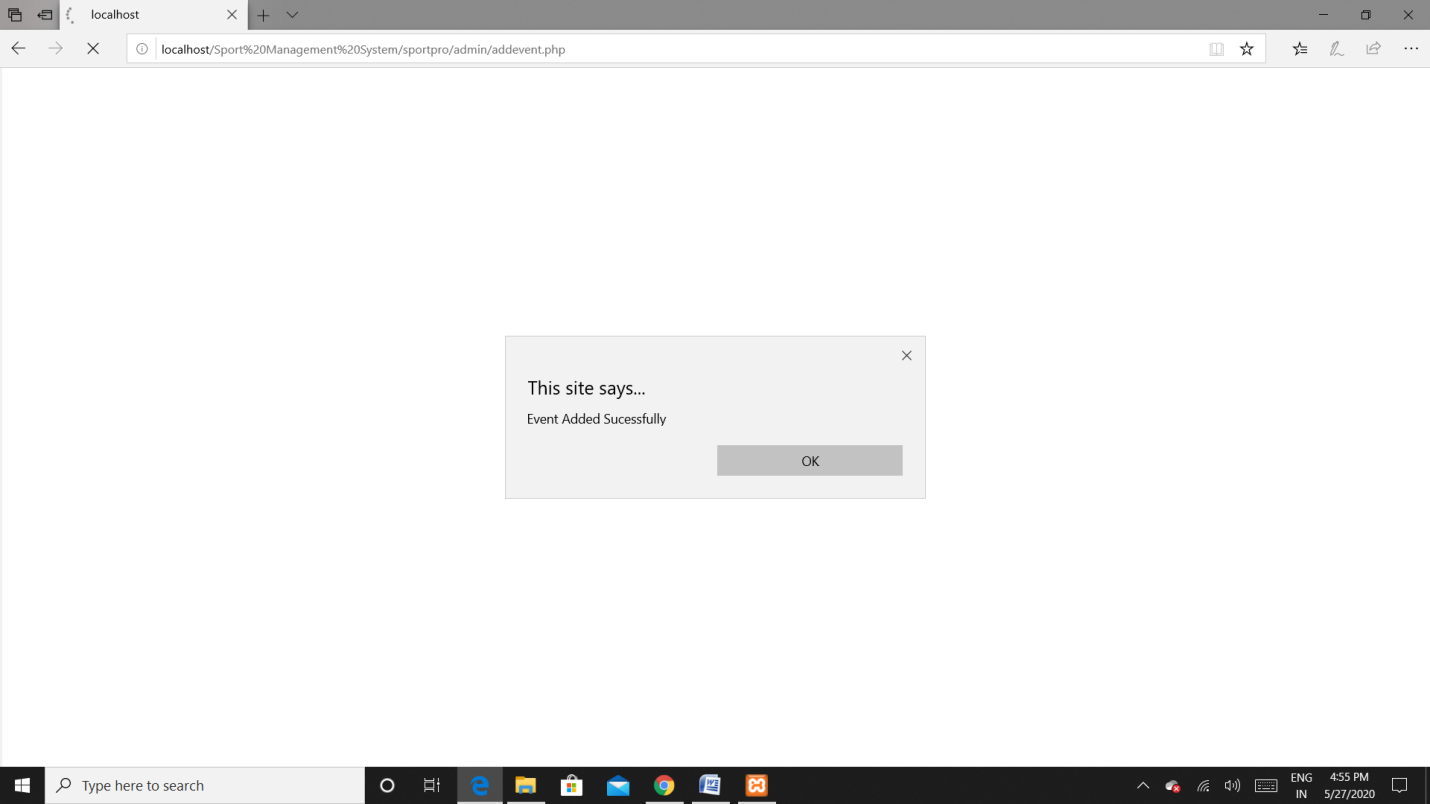
**Fig 5.7** admin login page – trying for sql injections.

**CHAPTER -6**

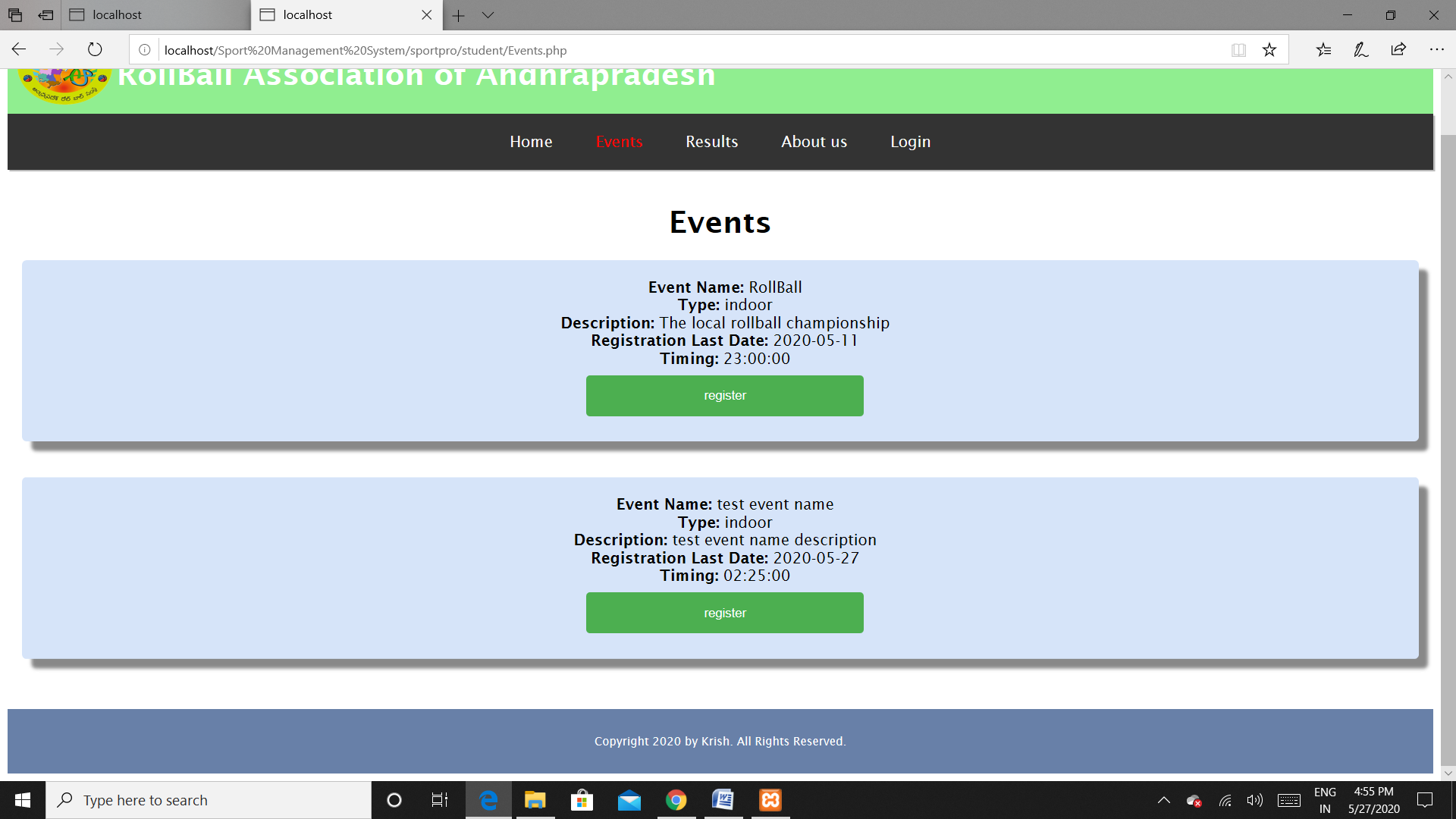
**RESULTS**

****

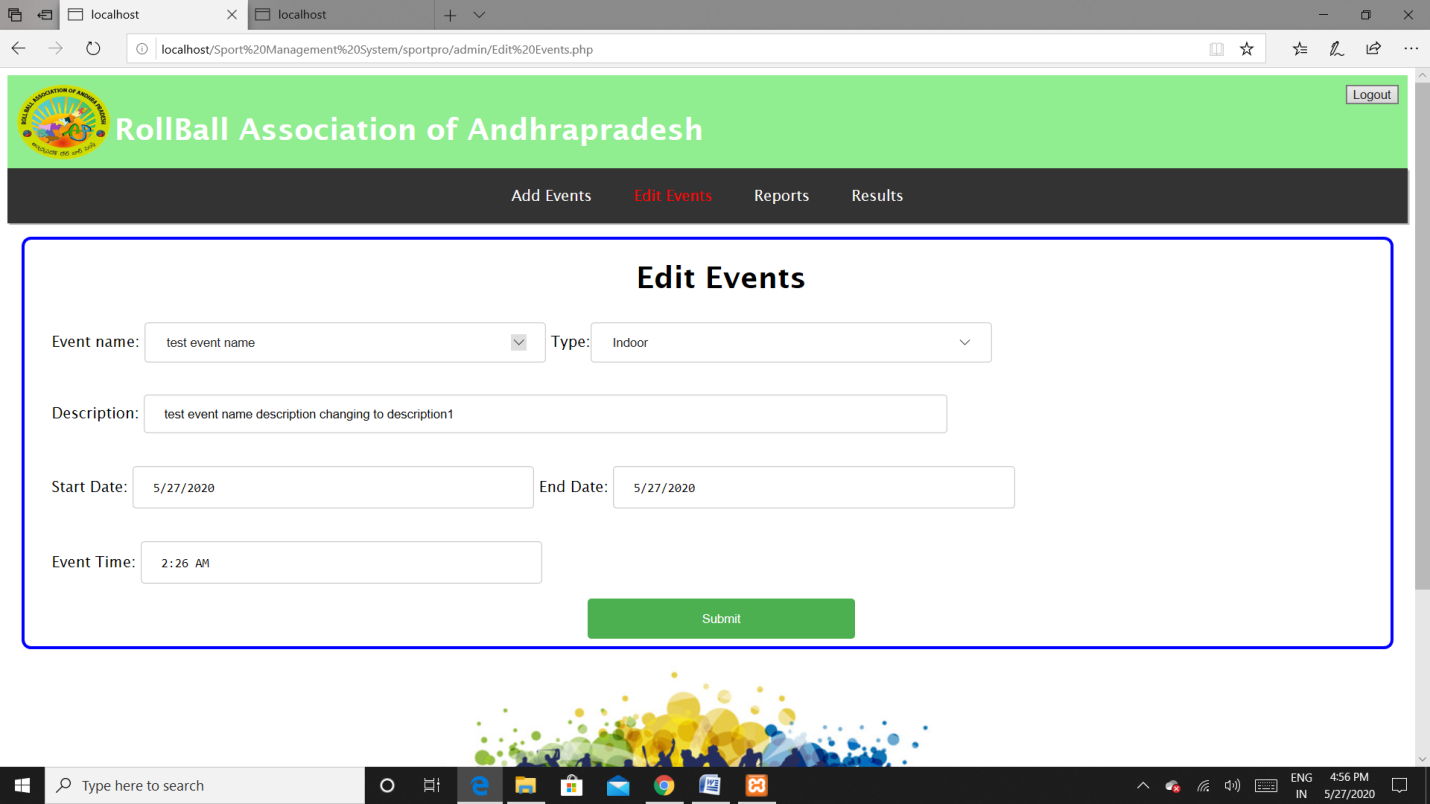
**Fig 6.1** admin add events page

****

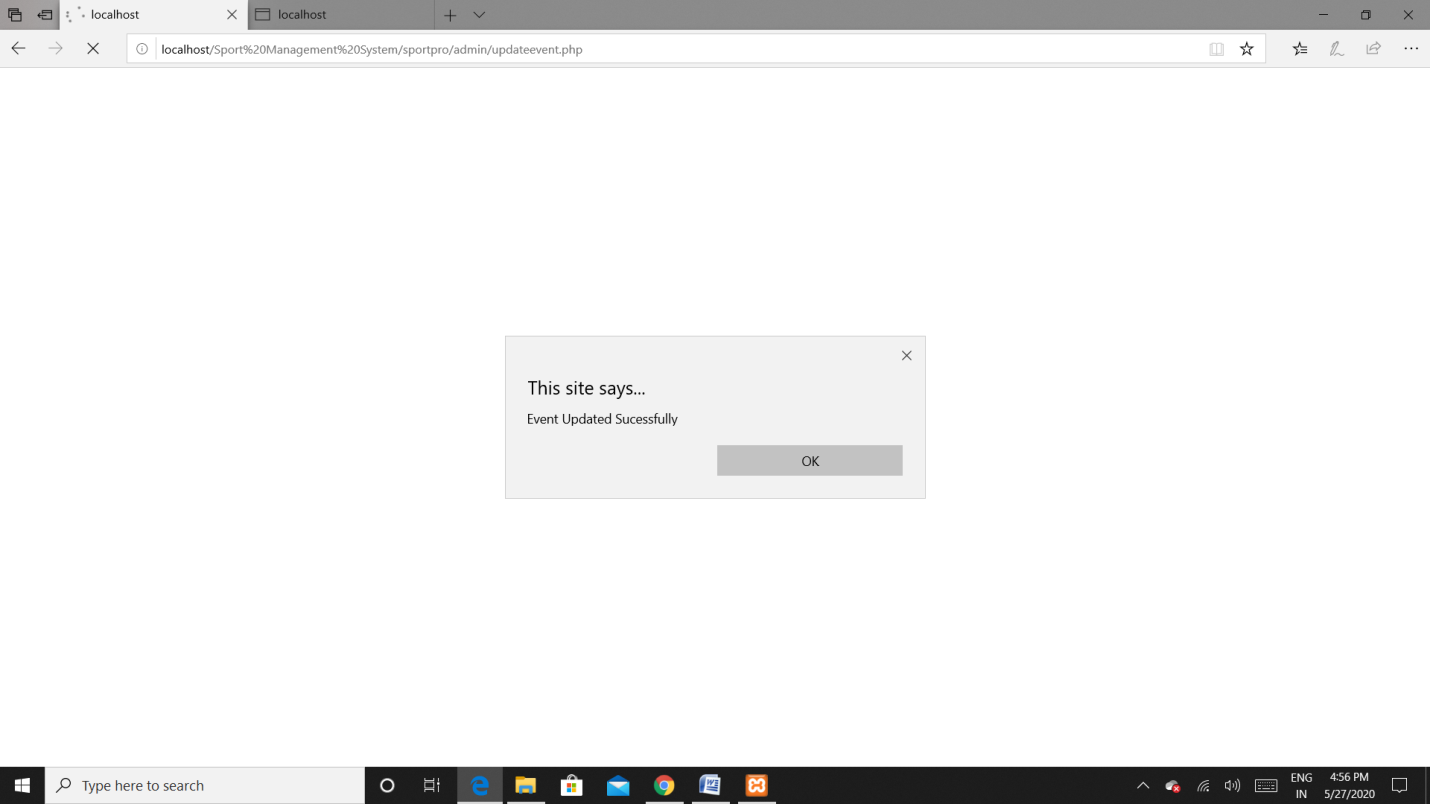
**Fig 6.2** admin add events page confirmation

****

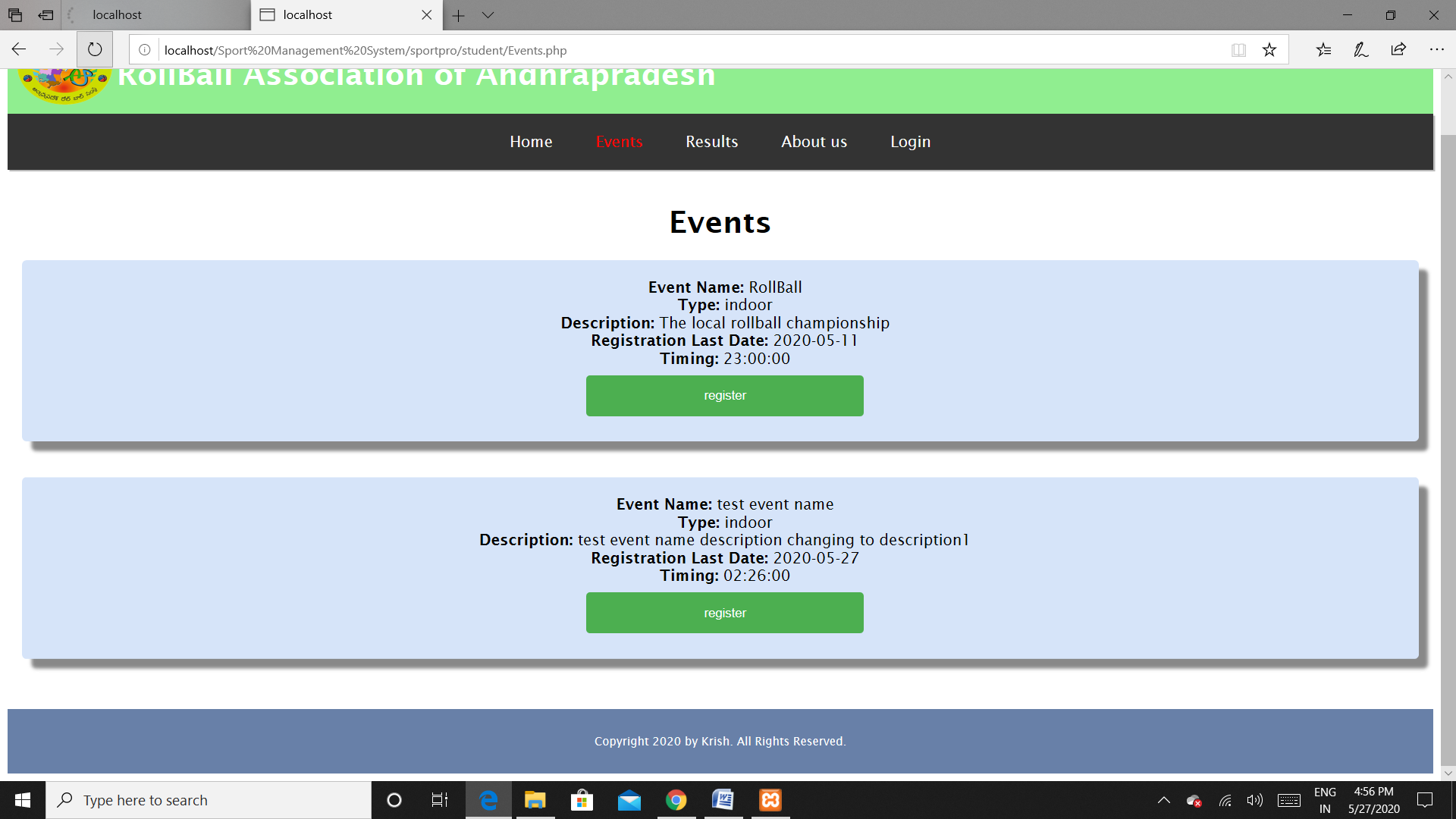
**Fig 6.3** players events page-successfully added by the admin.

****

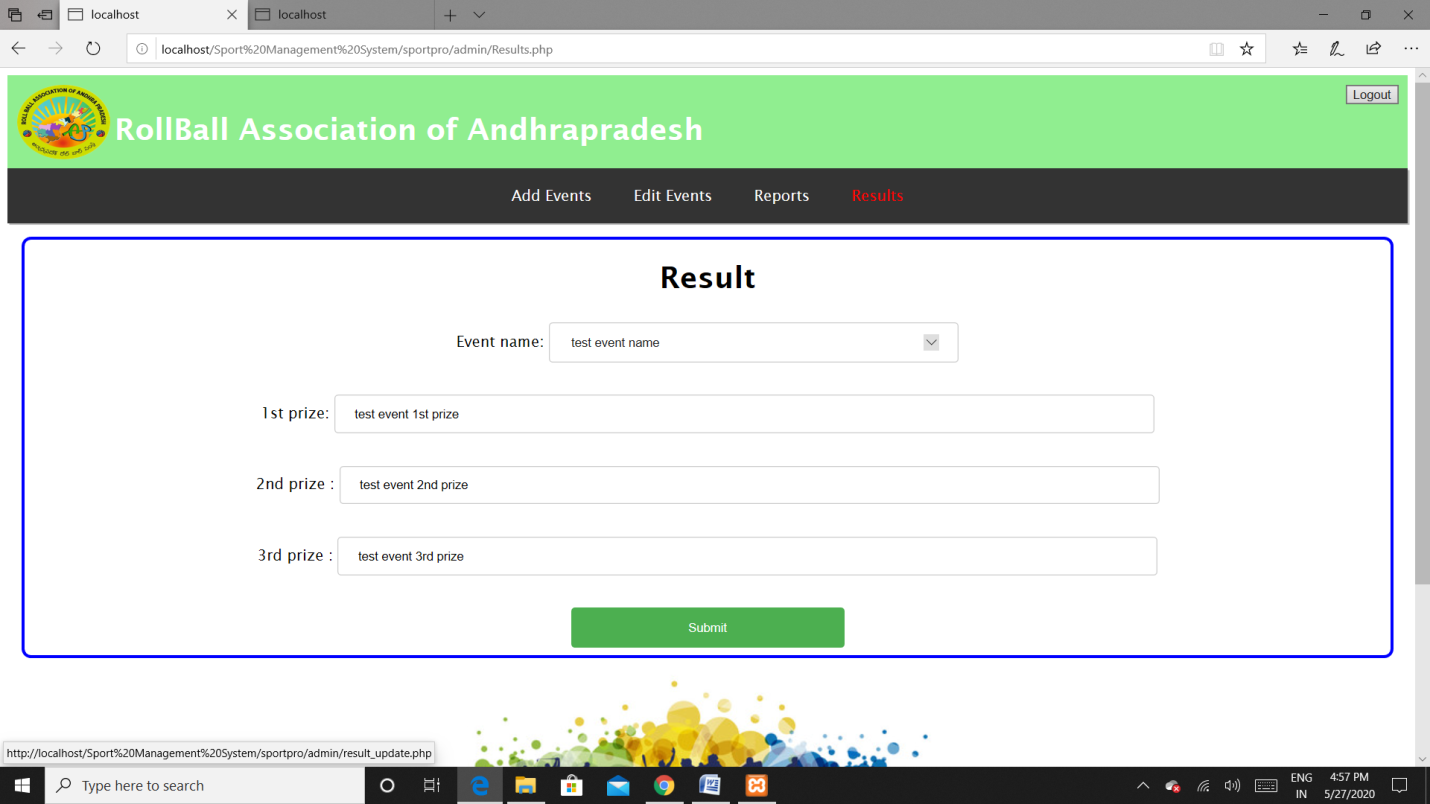
**Fig 6.4** admin edit events page

****

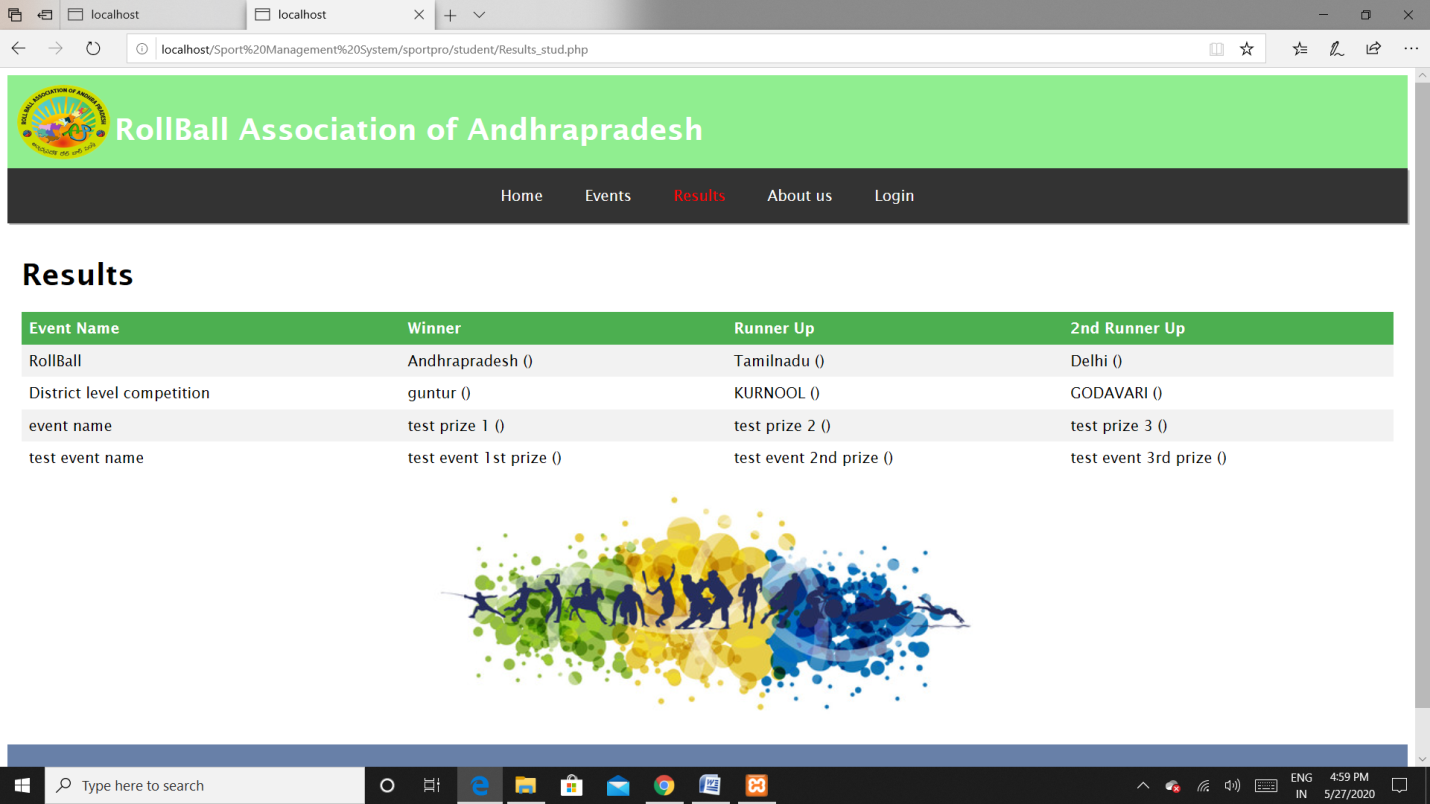
**Fig 6.5** admin edit events page confirmation

****

**Fig 6.6** players events page event description changed.

****

**Fig 6.7** admin results page

****

**Fig 6.8**  players results page successfully added prize winners.

**CHAPTER -7**

**CALCULATIONS**

**COCOMO Model:**

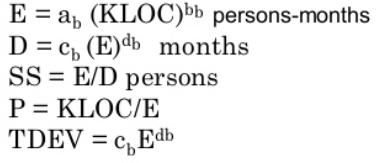
The total source lines of code in the project are 1558LOC=>1.558KLOC i.e., approximately to 2KLOC.

Most of the small to medium sized projects are estimated with Basic COCOMO model.

Since the total KLOC are approximately equal to 2, this is “Organic Mode”

So in the organic mode, the coefficients will be ab ,bb ,cb ,db

ab =2.4, bb =1.05, cb =2.5 db=0.38



Effort E= 2.4\*(1.558)1.05 =3.823

Deployment time D= 2.5\*(3.823)0.38 =4.1615

Staff Size SS = E/D = 3.823/4.1615 = 0.9186 ~ 1

Productivity P = KLOC/E = 1.558/3.823 = 0.4075

Development time TDEV = 2.5\*(3.823)0.38 =4.1615

**FP Computation:**

Assume the project is average complex system,

Therefore the complexity adjustment will be 3 so, ∑ fi =14× 3= 42

External Inputs => 14(email, password, name, contact, class, description, type, start date, end date, event name, event time, 3 prize winners.)

External Outputs=> 4(reports, results, events, login status.)

External Queries=> 1(registration enquiry)

Internal Logic Files=>6(admin login check, player login check, updating results, updating events, registration process, signup process.)

External Interface Files=> 17

Therefore count total value will be

|  |  |  |  |
| --- | --- | --- | --- |
| Functional units | Weight factors(complex) | Values | result |
| External inputs | 5 | 14 | 70 |
| External outputs | 7 | 4 | 28 |
| External Queries | 6 | 1 | 6 |
| Internal logic files | 15 | 6 | 90 |
| External Interface files | 10 | 17 | 170 |
| Count total============================🡺 | | | 364 |

Total FP = 364 × 42 = 15288.

Let the Burdened labor rate be $1000 per month and the average productivity 15 FP/pm.

Cost per lines of code = 1000/15 = $66.66

Total project cost = estimated fp × cost per fp

= 15288 × 66.66

= 1019200

People Effort = Total project cost/labor rate

= 1019200/1000

= 1019.2

**CHAPTER -8**

**REFERENCES**

1. [**https://www.geeksforgeeks.org/html-tutorials/**](https://www.geeksforgeeks.org/html-tutorials/)
2. [**https://www.w3schools.com/html/**](https://www.w3schools.com/html/)