VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, BELAGAVI – 590018

A MINI PROJECT REPORT

ON

“DEPARTMENT ASSOCIATION”

Submitted in partial fulfilment of requirements for the *course* **DBMS Laboratory with Mini Project [18CSL58]** *of Fifth Semester of Bachelor of Engineering in Computer Science & Engineering during the academic year 2021-22.*

Submitted By

“Mohammed Mudassir”	**“Ronish J Gowda”**
“4MH19CS125”	**“4MH19CS080”**

Under the Guidance of

“Prof. HEMANTH S R”

Assistant Professor,
Dept. of CS&E,

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna (T), Mandya (D) – 571477.

2021 - 2022

MAHARAJA INSTITUTE OF TECHNOLOGY MYSORE

Belawadi, S.R. Patna (T), Mandya (D) – 571477.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the mini project work entitled **“DEPARTMENT ASSOCIATION MANAGEMENT SYSTEM”** is a bonafide work carried out by **Mohammed Mudassir [4MH19CS125]** and **Ronish J Gowda [4MH19CS080]** in partial fulfilment for the **DBMS Laboratory with Mini Project (18CSL58)** prescribed by the Visvesvaraya Technological University, Belagavi during the year 2021-2022 for the fifth semester B.E in Computer Science and Engineering. The mini project report has been approved as it satisfies the academic requirements.

Signature of Guide	Signature of HOD
(**Prof. HEMANTH S R**)	(**Dr. Shivamurthy R C**)
Assistant Professor, Dept. of CS&E	Professor & Head, Dept. of CS&E
MIT Mysore	MIT Mysore

Name of the Examiners	**Signature with date**
1...................................... 2......................................	…............................... …...............................

ACKNOWLEDGEMENT

We sincerely owe our gratitude to all the persons who helped and guided us in completing this mini project work.

We are thankful to **Dr. B.G. Naresh Kumar, Principal, Maharaja Institute of Technology Mysore**, for having supported us in our academic endeavours and for providing basic facilities and necessary requirements for this project.

We are extremely thankful to **Dr. Shivamurthy R C, Professor & Head, Department of Computer Science and Engineering,** for allowing us to carry out this project and his timely inquiries into the progress of the work. His suggestion and his instructions have served as the major contributor towards the completion of the project.

We are greatly indebted to our guide **Prof. Hemanth S R, Assistant Professor, Department of Computer Science and Engineering,** for the invaluable guidance, constant encouragement, immense motivation, which has sustained our efforts at all the stages of this project.

We are obliged to all **teaching and non-teaching staff members** of **Department of Computer Science and Engineering,** for the valuable information provided by them in their respective field's. We are grateful for their co-operation during the period of our mini project.

Mohammed Mudassir (4MH19CS125)
Ronish J Gowda (4MH19CS080)

ABSTRACT

The overall aim of Department Association Management System is to automate the existing manual system by the help of computerized equipment and fully-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Department Association Management System, as described above, can lead to error free, secure, reliable & fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The main objective of the project is to provide the list of events containing workshop, online livestreams, projects etc. which is useful for students and institutions for remind them about the ongoing events. The system is intended for the student, faculty. The privileges that are provided to user is to view information of the events by providing their basic credentials. The whole Department Association will be under the control of the administrator and the admin as the full privileges to insert, update, modify, delete the events. And admin gives the privileges to the student to access the events.

~~~~~ TABLE OF CONTENTS ~~~~~

Chapter 01

INTRODUCTION

1.1 Aim of the Project

The aim of this project is to make it easy for faculties and students to access event details and also conduct events by providing credentials. Event details are fetched and stored in the database.

1.2 Overview of the Project

Department Association is a management system where department admin can conduct events and remind date and time of the particular event to all type of users of the system. It is an application which will automate a lot of activities in a college.

1.3 Outcome of the Project

This website in particular domain the main outcome of this project is to help users to get reminders and information easily and discussing the queries with the admins on the website, more the interests shared by the user so that more events can be conducted. Our website provides admins to delete the event, update and modify the events.

Information sorted by date and time which will displayed in admins page. This helps the user with all the content whatever required mostly websites do not provide this. To access it first user, have to create an account or sign up then they have to login. Database is created in which the user’s registration and other details are saved.

1.4 Software Requirements

- Operating System – Windows 7/8/10/11
- Front End – HTML, CSS, JS
- Back End – mySQL
- Tools – Xampp Server

Chapter 02

DESIGN

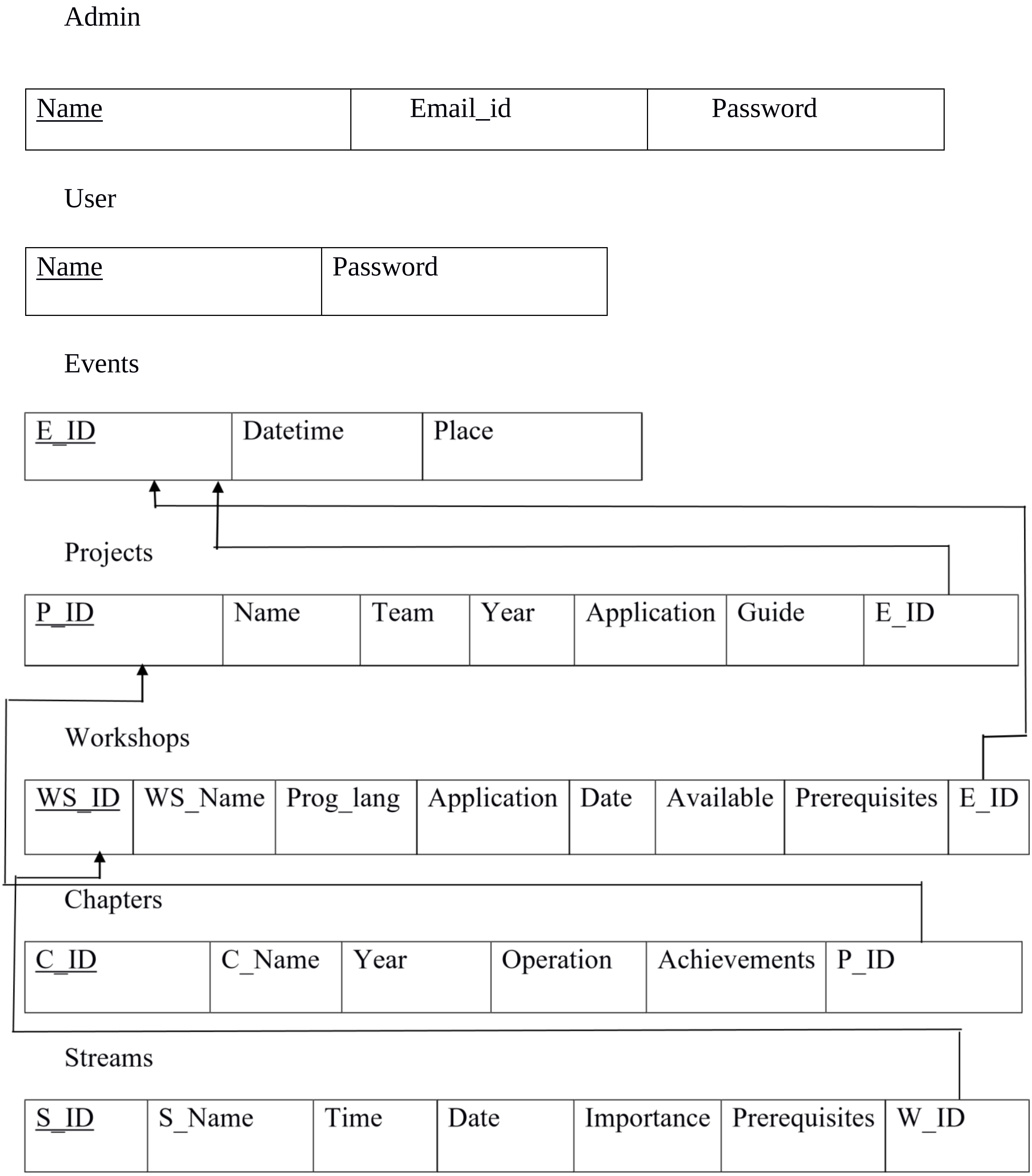
2.1 Schema Diagram

The design of the database is called a schema. This tells us about the structural view of the database. It gives us an overall description of the database. A database schema defines how the data is organized using the schema diagram. A schema diagram is a diagram which contains entities and the attributes that will define that schema. A schema diagram only shows us the database design. It does not show the actual data of the database.

Here in Department Association database, there are 7 tables namely admin, user, events, projects, workshops, chapters and streams.

- Admin table has 3 attributes such as name, Email_id and Password.
- User table has 2 attributes such as Name and password.
- Events table has 3 attributes such as E_ID (Event ID), Datetime and Place.
- Projects table has 7 attributes such as P_ID (Project ID), Name, Team, Year, Application, Guide and E_ID.
- Workshops table has 8 attributes such as WS_ID (Workshop ID), WS_Name (Workshop Name), Prog_lang, Application, Date, Available, Prerequisites, E_ID.
- Chapters table has 6 attributes such as C_ID (Chapter ID), C_Name (Chapter Name), Year, Operation, Achievements, P_ID.
- Streams table has 7 attributes such as S_ID (Stream ID), S_Name (Stream Name), Time, Date, Importance, Prerequisites, W_ID.

- Events table is related to Projects and Workshops tables.
- Projects table is related to Chapters table.
- Workshops table is related to Streams table.



Admin

Name	Email_id	Password

User

Name	Password

Events

E_ID	Datetime	Place

Projects

P_ID	Name	Team	Year	Application	Guide	E_ID

Workshops

WS_ID	WS_Name	Prog_lang	Application	Date	Available	Prerequisites	E_ID

Chapters

C_ID	C_Name	Year	Operation	Achievements	P_ID

Streams

S_ID	S_Name	Time	Date	Importance	Prerequisites	W_ID

Fig 2.1 Schema Diagram for Department Association

2.2 Entity Relationship Diagram

ER-Diagram is a pictorial representation of data that describes how data is communicated and related to each other. Any object, such as entities, attributes of an entity, sets of relationship, and other attributes of relationship, can be characterized with the help of the ER diagram.

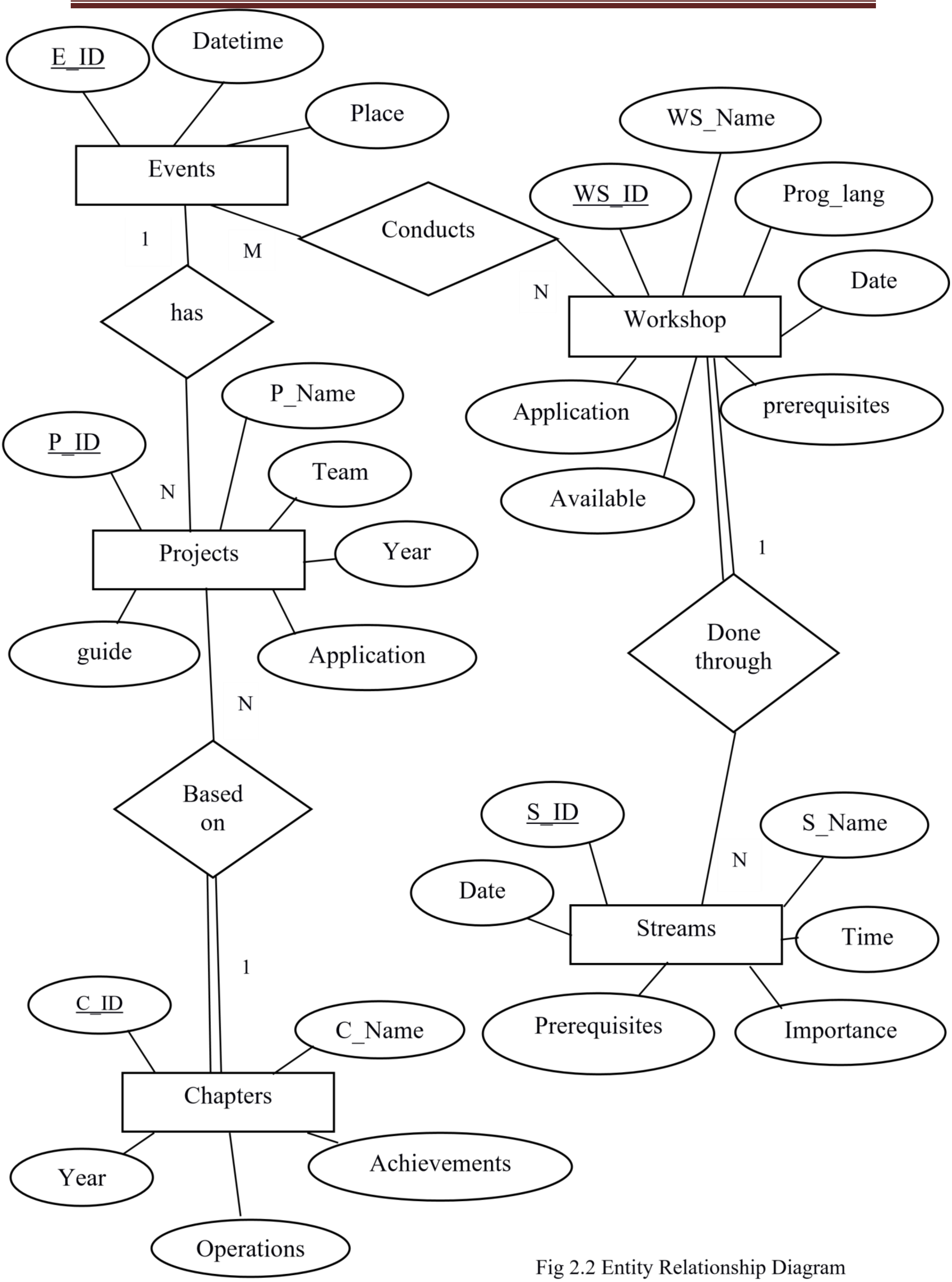


Fig 2.2 Entity Relationship Diagram

2.3 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

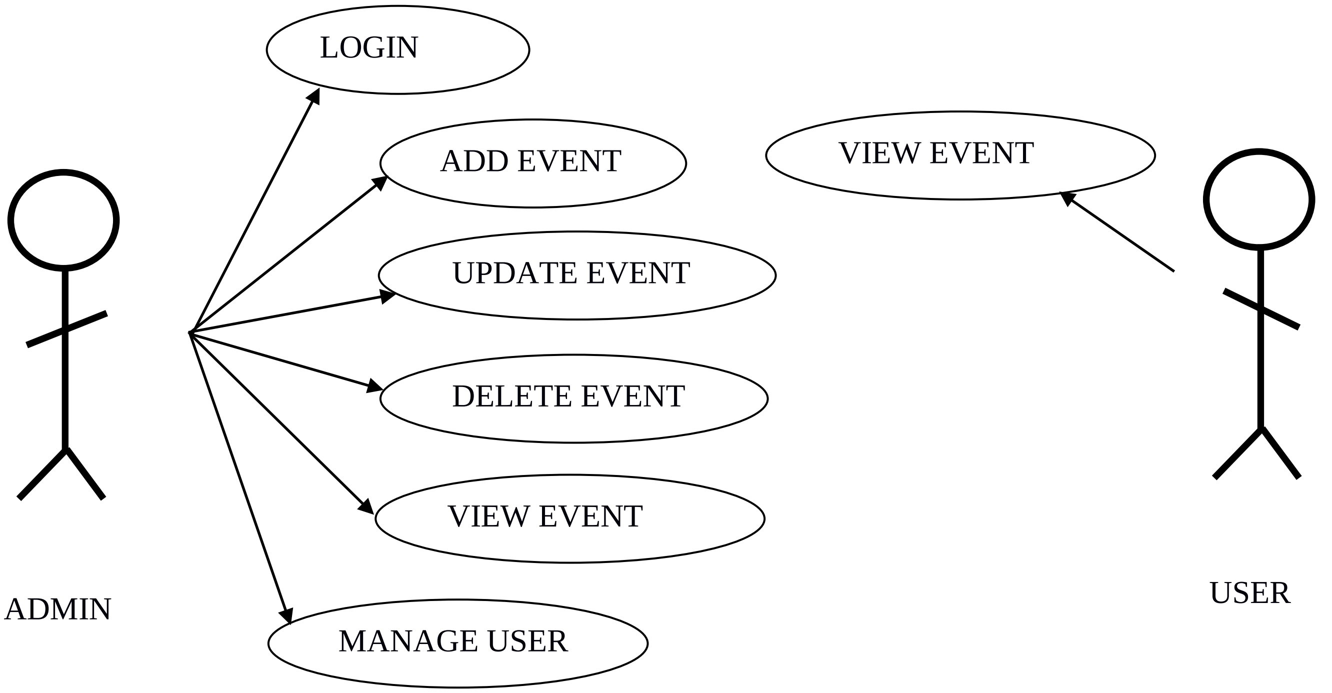


Fig 2.3 Use Case Diagram for Department Association

2.4 Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

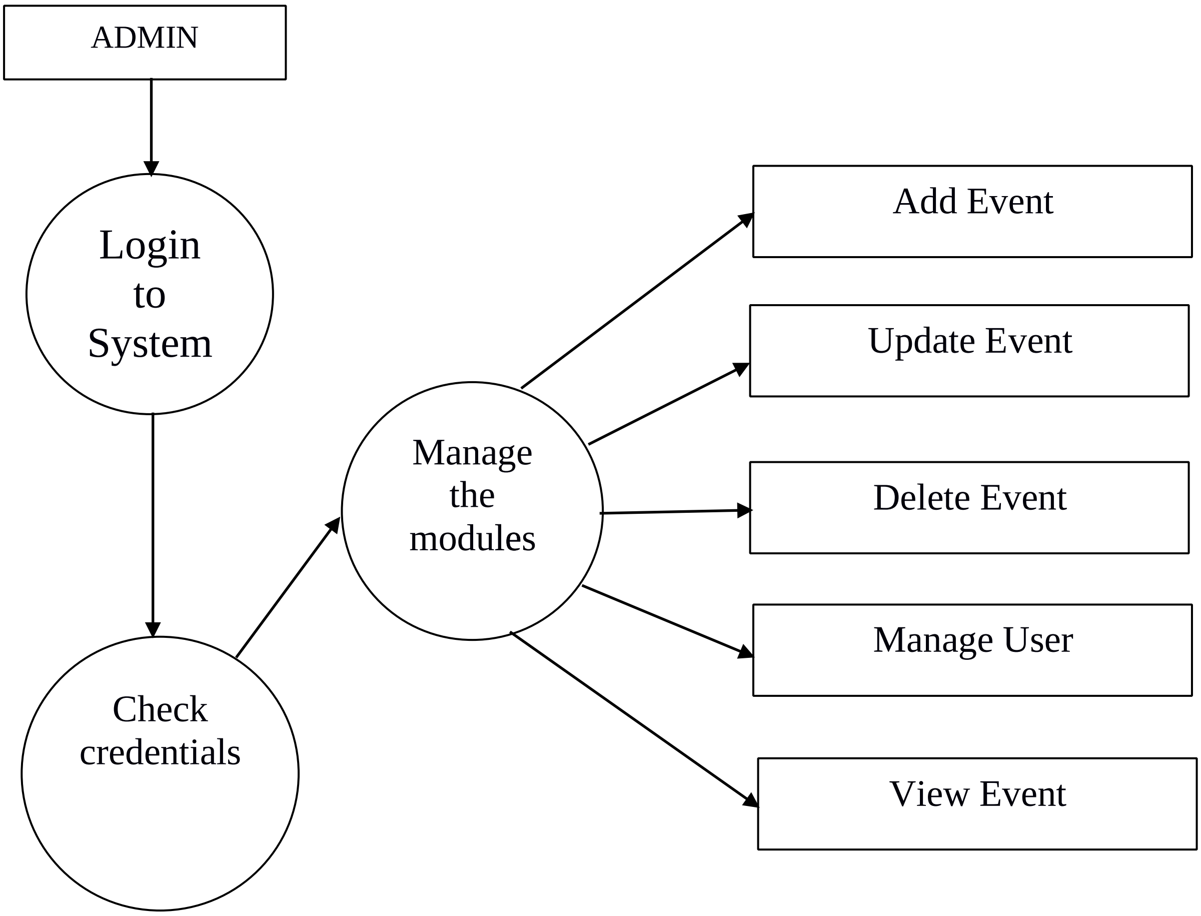


Fig 2.4 Data Flow Diagram for Department Association

2.5 Sequence Diagram

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

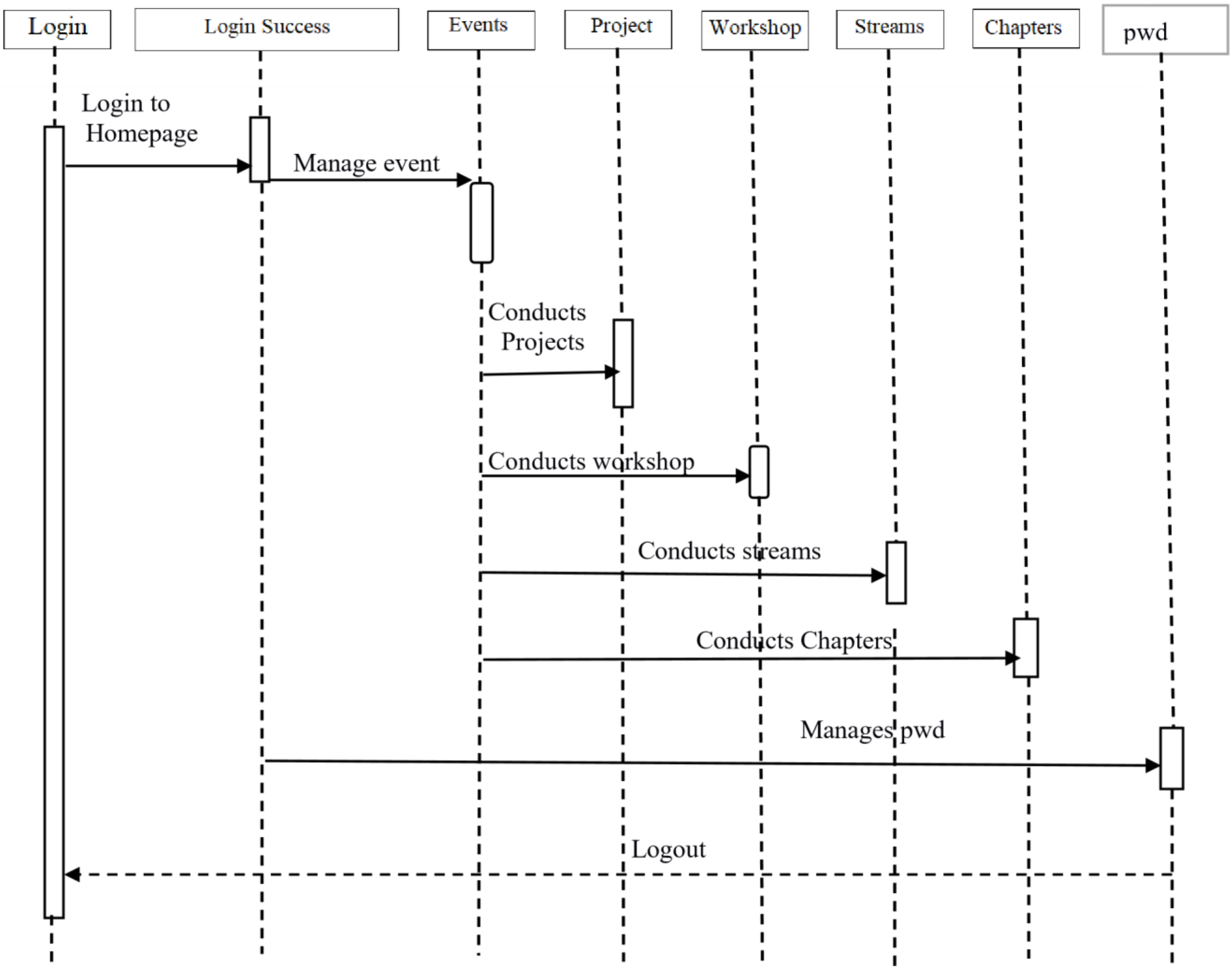


Fig 2.5 Sequence Diagram for Department Association

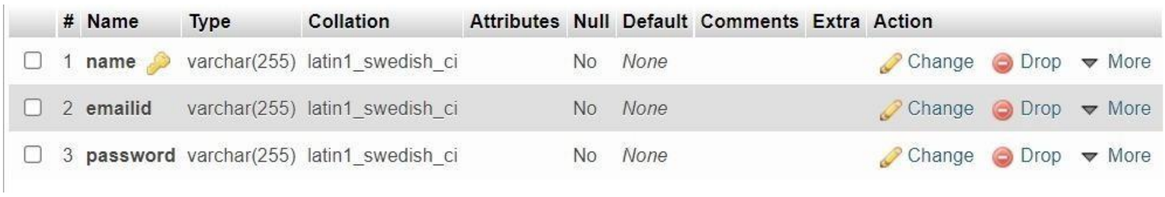
First the admin has to login by providing the password, after the login success the admin enters the homepage.
After entering the homepage, the admin enters into events page and manages the events.
Through events page, the admin manages projects, workshops, streams, chapters.
After performing these functions, the admin will logout.

Chapter 03

IMPLEMENTATION

3.1 Description of Tables

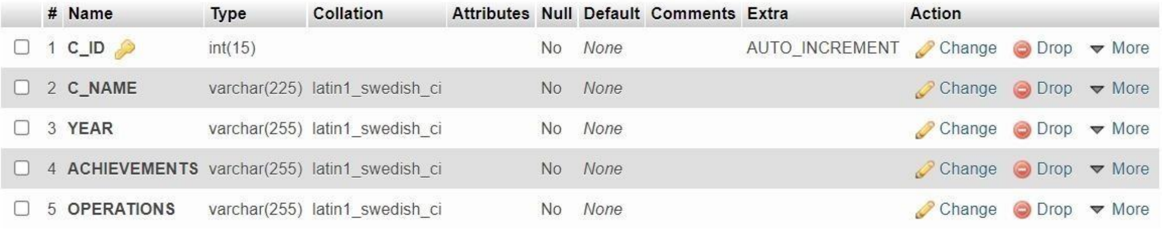
3.1.1 Admintable



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	name	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	2	emailid	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	3	password	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More

Fig 3.1 Admintable

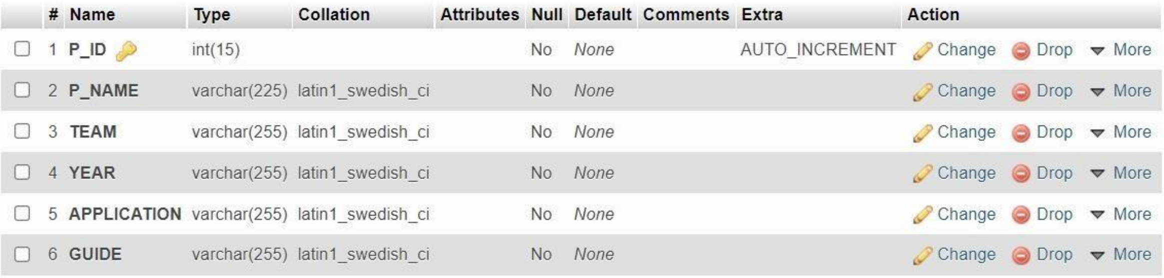
3.1.2 Chapters



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	C_ID	int(15)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	C_NAME	varchar(225)	latin1_swedish_ci		No	*None*			Change Drop More
□	3	YEAR	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	4	ACHIEVEMENTS	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	5	OPERATIONS	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More

Fig 3.2 Chapters

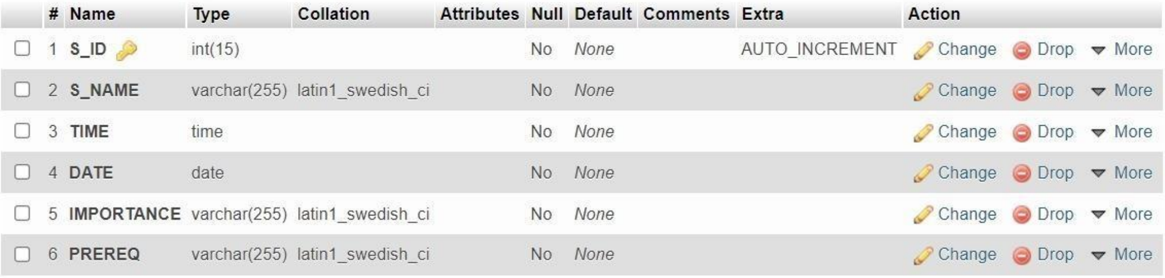
3.1.3 Projects



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	P_ID	int(15)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	P_NAME	varchar(225)	latin1_swedish_ci		No	*None*			Change Drop More
□	3	TEAM	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	4	YEAR	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	5	APPLICATION	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	6	GUIDE	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More

Fig 3.3 Projects

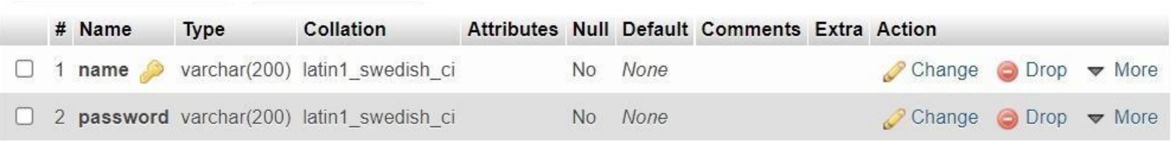
3.1.4 Streams



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	S_ID	int(15)			No	*None*		AUTO_INCREMENT	Change Drop More
□	2	S_NAME	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	3	TIME	time			No	*None*			Change Drop More
□	4	DATE	date			No	*None*			Change Drop More
□	5	IMPORTANCE	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More
□	6	PREREQ	varchar(255)	latin1_swedish_ci		No	*None*			Change Drop More

Fig 3.4 Streams

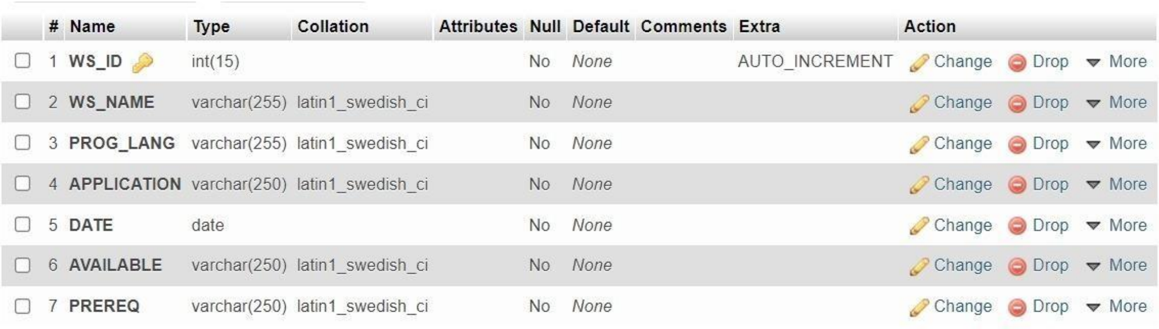
3.1.5 Usertable



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
□	1	**name**	varchar(200)	latin1_swedish_ci		No	*None*			Change	Drop	More
□	2	**password**	varchar(200)	latin1_swedish_ci		No	*None*			Change	Drop	More

Fig 3.5 Usertable

3.1.6 Workshops



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
□	1	**WS_ID**	int(15)			No	*None*		AUTO_INCREMENT	Change	Drop	More
□	2	**WS_NAME**	varchar(255)	latin1_swedish_ci		No	*None*			Change	Drop	More
□	3	**PROG_LANG**	varchar(255)	latin1_swedish_ci		No	*None*			Change	Drop	More
□	4	**APPLICATION**	varchar(250)	latin1_swedish_ci		No	*None*			Change	Drop	More
□	5	**DATE**	date			No	*None*			Change	Drop	More
□	6	**AVAILABLE**	varchar(250)	latin1_swedish_ci		No	*None*			Change	Drop	More
□	7	**PREREQ**	varchar(250)	latin1_swedish_ci		No	*None*			Change	Drop	More

Fig 3.6 Workshops

3.2 Constraints on Tables

i. In table admintable, name is primary key.

ii. In table Chapter, C_ID is a primary key and P_ID is foreign key.

iii. In table Project, P_ID is a primary key and E_ID is foreign key.

iv. In table Streams, S_ID is a primary key and WS_ID is foreign key.

v. In table Usertable, name is a primary key.

vi. In table Workshop, WS_ID is a primary key and E_ID is foreign key.

3.3 Back End Implementations

```
-- phpMyAdmin SQL Dump
-- version 5.1.1
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1:3306
-- Generation Time: Feb 07, 2022 at 11:58 AM
-- Server version: 5.7.36
-- PHP Version: 7.4.26

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time zone = "+00:00";


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
*/;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
*/;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `cse_department`
--

-- --------------------------------------------------------

--
-- Table structure for table `admintable`
```

```
--
DROP TABLE IF EXISTS `admintable`;
CREATE TABLE IF NOT EXISTS `admintable` (
  `name` varchar (255) NOT NULL,
  `emailid` varchar (255) NOT NULL,
  `password` varchar (255) NOT NULL,
  PRIMARY KEY (`name`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `admintable`
--

INSERT INTO `admintable` (`name`, `emailid`, `password`) VALUES
('aaa', 'aaa', 'aaa'),
('mudassir', '@gmail', '123');

-- --------------------------------------------------------


--
-- Table structure for table `chapter`
--
DROP TABLE IF EXISTS `chapter`;
CREATE TABLE IF NOT EXISTS `chapter` (
  `C_ID` int (15) NOT NULL AUTO_INCREMENT,
  `C_NAME` varchar (225) NOT NULL,
  `YEAR` varchar (255) NOT NULL,
  `ACHIEVEMENTS` varchar (255) NOT NULL,
  `OPERATIONS` varchar (255) NOT NULL,
  PRIMARY KEY (`C_ID`)
) ENGINE=MyISAM AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;

--
```

```
-- Dumping data for table `chapter`
--

INSERT INTO `chapter` (`C_ID`, `C_NAME`, `YEAR`, `ACHIEVEMENTS`,
`OPERATIONS`) VALUES
(1, 'IEEE', '2012', 'YES', 'YES');

-- --------------------------------------------------------


--
-- Table structure for table `project`
--
DROP TABLE IF EXISTS `project`;
CREATE TABLE IF NOT EXISTS `project` (
  `P_ID` int (15) NOT NULL AUTO_INCREMENT,
  `P_NAME` varchar (225) NOT NULL,
  `TEAM` varchar (255) NOT NULL,
  `YEAR` varchar (255) NOT NULL,
  `APPLICATION` varchar (255) NOT NULL,
  `GUIDE` varchar (255) NOT NULL,
  PRIMARY KEY (`P_ID`)
) ENGINE=MyISAM AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;

--
-- Dumping data for table `project`
--

INSERT INTO `project` (`P_ID`, `P_NAME`, `TEAM`, `YEAR`, `APPLICATION`,
`GUIDE`) VALUES
(1, 'DBMS', '1.4mh19cs125', '2012', 'MULTI PARADIGM', 'HEMANTH SIR'),
(3, 'dbms', '1.4mh19c', '2012', 'OHK', 'h');

-- --------------------------------------------------------
```

```
--
-- Table structure for table `stream`
--
DROP TABLE IF EXISTS `stream`;
CREATE TABLE IF NOT EXISTS `stream` (
  `S_ID` int (15) NOT NULL AUTO_INCREMENT,
  `S_NAME` varchar (255) NOT NULL,
  `TIME` time NOT NULL,
  `DATE` date NOT NULL,
  `IMPORTANCE` varchar (255) NOT NULL,
  `PREREQ` varchar (255) NOT NULL,
  PRIMARY KEY (`S_ID`)
) ENGINE=MyISAM AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;

--
-- Dumping data for table `stream`
--

INSERT INTO `stream` (`S_ID`, `S_NAME`, `TIME`, `DATE`, `IMPORTANCE`,
`PREREQ`) VALUES
(1, 'PYTHON', '14:15:00', '2022-01-15', 'FRONT END', 'ppp');

-- --------------------------------------------------------

--
-- Table structure for table `usertable`
--
DROP TABLE IF EXISTS `usertable`;
CREATE TABLE IF NOT EXISTS `usertable` (
  `name` varchar (200) NOT NULL,
  `password` varchar (200) NOT NULL,
  PRIMARY KEY (`name`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dumping data for table `usertable`
--

INSERT INTO `usertable` (`name`, `password`) VALUES
('mitm', 'mitm2023'),
('mudassir', '786'),
('shafiya', '123');

-- --------------------------------------------------------

--
-- Table structure for table `workshop`
--
DROP TABLE IF EXISTS `workshop`;
CREATE TABLE IF NOT EXISTS `workshop` (
  `WS_ID` int (15) NOT NULL AUTO_INCREMENT,
  `WS_NAME` varchar (255) NOT NULL,
  `PROG_LANG` varchar (255) NOT NULL,
  `APPLICATION` varchar (250) NOT NULL,
  `DATE` date NOT NULL,
  `AVAILABLE` varchar (250) NOT NULL,
  `PREREQ` varchar (250) NOT NULL,
  PRIMARY KEY (`WS_ID`)
) ENGINE=MyISAM AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;

--
-- Dumping data for table `workshop`
--

INSERT INTO `workshop` (`WS_ID`, `WS_NAME`, `PROG_LANG`, `APPLICATION`,
`DATE`, `AVAILABLE`, `PREREQ`) VALUES
(1, 'BASIC OF PYTHON', 'PYTHON ', 'MULTI PARADIGM', '2022-01-30', 'STUDENT', '--
NULL---'),
(3, 'DBMS', 'mysql', 'Data base', '2022-01-21', 'STUDENT', '---NULL---');
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

3.4 Front End Implementations

3.4.1 Connection

```
<?php
$dbhost='localhost';
$dbuser='root';
$dbpass='';
$dbname='cse_department';
//create connection
$conn=mysqli_connect ($dbhost, $dbuser, $dbpass, dbname);
?>
```

3.4.2 Add Chapter

```
<?php
                        include('connect-mysql.php');
                        if ($conn === false) {
   die ("ERROR: Could not connect. " . mysqli_connect_error ()); }

   $C_NAME=$_POST['C_NAME'];
   $YEAR=$_POST['YEAR'];
   $ACHIEVEMENTS=$_POST['ACHIEVEMENTS'];
   $OPERATIONS=$_POST['OPERATIONS'];


// Attempt insert query execution
$sql = "INSERT INTO chapter (C_NAME, YEAR, ACHIEVEMENTS, OPERATIONS)
VALUES
```

```
          ('$C_NAME','$YEAR','$ACHIEVEMENTS','$OPERATIONS')"; if
(mysqli_query ($conn, $sql)) {
   echo '<script>alert ("Data Inserted successfully”) </script>';
         header ('refresh:0; url=admin.php');
} else {
   echo '<script>alert ("Data Not Inserted”) </script> ‘;
}


mysqli_close($conn);
?>
```

3.4.3 Add Project

```
<?php
                              include('connect-
mysql.php');                        if ($conn === false) {
   die ("ERROR: Could not connect. " . mysqli_connect_error ()); }

   $P_NAME=$_POST['P_NAME'];
   $TEAM=$_POST['TEAM'];
   $YEAR=$_POST['YEAR'];
   $APPLICATION=$_POST['APPLICATION'];
         $GUIDE=$_POST['GUIDE'];



// Attempt insert query execution
$sql = "INSERT INTO project (P_NAME, TEAM, YEAR, APPLICATION, GUIDE)
VALUES
          ('$P_NAME','$TEAM','$YEAR','$APPLICATION','$GUIDE')"; if
(mysqli_query ($conn, $sql)) {
   echo '<script>alert ("Data Inserted successfully”) </script>';
         header ('refresh:0; url=admin.php');
} else {
   echo '<script>alert ("Data Not Inserted”) </script> ‘;}


mysqli_close($conn);
?>
```

3.4.4 Add Stream

```
<?php
                         include('connect-mysql.php');
```

```
                              if ($conn === false) {
    die ("ERROR: Could not connect. " . mysqli_connect_ error ()); }

   $S_NAME=$_POST['S_NAME'];
    $TIME=$_POST['TIME'];
    $DATE=$_POST['DATE'];
    $IMPORTANCE=$_POST['IMPORTANCE'];
          $PREREQ=$_POST['PREREQ'];




// Attempt insert query execution
$sql = "INSERT INTO stream (S_NAME, TIME, DATE, IMPORTANCE, PREREQ)
VALUES
           ('$S_NAME','$TIME','$DATE','$IMPORTANCE','$PREREQ')"; if
(mysqli_query ($conn, $sql)) {
   echo '<script>alert ("Data Inserted successfully") </script>';
header ('refresh:0; url=admin.php');
} else {
    echo '<script>alert ("Data Not Inserted") </script> ';
}


mysqli_close($conn);
?>
```

3.4.5 Add Workshop

```
<?php
                              include('connect-mysql.php');
                              if ($conn === false) {
    die ("ERROR: Could not connect. " . mysqli_connect_error ()); }

    $WS_NAME=$_POST['WS_NAME'];
    $PROG_LANG=$_POST['PROG_LANG'];
          $APPLICATION=$_POST['APPLICATION'];
          $DATE=$_POST['DATE'];
    $AVAILABLE=$_POST['AVAILABLE'];
          $PREREQ=$_POST['PREREQ'];



// Attempt insert query execution
```

```
$sql = "INSERT INTO workshop (WS_NAME, PROG_LANG, APPLICATION, DATE,
AVAILABLE, PREREQ) VALUES
('$WS_NAME','$PROG_LANG','$APPLICATION','$DATE','$AVAILABLE','$PREREQ')";
if (mysqli_query ($conn, $sql)) {
   echo '<script>alert ("Data Inserted successfully”) </script>';
        header ('refresh:0; url=admin.php');
} else {
   echo '<script>alert ("Data Not Inserted”) </script> ‘;
}

mysqli_close($conn);
?>
```

3.4.6 Delete Chapter

```
<?php

include('connect-mysql.php'); if(isset($_GET['deleteid']))
{
        $C_ID=$_GET['deleteid'];
        $query ="DELETE FROM chapter WHERE C_ID='$C_ID'";
        $result = mysqli_query ($conn, $query);

        if($result)
                {
                        echo '<script type="text/javascript"> alert ("Deleted Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Deleted")
</script>';
                }

}

?>
```

3.4.7 Delete Project

```
<?php

include('connect-mysql.php');
if(isset($_GET['deleteid'])) {
        $P_ID=$_GET['deleteid'];
        $query ="DELETE FROM project WHERE P_ID='$P_ID'";
        $result = mysqli_query ($conn, $query);
        if($result)
                {
                        echo '<script type="text/javascript"> alert ("Deleted Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Deleted")
</script>';
                }
}
?>
```

3.4.8 Delete Stream

```
<?php

include('connect-mysql.php'); if(isset($_GET['deleteid']))
{
        $S_ID=$_GET['deleteid'];
        $query ="DELETE FROM stream WHERE S_ID='$S_ID'";
        $result = mysqli_query ($conn, $query);

        if($result)
                {
                        echo '<script type="text/javascript"> alert ("Deleted Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Deleted")
</script>';
                }

}

?>
```

3.4.9 Delete Workshop

```
<?php

include('connect-mysql.php');
if(isset($_GET['deleteid'])) {
        $WS_ID=$_GET['deleteid'];
```

```
    $query ="DELETE FROM workshop WHERE WS_ID='$WS_ID'";
    $result = mysqli_query ($conn, $query);

    if($result)
        {
            echo '<script type="text/javascript"> alert ("Deleted Successfully")
</script>';
          header ('refresh:0; url=admin.php');
        }
        else {
            echo '<script type="text/javascript"> alert ("Data Not Deleted")
</script>';
        }
}
?>
```

3.4.9 Update Chapter

```
<?php  include('connect-
mysql.php');
    $C_ID=$_GET['updateid'];
    $query ="select * from chapter where C_ID=$C_ID";
    $result=mysqli_query ($conn, $query);
    $row=mysqli_fetch_assoc($result);


    $C_NAME=$row['C_NAME'];
    $YEAR=$row['YEAR'];
    $ACHIEVEMENTS=$row['ACHIEVEMENTS'];
    $OPERATIONS=$row['OPERATIONS'];

    if(isset($_POST['update']))         {

        $C_NAME=$_POST['C_NAME'];
        $YEAR=$_POST['YEAR'];
        $ACHIEVEMENTS=$_POST['ACHIEVEMENTS'];
        $OPERATIONS=$_POST['OPERATIONS'];


        $sql="update chapter set C_ID='$C_ID’, C_NAME='$C_NAME',
        YEAR='$YEAR’, ACHIEVEMENTS='$ACHIEVEMENTS',
          OPERATIONS='$OPERATIONS’,
    where C_ID=$C_ID";
          $result=mysqli_query ($conn, $sql);
    if($result)
```

```
                {
                        echo '<script type="text/javascript"> alert ("Updated Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Updated")
</script>';
                }
        }


?>
<html>
<head>
<title>Update</title>
<style>
        body {
                background-color: C87E9A;
        }
         input {width: 40%;              height:5%;
                  border: 1px;            border-
radius:05px;             padding: 8px 15px 8px
15px;            margin: 10px 0px 15px 0px;
         box-shadow: 1px 1px 2px 1px grey;
        }
</style>
</head>
<body>
        <center>
        <h1>UPDATE CHAPTER TABLE</h1>
        <form method="POST">

                Chapter Name:<br><input type="text" name="C_NAME" value="<?php echo
$C_NAME;>"    /><br>
                  YEAR:<br><input type="text" name="YEAR" placeholder="YEAR"
value="<?php echo $YEAR;>" /><br>
                  ACHIEVEMENTS:<br><input type="text" name="ACHIEVEMENTS"
placeholder="ACHIEVEMENTS" value="<?php echo $ACHIEVEMENTS;>" /><br>
         OPERATIONS:<br><input type="text" name="OPERATIONS"
placeholder="OPERATIONS" value="<?php echo $OPERATIONS;>" /><br>

                <input type="submit" name="update" value="update"/>
        </form>
        </center>
</body>
</html>
```

3.4.10 Update Project

```
<?php  include('connect-
mysql.php');
        $P_ID=$_GET['updateid'];
        $query ="select * from project where P_ID=$P_ID";
        $result=mysqli_query ($conn, $query);
        $row=mysqli_fetch_assoc($result);



        $P_NAME=$row['P_NAME'];
        $TEAM=$row['TEAM'];
        $YEAR=$row['YEAR'];
        $APPLICATION=$row['APPLICATION'];
        $GUIDE=$row['GUIDE'];

        if(isset($_POST['update']))        {

                $P_NAME=$_POST['P_NAME'];
                $TEAM=$_POST['TEAM'];
                $YEAR=$_POST['YEAR'];
                $APPLICATION=$_POST['APPLICATION'];
                $GUIDE=$_POST['GUIDE'];



                  $sql="update project set P_ID='$P_ID’, P_NAME='$P_NAME’,
         TEAM='$TEAM’, YEAR='$YEAR',
                APPLICATION='$APPLICATION’, GUIDE='$GUIDE',
                where P_ID=$P_ID";
                  $result=mysqli_query ($conn, $sql);
         if($result)
                {
                        echo '<script type="text/javascript"> alert ("Updated Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Updated")
</script>';
                }
        }


?>
<html>
<head>
<title>Update</title>
<style>
        body {
                background-color: C87E9A;
        }
         input {width: 40%;             height:5%;
                  border: 1px;             border-
radius:05px;             padding: 8px 15px 8px
15px;             margin: 10px 0px 15px 0px;
         box-shadow: 1px 1px 2px 1px grey;
        }
</style>
</head>
<body>
        <center>
        <h1>UPDATE PROJECT TABLE</h1>
        <form method="POST">

                Project Name:<br><input type="text" name="P_NAME" value="<?php echo
$P_NAME;?>" /><br>
                  TEAM:<br><input type="text" name="TEAM" placeholder="TEAM"
value="<?php echo $TEAM;>" /><br>
                  YEAR:<br><input type="text" name="YEAR" placeholder="YEAR"
value="<?php echo $YEAR;>" /><br>
                  APPLICATION<br><input type="text" name="APPLICATION"
placeholder="APPLICATION" value="<?php echo $APPLICATION;>" /><br>
         GUIDE:<br><input type="text" name="GUIDE" placeholder="GUIDE"
value="<?php echo $GUIDE;?>" /><br>


                <input type="submit" name="update" value="update"/>
        </form>
        </center>
</body>
</html>
```

3.4.11 Update Streams

```
<?php  include('connect-
mysql.php');
        $S_ID=$_GET['updateid'];
        $query ="select * from stream where S_ID=$S_ID";
        $result=mysqli_query ($conn, $query);
        $row=mysqli_fetch_assoc($result);


        $S_NAME=$row['S_NAME'];
        $TIME=$row['TIME'];
        $DATE=$row['DATE'];
        $IMPORTANCE=$row['IMPORTANCE'];
        $PREREQ=$row['PREREQ'];

        if(isset($_POST['update']))        {

                $S_NAME=$_POST['S_NAME'];
                $TIME=$_POST['TIME'];
                $DATE=$_POST['DATE'];
                $IMPORTANCE=$_POST['IMPORTANCE'];
                $PREREQ=$_POST['PREREQ'];

                $sql="update stream set S_ID='$S_ID’, S_NAME='$S_NAME',
                TIME='$TIME’, DATE='$DATE',
                IMPORTANCE='$IMPORTANCE',
                PREREQ='$PREREQ' where S_ID=$S_ID";
                  $result=mysqli_query ($conn, $sql);
         if($result)
                {
                        echo '<script type="text/javascript"> alert ("Updated Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Updated")
</script>';
                }
        }

?>
<html>
<head>
```

```
<title>Update</title>
<style>
        body {
                background-color: C87E9A;
        }
         input {width: 40%;
         height:5%;                border: 1px;
                  border-radius:05px;
         padding: 8px 15px 8px 15px;
                  margin: 10px 0px 15px
0px;
                box-shadow: 1px 1px 2px 1px grey;
        }
</style>
</head>
<body>
        <center>
        <h1>UPDATE STREAM</h1>
        <form method="POST">
  <! --<input type="text" name="CN_ID" placeholder="Enter the Component ID" valuebr>-->
                  STREAM NAME:<br><input type="text" name="S_NAME" value="<?php
echo $S_NAME;?>"     /><br>
                  TIME:<br><input type="time" name="TIME" placeholder="Enter the time of
stream" value="<?php echo $TIME;?>" /><br>
                   DATE:<br><input type="date" name="DATE" placeholder="Enter the Date of
Stream" value="<?php echo $DATE;?>" /><br>
                IMPORTANCE:<br><input type="text" name="IMPORTANCE"
placeholder="IMPORTANCE" value="<?php echo $IMPORTANCE;?>" /><br>
         PREREQ:<br><input type="text" name="PREREQ"
placeholder="PREREQUISITES" value="<?php echo $PREREQ;?>"/><br>

                <input type="submit" name="update" value="update"/>
        </form>
        </center>
</body>
</html>
```

3.4.12 Update Workshop

```
<?php  include('connect-
mysql.php');
        $WS_ID=$_GET['updateid'];
        $query ="select * from workshop where WS_ID=$WS_ID";
        $result=mysqli_query ($conn, $query);
        $row=mysqli_fetch_assoc($result);


        $WS_NAME=$row['WS_NAME'];
        $PROG_LANG=$row['PROG_LANG'];
        $APPLICATION=$row['APPLICATION'];
        $DATE=$row['DATE'];
        $AVAILABLE=$row['AVAILABLE'];
        $PREREQ=$row['PREREQ'];

        if(isset($_POST['update']))         {

                $WS_NAME=$_POST['WS_NAME'];
                $PROG_LANG=$_POST['PROG_LANG'];
                $APPLICATION=$_POST['APPLICATION'];
                $DATE=$_POST['DATE'];
                $AVAILABLE=$_POST['AVAILABLE'];
                $PREREQ=$_POST['PREREQ'];

                $sql="update workshop set WS_ID='$WS_ID’, WS_NAME='$WS_NAME',
                PROG_LANG='$PROG_LANG’, APPLICATION='$APPLICATION',
                DATE='$DATE’, AVAILABLE='$AVAILABLE',
                PREREQ='$PREREQ' where WS_ID=$WS_ID";
                  $result=mysqli_query ($conn, $sql);
         if($result)
                {
                        echo '<script type="text/javascript"> alert ("Updated Successfully")
</script>';
                   header ('refresh:0; url=admin.php');
                }
                else {
                        echo '<script type="text/javascript"> alert ("Data Not Updated")
</script>';
                }
        }
 ?>
<html>
<head>
<title>Update</title>
<style>
        body {
                background-color: C87E9A;
        }
```

```
          input { width: 40%;
          height:5%;                border: 1px;
                    border-radius:05px;
          padding: 8px 15px 8px 15px;
                    margin: 10px 0px 15px
0px;
                  box-shadow: 1px 1px 2px 1px grey;
         }
</style>
</head>
<body>
         <center>
         <h1>UPDATE COMPUTING NODES TABLE</h1>
         <form method="POST">
  <! --<input type="text" name="CN_ID" placeholder="Enter the Component ID" valuebr>-->
                    Workshop Name:<br><input type="text" name="WS_NAME" value="<?php
echo $WS_NAME;>"     /><br>
                    Programming Language:<br><input type="text" name="PROG_LANG"
placeholder="PROG_LANG" value="<?php echo $PROG_LANG;?>" /><br>
          APPLICATION:<br><input type="text" name="APPLICATION"
placeholder="APPLICATION" value="<?php echo $APPLICATION;?>" /><br>
          DATE:<br><input type="date" name="DATE" placeholder="Enter the DATE"
value="<?php echo $DATE;?>" /><br>
                    AVAILABLE:<br><input type="text" name="AVAILABLE"
placeholder="AVAILABLE TO" value="<?php echo $AVAILABLE;?>" /><br>
          PREREQISITES:<br><input type="text" name="PREREQ" placeholder="Enter the
PREREQISITES" value="<?php echo $PREREQ;?>"/><br>

                  <input type="submit" name="update" value="update"/>
         </form>
         </center>
</body>
</html>
```

Chapter 04

SNAP SHOTS AND DISCUSSION

4.1 Snapshots

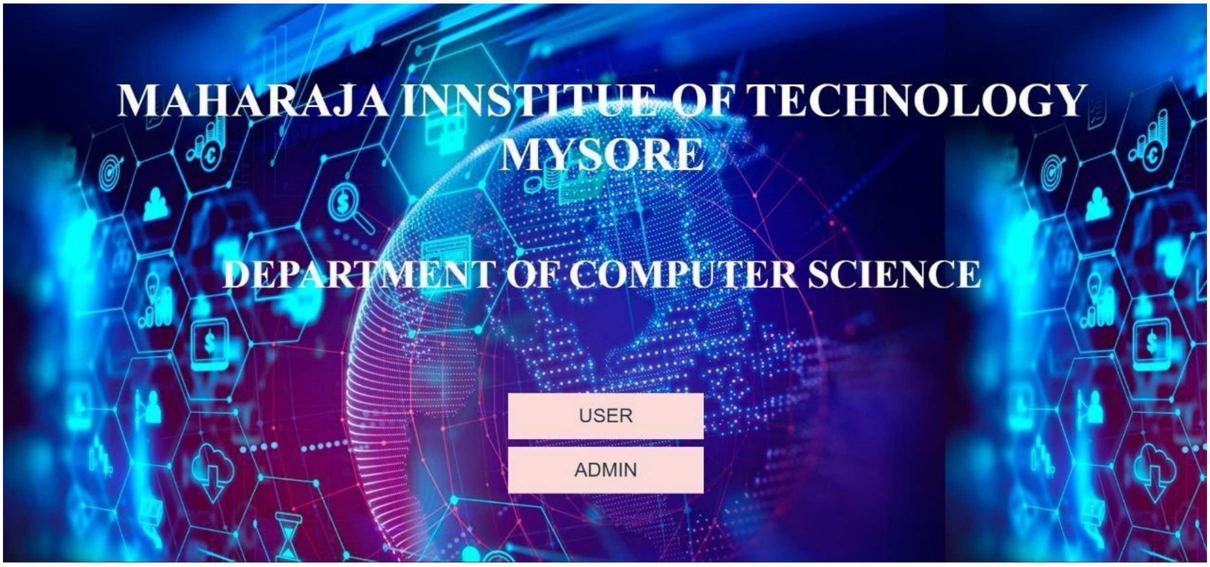

MAHARAJA INNSTITUE OF TECHNOLOGY
MYSORE
DEPARTMENT OF COMPUTER SCIENCE
USER
ADMIN

Fig 4.1 Department Association Management System Homepage

This fig 4.1 displays the optional entry for user and admin.

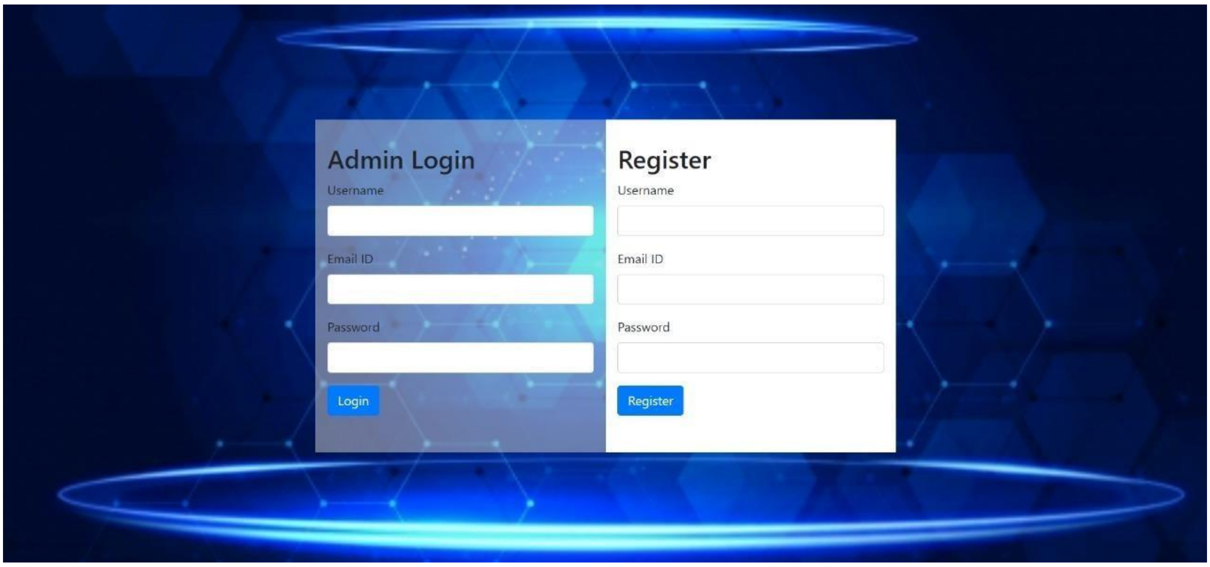


Fig 4.2 Admin Login and Registeration

Fig 4.2 displays the Login and Registeration of the Admin asking for Username, Email ID, Password. If he/she has is an Admin with no account they must register first. If account has already been created, he/she can directly login with the above-mentioned credentials.

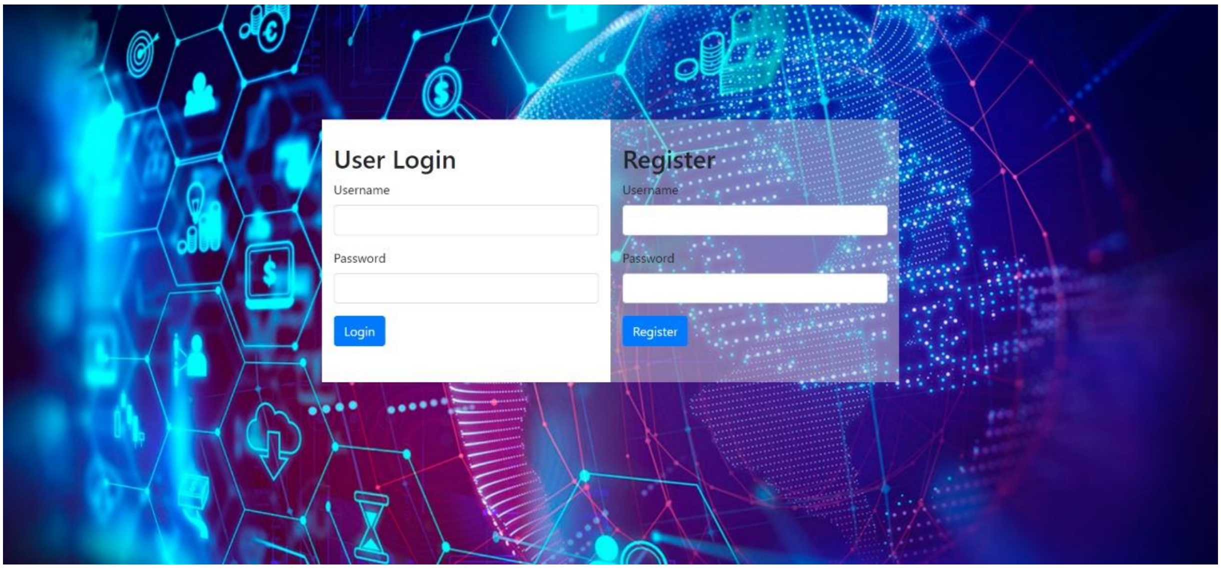


Fig 4.3 User Login and Registeration

Fig 4.2 displays the Login and Registeration of the User asking for Username, Password. If he/she has is an User with no account they must register first. If account has already been created, he/she can directly login with the above-mentioned credentials.

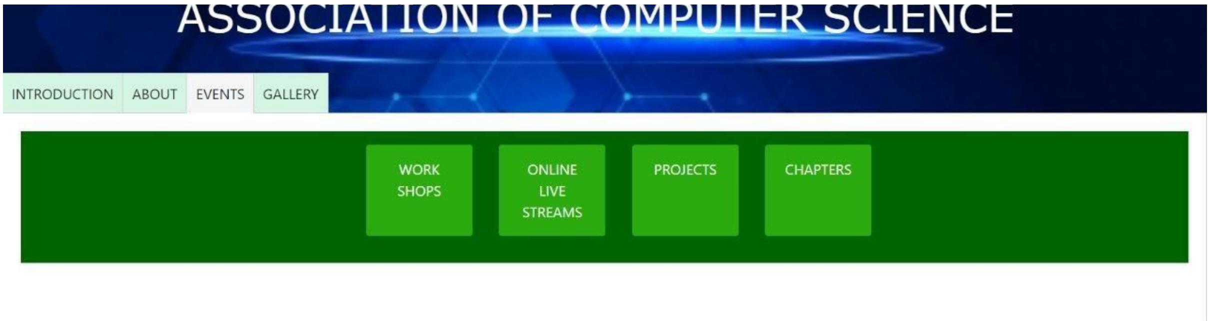
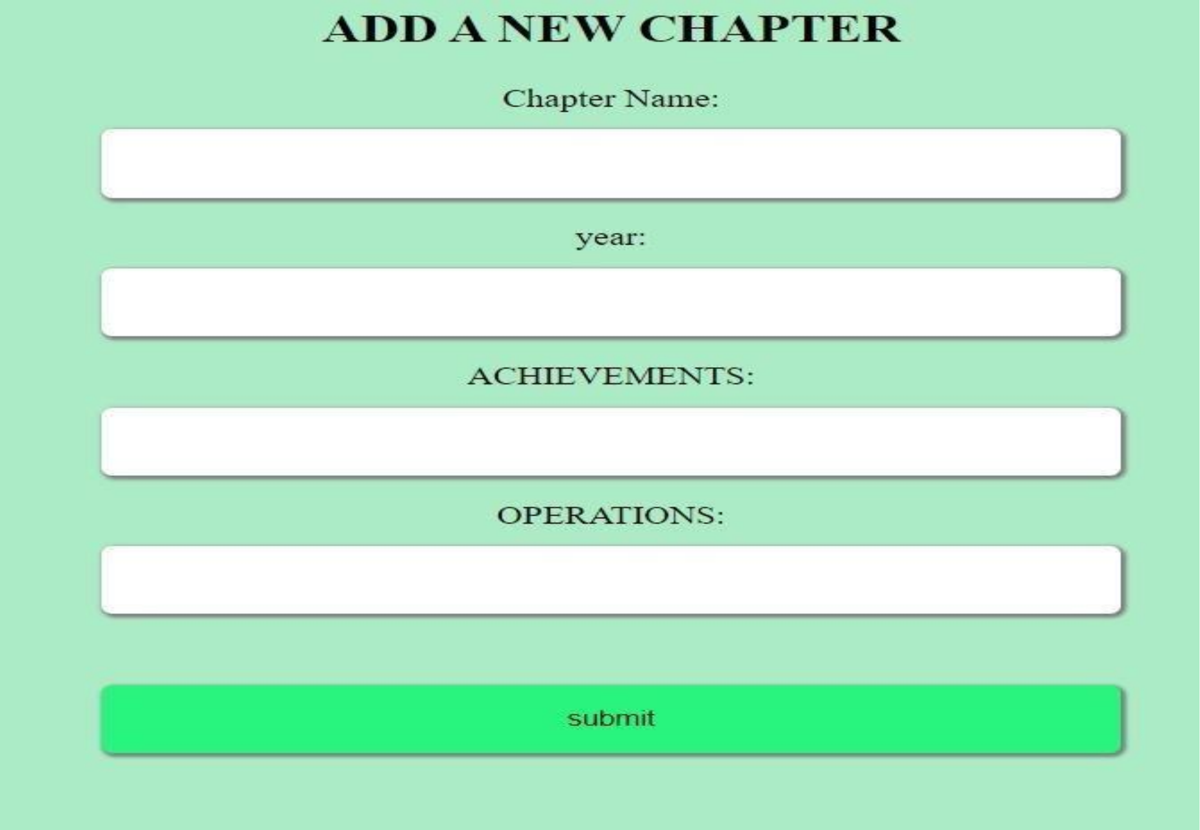


Fig 4.4 Event Page

Fig 4.4 displays dashboard containing Introduction, About, Events and Gallery.

Events are further distributed into Workshops, Online Live Steam, Projects and Chapters where the admin can add the new ones. If he/she is user then they can view the events conducted by admins in this page.



ADD A NEW CHAPTER

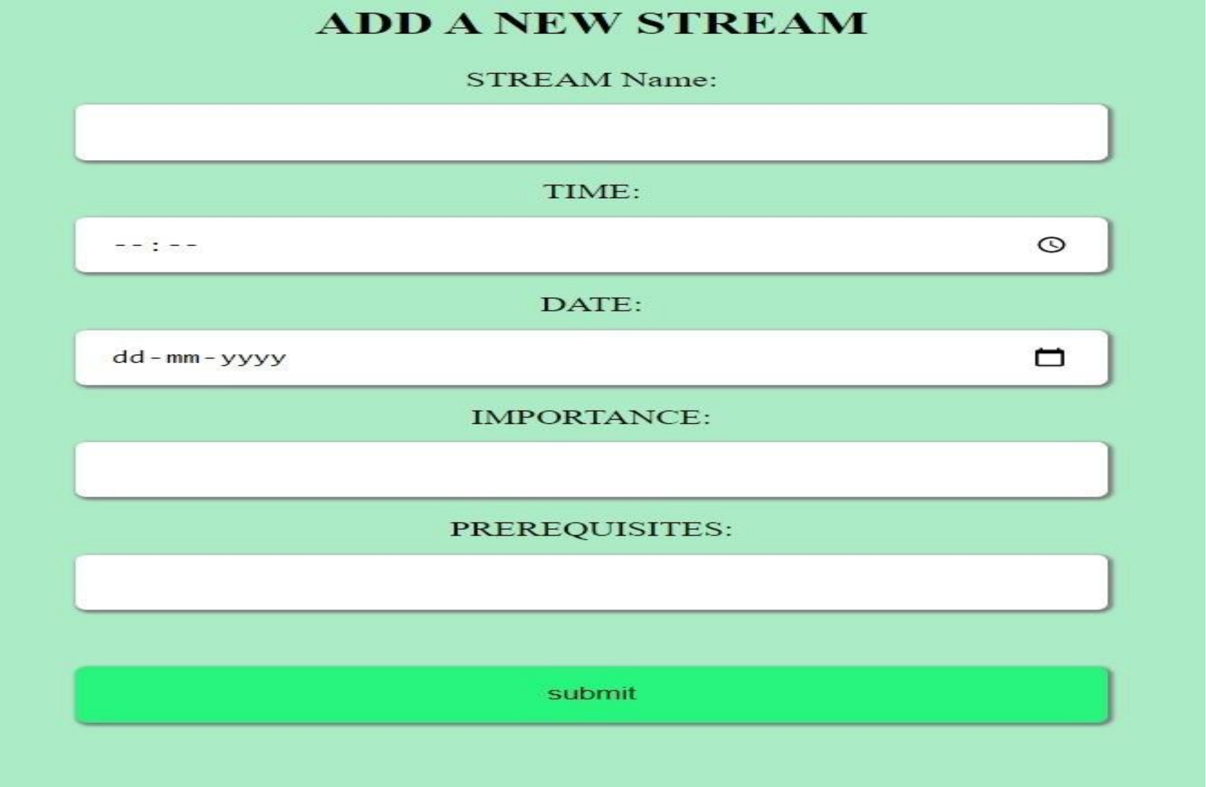
Chapter Name:

year:

ACHIEVEMENTS:

OPERATIONS:

Fig 4.5 Add New Chapter Page



ADD A NEW STREAM

STREAM Name:

TIME:

--:--

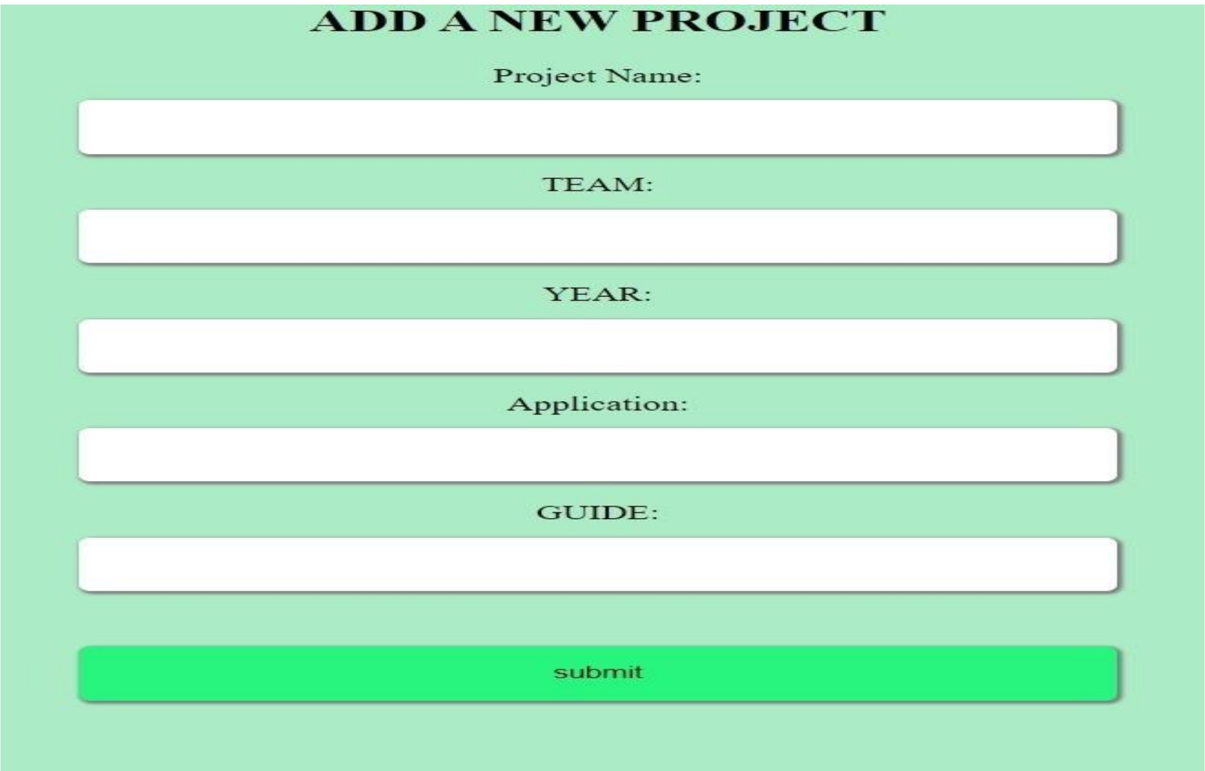
DATE:

dd-mm-yyyy

IMPORTANCE:

PREREQUISITES:

Fig 4.6 Add New Stream Page



ADD A NEW PROJECT

Project Name:

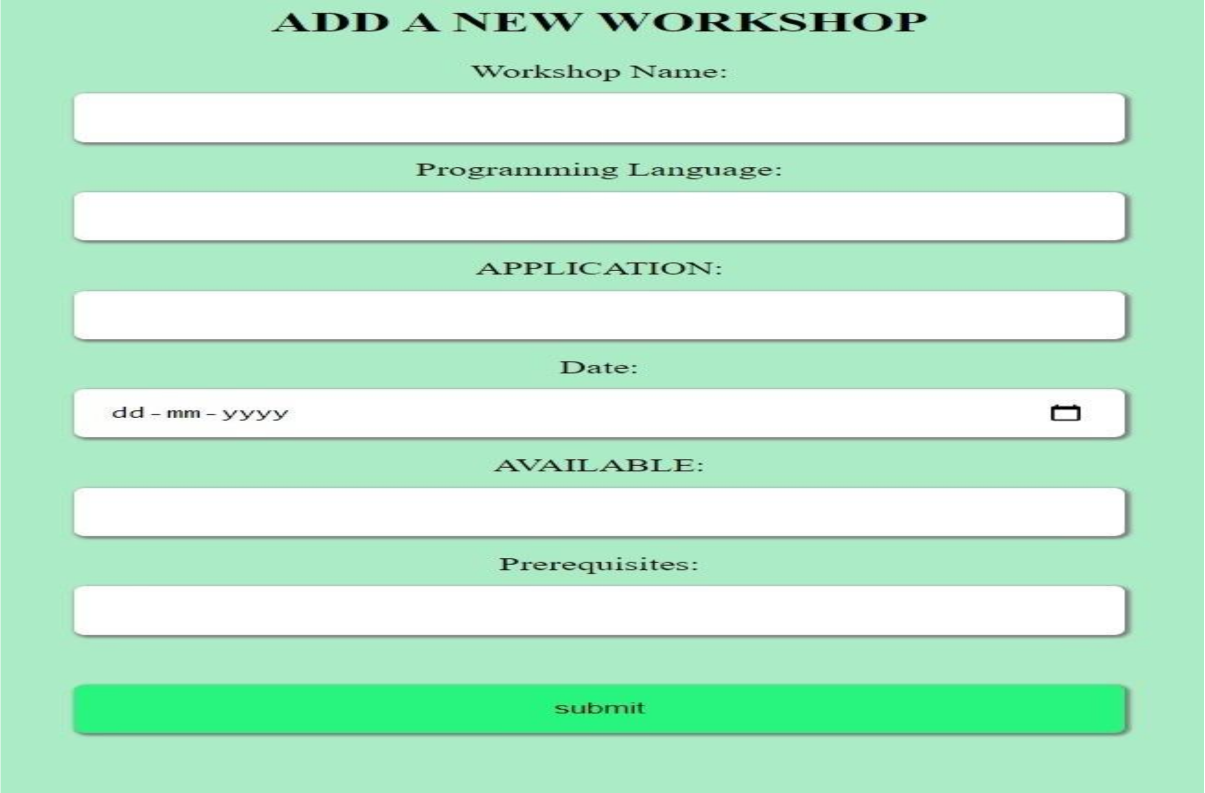
TEAM:

YEAR:

Application:

GUIDE:

Fig 4.7 Add New Project Page



ADD A NEW WORKSHOP

Workshop Name:

Programming Language:

APPLICATION:

Date:

dd - mm - yyyy

AVAILABLE:

Prerequisites:

Fig 4.8 Add New Workshop Page

4.2 Discussion

- Home Page

Here a visitor can register if they not yet registered. If they are registered then they can login and view notification.

- Admin Login Page

Here if the visitor is an admin, then by using his/her login credentials he/she can login to admin page.

- Event Page

User is a knowledge seeker, or more theoretical word, a learner. The User has to be registered and log in into the system before he/she can view the events conducted by admins.

- Admin Home Page

After admin login page, if user has valid credentials, then he/she can add/update/delete/view events i.e., project, workshop, chapter & streams.

4.3 Testing

Table 4.1 Test Cases

Test Case ID	Test Case	Expected Output	Actual Output	Status
TC1	Login with wrong email address and wrong password	Invalid email or password	As expected	Pass
TC2	Login with correct email address and wrong password	Invalid email or password	As expected	Pass
TC3	Login with wrong email address and correct password	Invalid email or password	As expected	Pass
TC4	Login with correct email address and correct password	Redirected to admin home page	As expected	Pass
TC5	Reminder send to wrong email address	Reminder found	As expected	Pass
TC6	Reminder send to correct email address	Reminder sent to the respective email address	As expected	Pass
TC7	Reminder viewed by correct user	Reminder has displayed correctly to that user	As expected	Pass
TC8	If entered Event name is already present in respective database.	Event not conducted	As expected	Fail
TC9	If entered Event name is not present in respective database.	New Event added	As expected	Pass

.

Chapter 05

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Department was facing difficulties while fetching list of events conducted after some months of conduction of the event. But using this system, department will be able to view the events even after some months/years. And also, entry the events to be conducted with respective date and time which also reminds the user.

5.2 Future Work

The project has a very vast scope in future by adding more functionality like validating forms, in events such as projects and workshops so that the user can provide their feedback and interests based on events conducted earlier which enhances the talents and skills of students of the respective department.

BIBLIOGRAPHY

1. HTML Tutorial - https://www.w3schools.com/html
2. CSS Tutorial - https://www.w3schools.com/Css
3. PHP Tutorial - https://www.w3schools.com/php
4. JS Tutorial - https://www.w3schools.com/Js
5. Book References - Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe.
6. Book References - Database management systems, Ramakrishnan, and Gehrke.