CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION ABOUT DBMS

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.

A DBMS makes it possible for end users to create, read, update and delete data in a database. The DBMS essentially serves as an interface between the database and end users or application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified and the database schema, which defines the database's logical structure. These three foundational elements help provide concurrency, security, data integnty and uniform administration procedures. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

The DBMS is perhaps most useful for providing a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner. A DBMS can limit what data the end user sees, as well as how that end user can view the data, providing many views of a single database schema. End users and software programs are free from having to understand where the data is physically located or on what type of storage media it resides because the DBMS handles all requests.

The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data (storage and hardware).

1.2 INTRODUCTION TO MEMEBERSHIP MANAGEMENT SYSTEM:

Membership management ssystem is a specialized solution for centralizing administrative and communication activities of a membership organization. The software creates a digital space where community members can interact with managers and with each other, and where all organizational issues can be solved

Online membership management is applicable across industries such as sports, healthcare, wellness and beauty, charity, education and more.The core objective of such a solution is to share information between community members without delays, thus streamlining all organizational processes. At the same time, its capabilities can be tailored to the specific needs of certain businesses, be it a gym or a writers association. The core objective of such a solution is to share information between community members without delays, thus streamlining all organizational processes. At the same time, its capabilities can be tailored to the specific needs of certain businesses, be it a gym or a writers association.

Essential Features of Membership Software Solutions:

Club or community managers have to take on a range of organizational duties such as sharing information, contacting members, gathering feedback, scheduling events, processing payments and much more.Doing all these manually wastes a lot of time. By taking over these menial tasks, club software can relieve managers of these duties.

- **Online registration form** — users can join a club with just a couple of clicks.
- **Member profiles** — member account details are held in a centralized database that's updated automatically, so managers can easily search for information and contact people if needed.
- **Content management** — community owners can build full-fledged websites or applications, where they can create, edit and deliver relevant content to their audience.
- **Mobile optimization** — a club website should be mobile responsive, and there should be mobile application to allow users to manage activities on the go.

1.3 PROBLEM STATEMENT

Registering to the college clubs will be made easy so that students can be able to register the clubs that are present in AIET and admin is now able to add the clubs. Users can view the details of club members and also about the club. Users can send their queries and also the feedback to admin

1.3 MOTIVATION AND OBJECTIVE OF THE PROJECT

The aim of the project is to analyze the current club system in AIET and suggest an offline membership management system which will allow student to register any of the clubs in AIET in a more convenient way, and to know more information of clubs resources , current status of the student in club system

1.4 PROPOSED SOLUTIONS AND ADVANTAGES

Online voting allows people in today's mobile and digitally advanced society to participate in the democratic process over the internet. The POLYAS online voting system offers the highest levels of transparency, control, security and efficiency of election processes. Online elections provide voters with a comfortable and secure voting experience and allow election organizers to save resources in planning their next election. For election organizers, planning postal or ballot box elections is synonymous with high costs: you need to find an appropriate physical space to hold the election, inform eligible voters, prepare postal voting forms and organize the vote count.

Give yourself maximum flexibility and efficiency in election planning and create ballots easily online. Send out election invitations and reminders conveniently via email. Observe voter turnout in real-time and engage voters with a few mouse clicks. Easy ballot creation and personal election invitation design, as well as automatic vote counting, greatly simplifies election organization.

CHAPTER 2

SYSTEM DESIGN

2.1 SCHEMA DIAGRAM

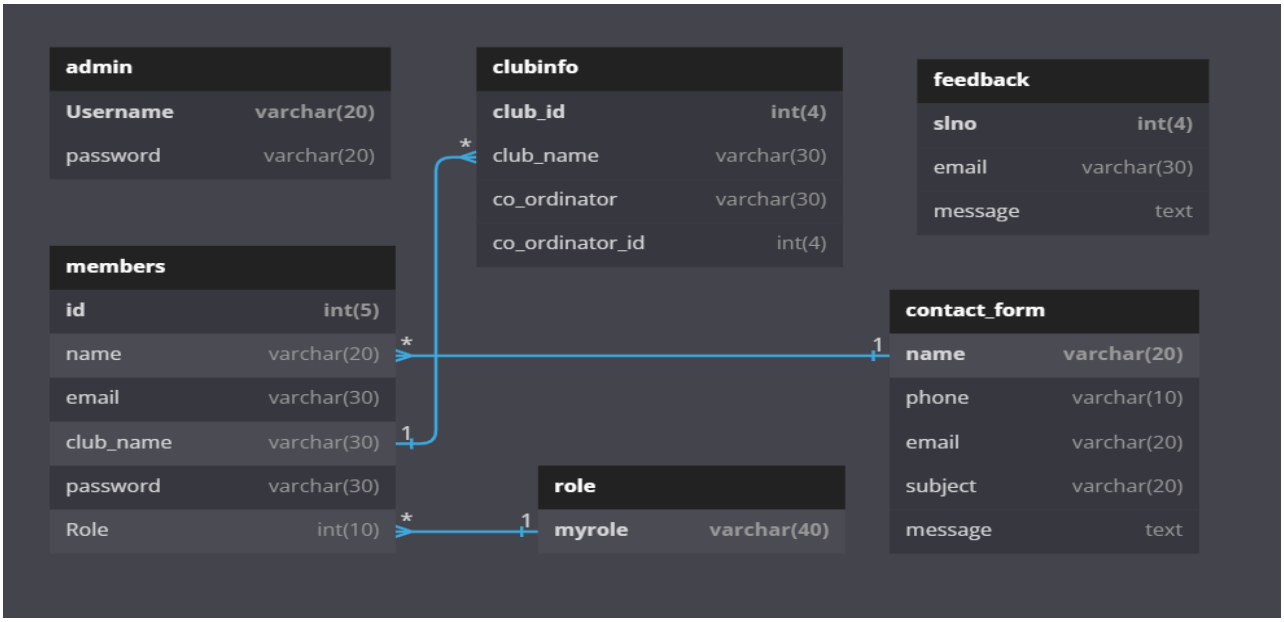

Figure 2.1 Schema Diagram

The Figure 2.1 is the schema diagram for membership management system. It shows the relationship between different table attribut

A database schema specifies, based on the database administrator's knowledge of possible applications, the facts that can enter the database, or those of interest to the possible end-users The notion of a database schema plays the same role as the notion of theory in predicate calculus. A model of this "theory" closely corresponds to a database, which can be seen at any instant of time as a mathematical object. Thus a schema can contain formulas representing integrity constraints specifically for an application and the constraints specifically for a type of database, all expressed in the same database language

2.2 E R DIAGRAM .

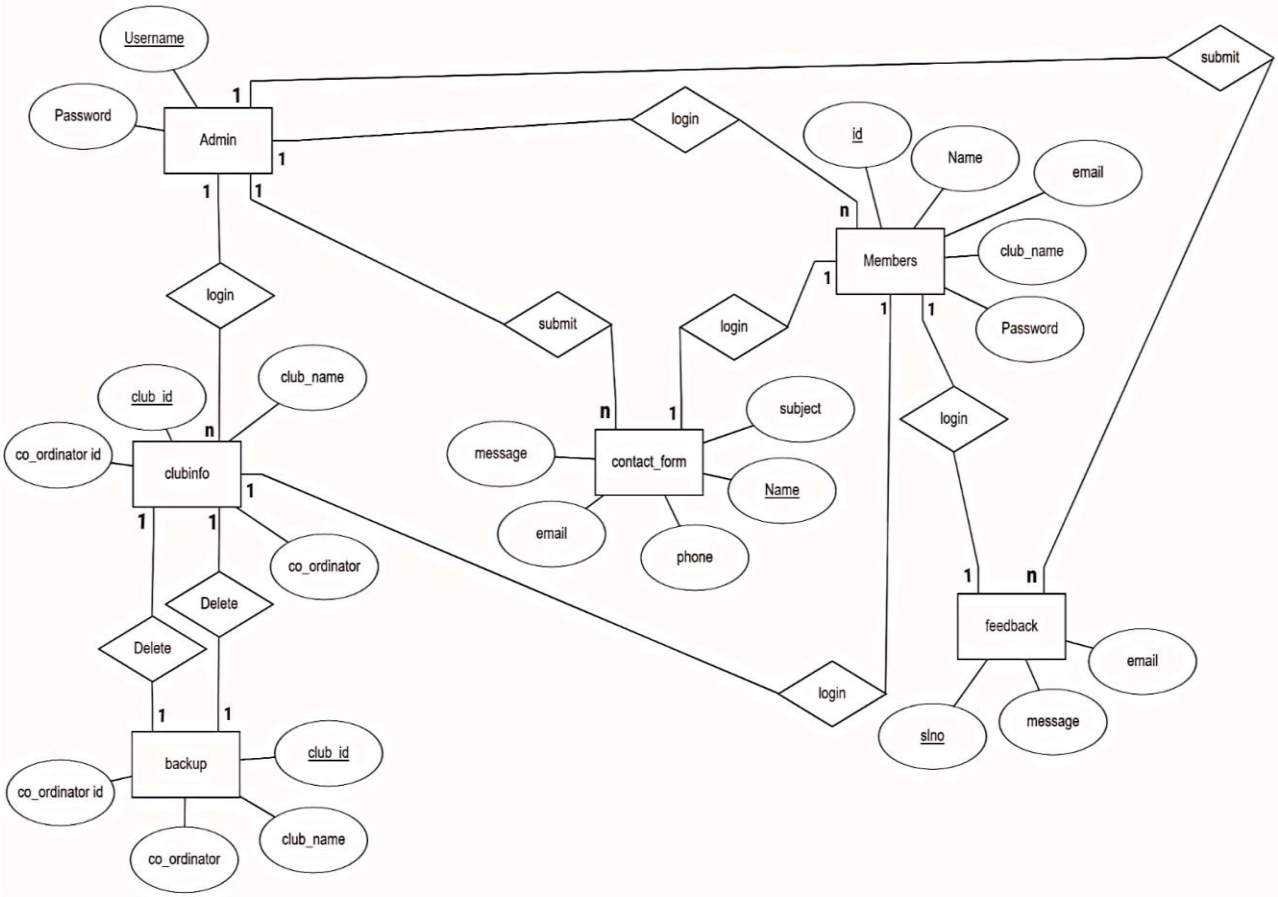

Figure 2.2 E R Diagram

This figure 2.2 shows the E R diagram of online voting system.

An entity-relationship model (ER model) describes inter-related things of Interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types. ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure that can be emplemented in a database, typically a relational database. The main components of ER model are: entity

set and relationship set. Here are the geometric shapes and their meaning in an ER Diagram Rectangle Represents Entity sets.

Ellipses : Attributes.

Diamonds: Relationship set.Lines : They link attributes to Entity Sets and this to Relationship Set.

CHAPTER 3

IMPLEMENTATION

In this chapter the implementation details of the project have been specified.

3.1 HARDWARE SPECIFICATIONS:

- 40 GB hard disk space.
- 2 GB RAM.
- Hi-Speed Network Connectivity.

3.2 SOFTWARE SPECIFICATIONS:

- Windows(x64) Operating System.
- MySQL Server.
- Apache Server.
- Xampp
- Vs code

3.2.1 LANGUAGE USED FOR IMPLEMENTATION

The languages used for implementation are as follows:

- Front end: - HTML ,CSS (Tails wind) ,JAVASCRIPS,JQUERY
- Back end: - MySQL , PHP

HTML: -

Hypertext Mark-up Language (HTML) is the standard mark-up language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from local storage and render them into multimedia web pages.

JAVASCRIPS :-

JS is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

JQUERY :-

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax tis free, open-source software using the permissive MIT License. As of Aug 2022, jQuery is used by 77% of the 10 million most popular websites Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having at least 3 to 4 times more usage than any other JavaScript library

MySQL: -

MySQL is an open-source relational database management system(RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality.

PHP: -

PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Leadoff in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym. PHP code may be embedded into HTML or HTML5 mark up, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical.

3.2.2 PLATFORM USED FOR IMPLEMENTATION

XAMPP

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes. Everything needed to set up a web server – server application (Apache), database (MariaDB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well.

PhpMyAdmin

PhpMyAdmin is a free and open source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services.

Visual Studio Code :-

Visual Studio Code, also commonly referred to as **VS Code**,[10] is a source-code editor made by Microsoft with the Electron Framework, for Windows, Linux and macOS.[11] Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality In the Stack Overflow 2021 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool among 82,000 respondents, with 70% reporting that they use it.

3.3 SQL COMMANDS AND QUERIES

The queries used for creating these tables are as follow:

```
CREATE TABLE `admin` (
`Username` varchar(20) NOT NULL,
`password` varchar(20) NOT NULL)
ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

Table structure for table `clubinfo`

```
CREATE TABLE `clubinfo` (
  `club_id` int(4) NOT NULL,
  `club_name` varchar(30) NOT NULL,
  `co_ordinator` varchar(30) NOT NULL,
  `co_ordinator_id` int(4) NOT NULL) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci
```

Table structure for table `contact_form`

```
CREATE TABLE `contact_form` (
  `name` varchar(20) NOT NULL,
  `phone` varchar(10) NOT NULL,
  `email` varchar(20) NOT NULL,
  `subject` varchar(20) NOT NULL,
  `message` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

Table structure for table `feedback`

```
CREATE TABLE `feedback` (
  `slno` int(4) NOT NULL,
  `email` varchar(30) NOT NULL,
  `message` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

Table structure for table `members`

```
CREATE TABLE `members` (
  `id` int(5) NOT NULL,
  `name` varchar(20) NOT NULL,
  `email` varchar(30) NOT NULL,
  `club_name` varchar(30) DEFAULT NULL,
  `password` varchar(30) NOT NULL,
  `Role` int(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

Table structure for table `role`

```
CREATE TABLE `role` (
  `myrole` varchar(40) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_general_ci;
```

Indexes for dumped tables

Indexes for table `admin`

```
ALTER TABLE `admin` ADD PRIMARY KEY (`Username`);
```

Indexes for table `clubinfo`

```
ALTER TABLE `clubinfo` ADD PRIMARY KEY (`club_id`);
```

Indexes for table `contact_form`

```
ALTER TABLE `contact_form`ADD PRIMARY KEY (`name`);
```

Indexes for table `feedback`

```
ALTER TABLE `feedback` ADD PRIMARY KEY (`slno`);
```

Indexes for table `members`

```
ALTER TABLE `members`ADD PRIMARY KEY (`id`);
```

Indexes for table `role`

```
ALTER TABLE `role`
ADD PRIMARY KEY (`myrole`);
```

AUTO_INCREMENT for dumped tables

AUTO_INCREMENT for table `feedback`

```
ALTER TABLE `feedback`
MODIFY `slno` int(4) NOT NULL AUTO_INCREMENT;
```

AUTO_INCREMENT for table `members`

```
ALTER TABLE `members`
 MODIFY `id` int(5) NOT NULL AUTO_INCREMENT;
 COMMIT;
```

3.4 TRIGGERS

```
CREATE TRIGGER `mybackup` BEFORE DELETE ON `clubinfo`
FOR EACH ROW INSERT INTO backup
VALUES(OLD.club_id,OLD.club_name,OLD.co_ordinator,OLD.co_ordinator_id)
```

```
CREATE TRIGGER `restore` BEFORE DELETE ON `backup`
 FOR EACH ROW INSERT INTO clubinfo VALUES (OLD.club_id,
old.club_name,old.co_ordinator,old.co_ordinator_id)
```

3.5 PHP CODES

```
<?php
$server = "localhost";
$username = "root";
$password = "";
$dbname = "aietclub";

// Create connection
$conn = mysqli_connect($server, $username, $password, $dbname);

// Check connection
if (!$conn) {
die("Connection failed: " . mysqli_connect_error());
}

if (isset($_POST['login'])) {
$username = $_POST['username'];
$password = $_POST['password'];

$query = "SELECT * FROM admin WHERE username='$username' AND
password='$password'";
$result = mysqli_query($conn, $query);
if (mysqli_num_rows($result) == 1) {
// Login success
session_start();
$_SESSION['loggedin'] = true;
header('location: ../home/home.php');
} else {
```

```
// Login failed
$error = "Invalid username or password";
}
}

if(isset($_POST['createacc'])){
session_start();
header('location: ../create_account/create_account.php');
}
?>
<!DOCTYPE html>
<html>
<head>
<title>Admin Login</title>
<script src="https://cdn.tailwindcss.com"></script>
<link rel="stylesheet" href="style.css">
<link
rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css"
/>
</head>
<body class="bg-[url('../images/college.jpg')]">
<!-- <form method="post" action="">
<label for="username">Username:</label><br>
<input type="text" id="username" name="username"><br>
<label for="password">Password:</label><br>
<input type="password" id="password" name="password"><br><br>
<input type="submit" name="login" value="Login">
</form>  -->
<form action="" method="post">
<div class="flex justify-center animate_animated animate_jello" id="container">
<div class="flex flex-col justify-center">
```

```
<div class="flex flex-col md:flex-row max-w-7xl justify-center items-center ">
<div class="overflow-hidden w-full m-4 flex justify-center bg-gray-50 rounded-lgshadow-
xl">
<div class="flex flex-col md:flex-row items-center shadow-md h-full  ">
<div class="  md:w-1/2 overflow-hidden ">
<div class="flex flex-col items-center justify-center text-stone-400">
        <!-- <ion-icon name="logo-amplify" class="text-5xl text-fuchsia-600"></ion-icon>
        -->
<div class=" md:w-1/6 overflow-hidden ">
<img src="../images/aiet-logo.png" alt="" class="" />
</div>
<div class="flex flex-col ">
<div class="m-2">USERNAME</div>
<input class="border-b m-2  bg-gray-50 focus:outline-none" type="text" id="username"
name="username" />
<div class="m-2">PASSWORD</div>
<input class="border-b m-2  bg-gray-50  focus:outline-none" type="password"
id="password" name="password"/>
        <?php
if (isset($error)) {
echo $error;
}
?>
<!-- <div class="flex m-2">
<input class="border-b  border-stone-400 " type="checkbox" />
<div class="ml-1">Remember Me</div>
</div> -->
<div class="flex m-2">
<input
class="bg-gradient-to-l from-fuchsia-600 to-cyan-400 rounded-2xl px-6 py-1 text-white
font-medium" type="submit" name="login" value="Login" />
<!-- <input
```

```
class="text-transparent  bg-clip-text bg-gradient-to-l from-fuchsia-600 to-cyan-400 font-bold ml-2 border-2 rounded-2xl px-6 border-cyan-400" name="createacc" type="submit" value="Create new Account" /> -->
</div>
<!-- <div class="m-2">Forgotten your login details?</div>
<div class="font-medium ml-2">Get Help Signing In</div> --
</div>
</div>
</div>
<div class=" md:w-1/2 overflow-hidden ">
<img src="../images/Aiet-College.jpg" alt="" class="" />
</div>
</div>
</div>
</div>
</div>
</div>
<script type="module"
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>
<script nomodule
src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>
</form>
</body>
</html>
```

3.4 OUTPUT TESTING

- While executing php MySQL connection code we were not able to make the connection of backend MySQL to front end php. So, to solve this problem we had to create a new MySQL user with password. After this the connection was successful.
- The connection was successful but the data entered in front end was not storing in backend, since all the attributes data types in backend were not set to varchar. So, we modified the php code and MySQL query accordingly.
- If we enter wrong password, local host says wrong password. If XAMPP server is not started, then we cannot run local host.

CHAPTER 4

RESULT

4.1 SNAPSHOTS

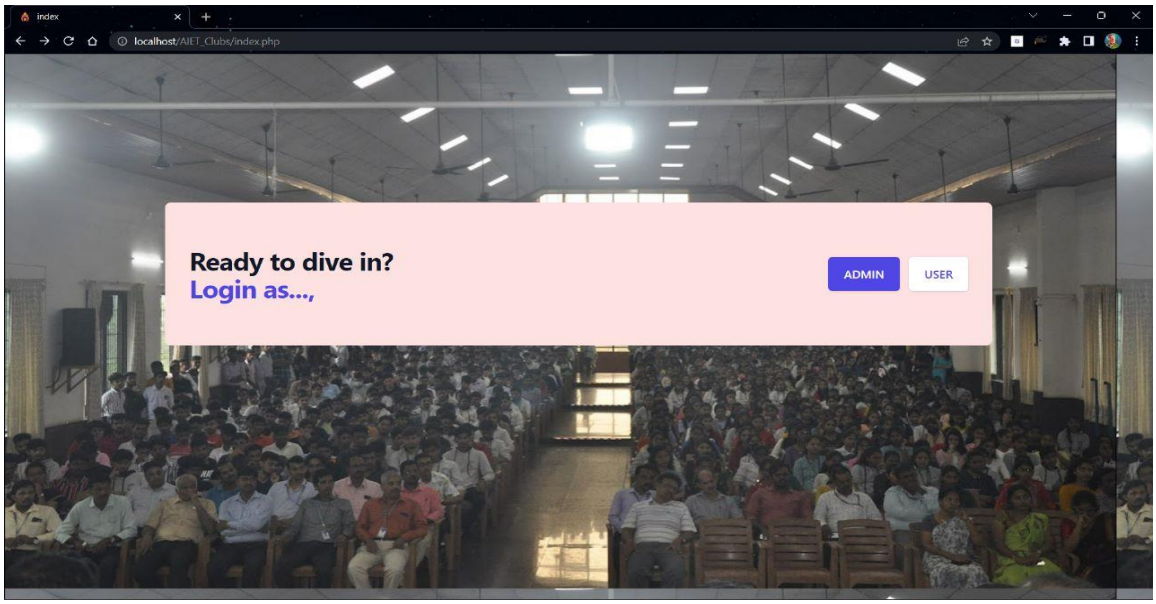

Figure 4.1.1 front Page

Here it shows front page of the membership management of AIET. when you click that it will redirect to the selection login.\ form

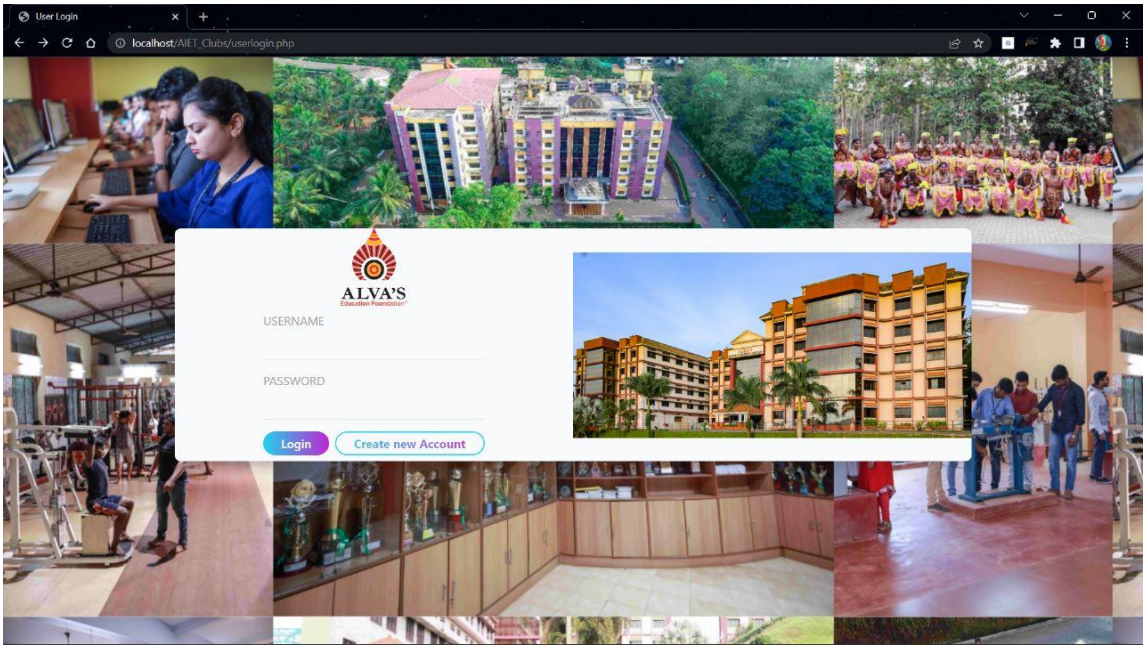

Figure 4.1.2 : user login page

Here it shows the user login page of AIET .where user can create a new account And can login to the pagess

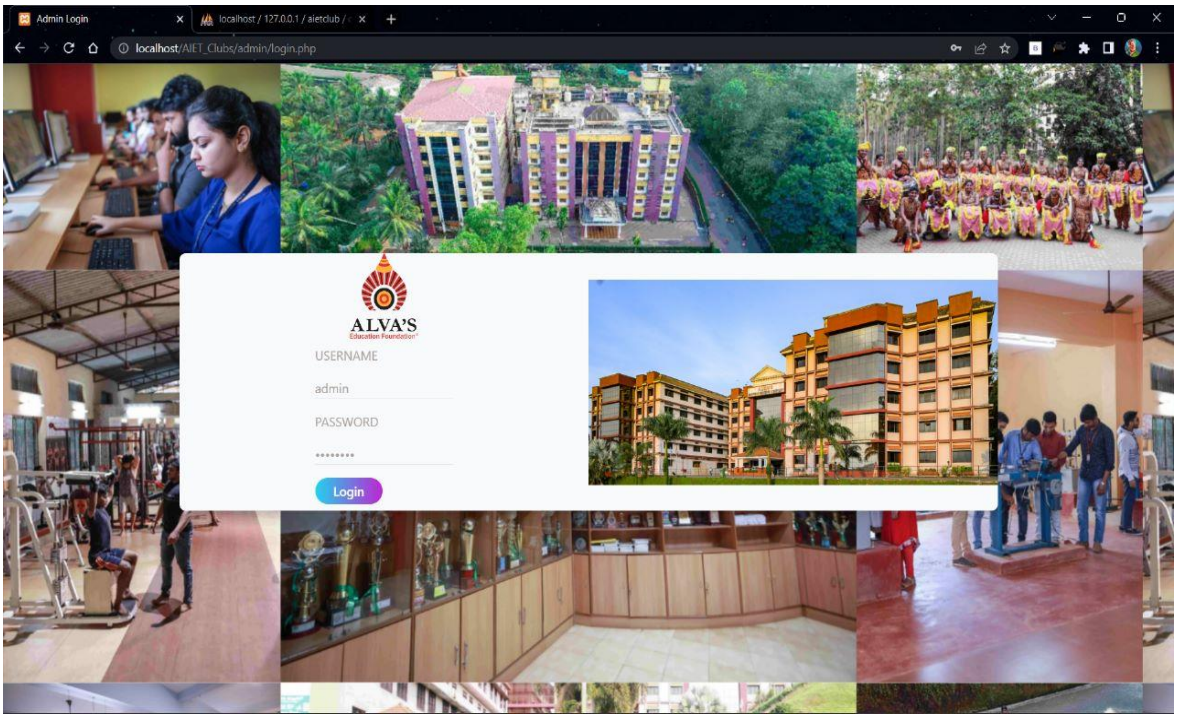

Figure 4.1.3 : admin login

Here it shows Admin login of AIET membership where admin can login to page

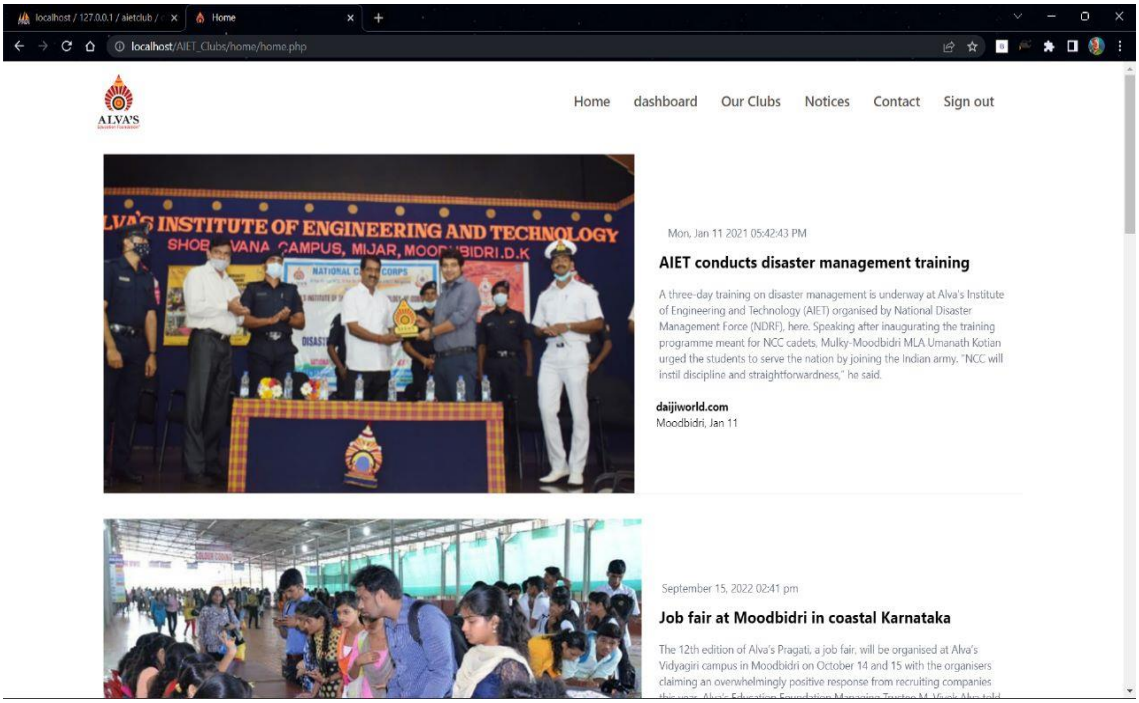

Figure 4.1.4 home page

The above figure 4.1.4 show the home page where some description are contained about the activities

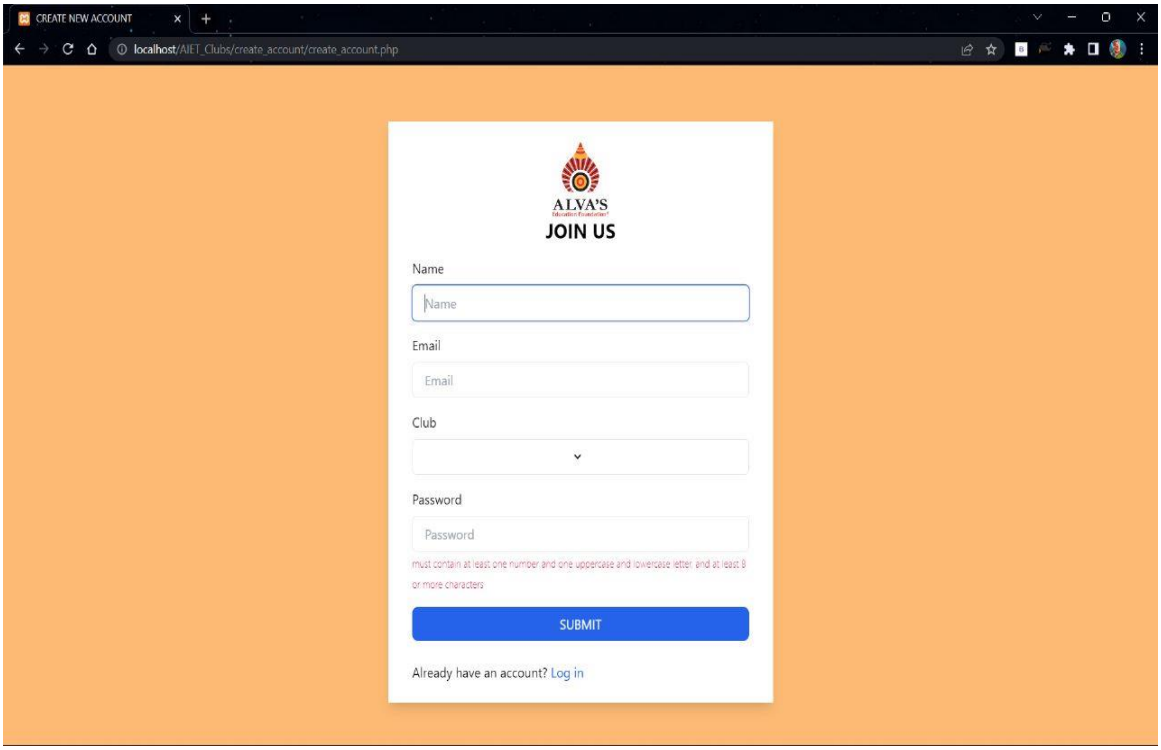

Figure 4.1.5 sign up page

The above figure 4.1.5 shows user can create a new account by entering the details

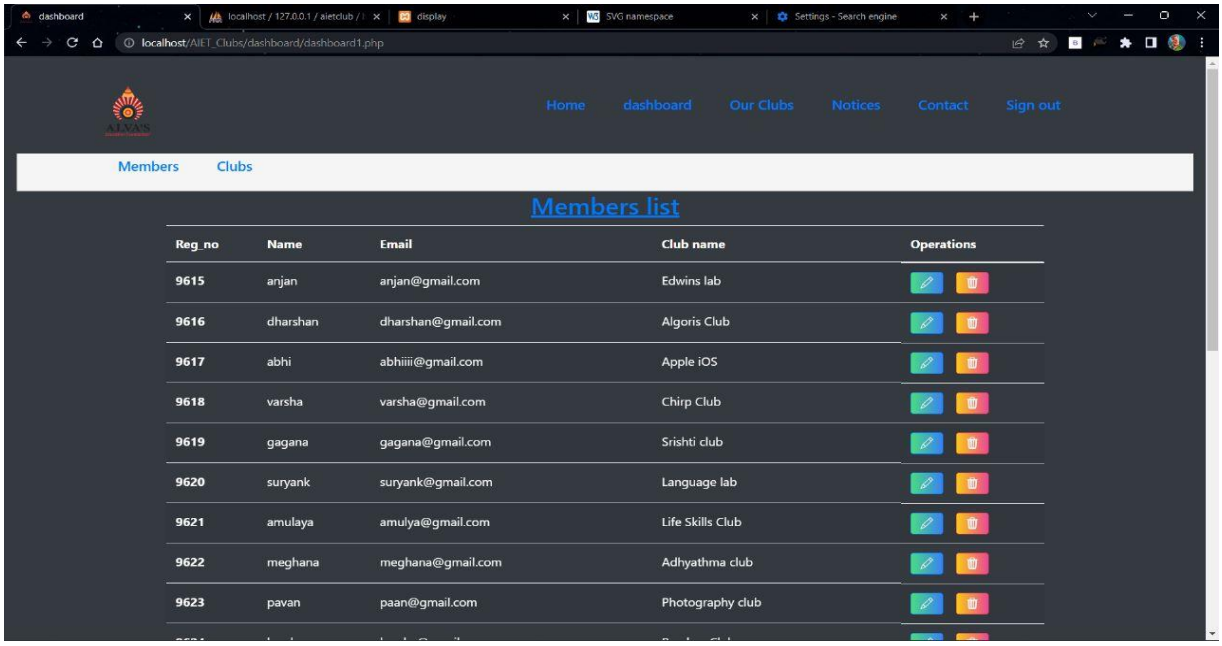

Figure 4.1.6 : member list

The above figure 4.1.6 shows the member of user and give some information about users

And this is only for admin where admin can update or delete the users

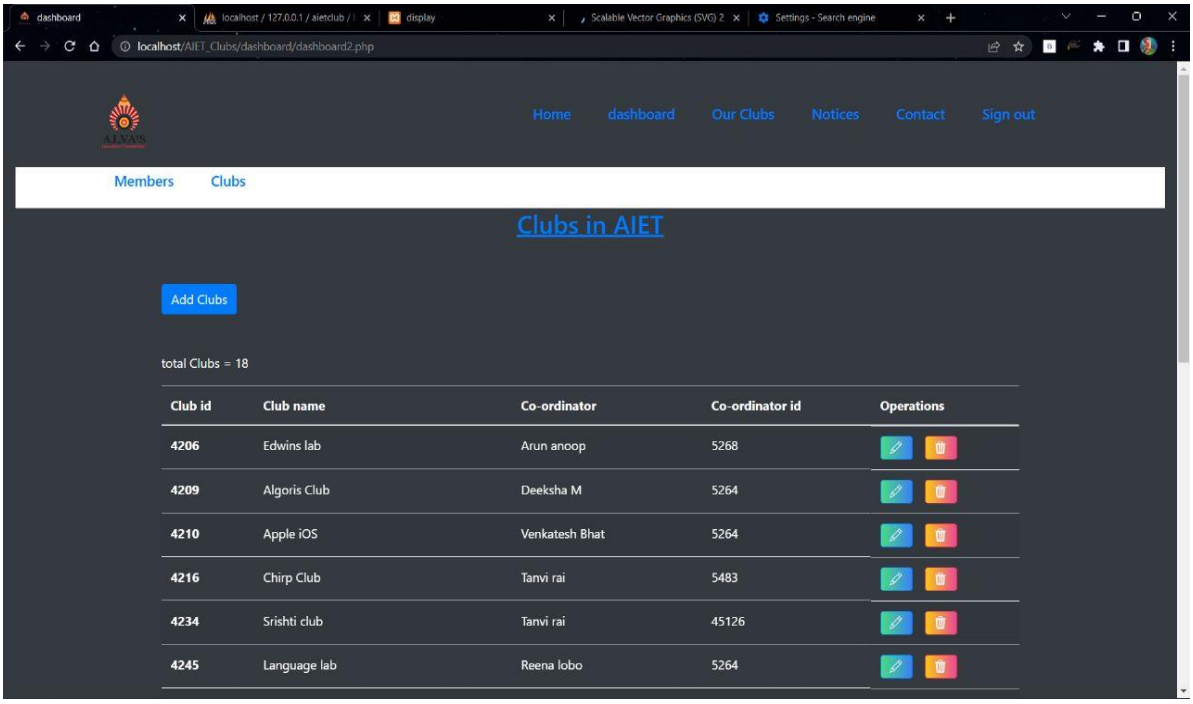

Figure 4.1.7: clubs list

The above figure 4.1.7 shows the club list where admin can add new clubs , can modify or delete the clubs

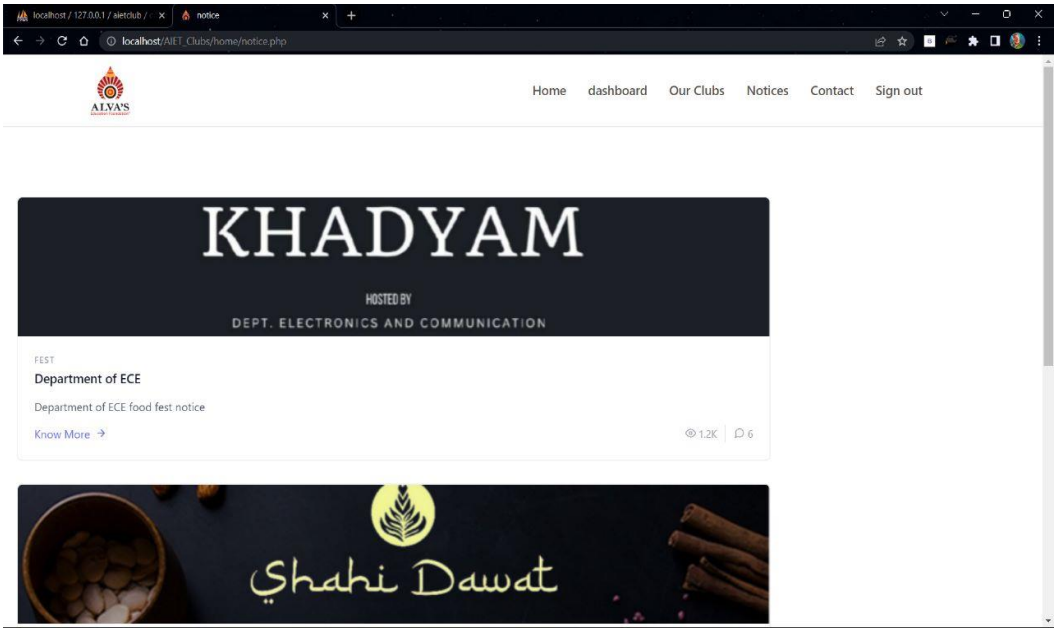

Figure 4.1.8: notices pages

The above figure 4.1.8 shows the notices about events

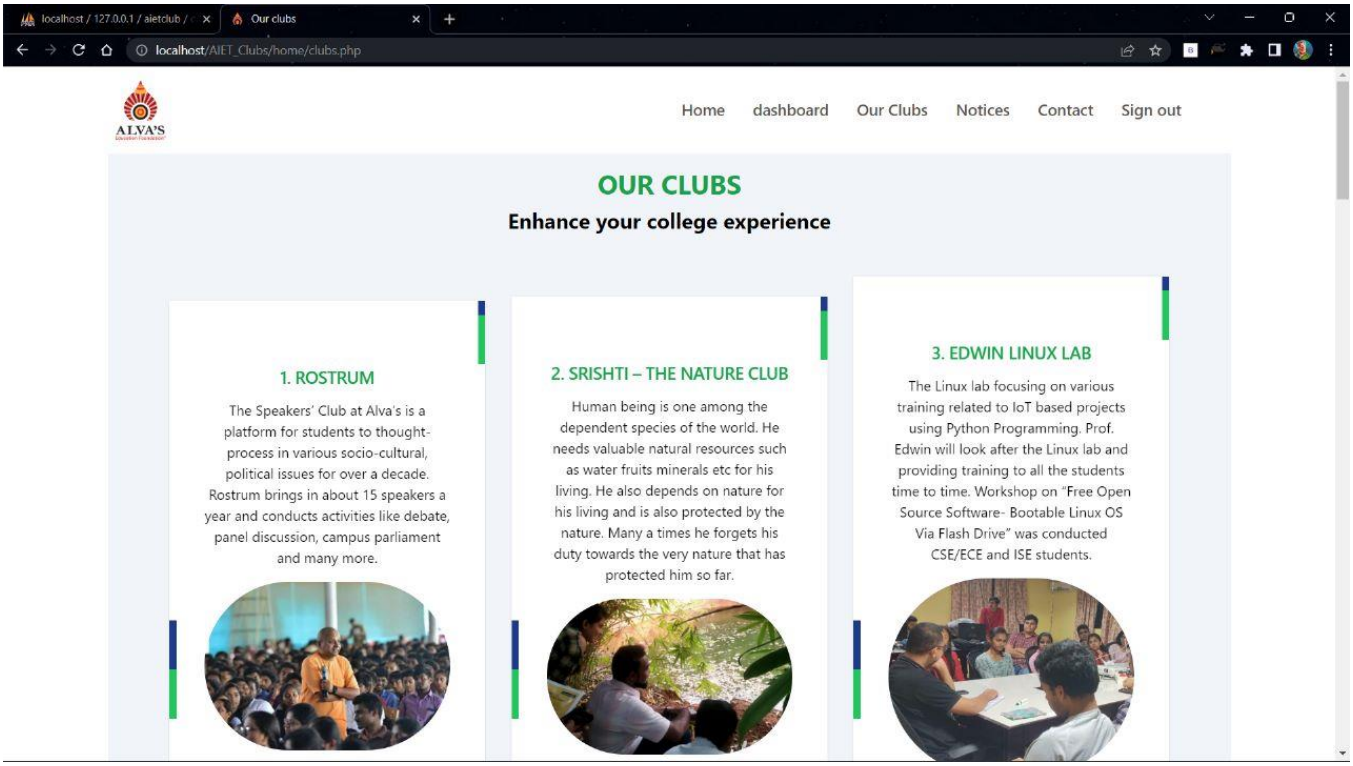

Figure 4.1.9 : Descriptions about clubs

The above figure 4.1.9 shows the description of clubs and give more information about the clubs

CHAPTER 5

CONCLUSION AND FUTURE ENHANCEMENT

In this chapter the project is concluded and its future enhancement has been specified.

5.1 CONCLUSION

The main aspect behind Online voting system is that it enabled us to bring out the new ideas that were sustained within us for many days. This project offers the voters to cast their votes easily. Vote counting is also made easy by online voting system since it’s just a matter of querying the database. Online voting system is used by a number of countries today. Developing a good system is critical to the success of the system to prevent system failures and to gain wide acceptance as the best method available. The project that we have built is a prototype and, in the future, can be enhanced by adding features some basic features such as retrieving of the user password in case if forgotten. Some advanced technologies that can be implemented is blockchain technology which we are definitely looking forward to it.

5.2 FUTURE ENHANCEMENT

The Club membership system can be made more secure by using the following methods

- Password Changing
- Can able to register club activities
- The annual club events schedule will be displayed in the site

REFERENCES

1. www.aiet.org.in
2. https://www.w3schools.com/
3. https://stackoverflow.com/questions/tagged/php
4. https://animate.style/
5. https://tailblocks.cc/