

Project Report
on
CAMPUS PLACEMENT WEBSITE

Submitted By

Soumya Jain - 2100290140135
Shagun Jindal - 2100290140120
Bunty Chauhan- 2100290140053
Munish Rai - 2100290140090

Master Of Computer Application

Under the Supervision of
Ms. DIVYA SINGHAL
Assistant Professor



Submitted To

Department of Computer Applications,
KIET Group of Institutions,
Delhi-NCR, Ghaziabad
Uttar Pradesh – 201206
(December 2022)

ACKNOWLEDGEMENTS

It gives us a great sense of pleasure to present the report of the Masters Of Computer Application Project undertaken during MCA . We owe special debt of gratitude to PROF. ARUN TRIPATHI, HEAD OF DEPARTMENT, KIET Group of Institute of Engineering and Technology, Ghaziabad for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavours have seen light of the day.

We also take the opportunity to acknowledge the contribution of ASSISTANT PROFESSOR DIVYA SINGHAL, DEPARTMENT OF MCA ,(KIET) for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Soumya Jain
Shagun Jindal
Bunty Chauhan
Munish Rai

ABSTRACT

From a student's perspective, placements can bring a wide range of benefits and opportunities. Training and management of placement is a crucial part of an educational institution in which most of the work is done manually. Manual system in the colleges requires a lot of manpower and time. With this project we aim to develop a web portal to solve this issue. The project is aimed at developing a website for the placement department of the college. The system which will be accessed and effectively used throughout the organization with proper login enabled. It can also be used by the Placement Officers in the college to manage the student information about placement thus reducing the manual work and consumes less paperwork. The system also provides the facility of viewing the personal and academic information of the student. The system gets the requested list of candidates for the companies who would like to recruit the people according to their eligibility criteria. The project is basically a website which can be easily accessed through mobile on the go.

TABLE OF CONTENTS

1 Introduction

1.1 Project Description	7
1.2 Project Scope	8
1.3 Hardware / Software used in project	
1.3.1 Hardware Requirements	9
1.3.2 Software Requirements	9
1.4 Module Description	
1.4.1 Admin	10
1.4.2 Student	10
1.4.3 Company	11

2 Feasibility Study

2.1 Technical feasibility	12
2.2 Economical feasibility	12
2.3 Physical feasibility	13

3 Database Design

3.1 Database Tables	
3.1.1 Student SignUp DB	15
3.1.2 Company Login DB	15
3.1.3 Student Registration DB	16
3.1.4 Admin Login DB	17
3.1.5 Company Detail DB	17

3.2 E-R Model	18
3.2.1 E-R Diagram	19
3.3 Use Case Diagram	
3.3.1 Student	20
3.3.2 Company	21
3.3.3 Admin	22
4 Form Design	
4.1 Home Page	23
4.2 Login Page	24
4.3 Company Dashboard	24
4.4 Feedback Page	25
4.5 Student Dashboard	25
4.6 Company Details	26
5 Coding	
• PHP Code(All Modules)	27
5.1 Admin Module	
• HTML Code	33
• CSS Code	34
5.2 Company Module	
• HTML Code	35
• CSS Code	36
5.3 Student Module	
• HTML Code	37
• CSS Code	38

6 System Testing

6.1 Unit Testing 42

6.2 Integration Testing 42

6.3 Performance Testing 42

7 Conclusion

8 Future Scope

9 Bibliography

CHAPTER - 1

INTRODUCTION

1.1 Project Description

The purpose of this Campus Recruitment Application is to simplify the existing handbook System by the help of Full-fledged computer Software, fulfilling their requirements.

The Organisation can maintain Computerized record without duplicacy in it. System provides the list of suitable companies to the students, according to their educational qualification, experience and their preferences. System provides the list of eligible students from a pool according to required skill for vacancy of company.

The three main users involved in this system are

- Student
- Employer
- Administrator

1.2 Project Scope

To eliminate the need of putting up notice or Emailing every student about the company coming in campus looking for potential employees/ Interns. The Students can keep updated themselves through this software.

To enable companies to view all students detail and system can shortlist students according to their criteria instead of doing manually or waiting till students or graduates physically go to organisations apply.

To enable students going for internship register online instead of Platform looking for internship or going to placement department for registration.

1.3 Hardware and Software Used

1.3.1 Hardware Requirements:

- Processor : Intel Pentium22020M@2.40GHZ
- Ram : 4.00GB
- HDD : 10GBFreespace or above
- System Type : 64/128-Bit Operating system

1.3.2 Software Requirements:

- Backend : PHP
- Database : MY SQL
- Front End : HTML, CSS, BOOTSTRAP
- Operating System : WINDOWS10/WINDOWS11

1.4 Modules Description

1.4.1 Admin

- **Dashboard:** In this section, admin can see all detail in brief like Total Company Registered, Total User (Candidates) Registered and Total Vacancy Listed.
- **Total Registered Company:** In this section, admin can view detail of registered company.
- **Total Registered Users:** In this section, admin can view detail of users.
- **Pages:** In this section, the admin can manage about us and contact us pages.
- **Reports:** In this section admin can view how many company has been registered in particular period and also view how many vacancy counts listed by particular company in particular periods.

1.4.2 Student

A student is registered by the site. A student can apply a job for the company and eligible students give an online exam which is held by the company . The student module deals with the information of the student.

A student who has added by the administrator to the system successfully can only able to access the system with their valid user name and password provided by the administrator. The first student should log in to the system by entering PRN as their user name and password. Students can able to update

his information such as name, branch, year, aggregate marks, contact number, email, etc. by clicking on Update Details option and also upload their Resume.

The change password field is used by the student if he needs to change his password as same in the admin module. After completing a task successfully by click on the Logout, the student can successfully log out from the system.

1.4.3 Company

- **Dashboard:** In this section, company can see all detail in brief like Total Number of application received, Total number of new applications, Total number selected application, Total number of rejected applications.
- **Post Vacancy:** In this section, company can manage job posting(Add/Manage).
- **Job Application:** In this section, company can view total new applications receive, total sorted applications and total rejected applications and company also have right to sort application and reject application and this selected and rejected message send to candidates.
- **Reports:** In this section, company can view job posting in a particular periods and also see how many application has been received in a particular periods.

Company can also update his/her company profile, change the password, see the notifications of new applications received and recover the password.

CHAPTER - 2

FEASIBILITY STUDY

Whenever we design a new system, normally the management will ask for a feasibility report of the new system.

The management wants to know the technicalities and cost involved in creation of new system.

- Technical feasibility
- Economic feasibility
- Physical feasibility

2.1 Technical Feasibility :

Technical feasibility involves study to establish the technical capability of the system being created to accomplish all requirements to the user. The system should be capable of handling the proposed volume of data and provide users and operating environment to increase their efficiency.

For example, system should be capable of handling the proposed volume of data and provide users.

2.2 Economic Feasibility:

Economic feasibility involves study to establish the cost benefit analysis. Money spent on the system must be recorded in the form of benefit from the system. The benefits are of two types:

2.2.1 Tangible benefits:

- Saving man labor to do tedious tasks saves time.

2.2.2 Intangible benefits:

- Improves the quality of organization.

2.3 Physical Feasibility:

It involves study to establish the time responses of the new system being created. For e.g., if the new system takes more than one day to prepare crucial finance statement for the management, wherever it was required in an hour, the system fails to provide the same.

It should be clearly establish that the new system requirements in the form of time responses would be completely met with. It may call for increase in cost. If the required cost is sacrificed then the purpose of the new system may not be achieved even if it was found to be technically feasible.

CHAPTER – 3

DATABASE DESIGN

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The MS Access database has been chosen for developing the relevant databases.

3.1 Database Tables

3.1.1 Student SignUp DB:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	UserName	varchar(25)	latin1_swedish_ci		No	None		Change Drop More
<input type="checkbox"/>	2	Email	varchar(56)	latin1_swedish_ci		No	None		Change Drop More
<input type="checkbox"/>	3	Password	varchar(10)	latin1_swedish_ci		No	None		Change Drop More

Fig:1

3.1.2 Company Login DB:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	UserName	varchar(10)	latin1_swedish_ci		No	None		Change Drop More
<input type="checkbox"/>	2	Password	varchar(24)	latin1_swedish_ci		No	None		Change Drop More

Fig:2

3.1.3 Student Registration DB:

































	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	FirstName	varchar(34)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	2	LastName	varchar(34)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	3	EmailId	varchar(45)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	4	MobileNo	int(12)			No	None			 Change  Drop  More
<input type="checkbox"/>	5	Gender	varchar(2)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	6	DOB	int(24)			No	None			 Change  Drop  More
<input type="checkbox"/>	7	Address	varchar(45)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	8	City	varchar(50)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	9	Pincode	int(7)			No	None			 Change  Drop  More
<input type="checkbox"/>	10	State	varchar(30)	latin1_swedish_ci		No	None			 Change  Drop  More
<input type="checkbox"/>	11	Country	varchar(32)	latin1_swedish_ci		No	None			 Change  Drop  More

Fig:3

3.1.4 Admin Login DB:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	UserName	varchar(10)	latin1_swedish_ci		No	None		 Change  Drop  More
<input type="checkbox"/>	2	Password	varchar(24)	latin1_swedish_ci		No	None		 Change  Drop  More

Fig:4

3.1.5 Company Detail DB:













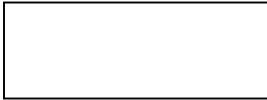
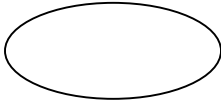
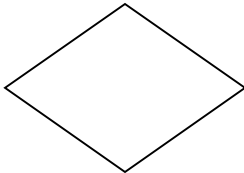

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	Details	varchar(500)	latin1_swedish_ci		No	None		 Change  Drop  More
<input type="checkbox"/>	2	Eligibility	varchar(150)	latin1_swedish_ci		No	None		 Change  Drop  More
<input type="checkbox"/>	3	Form	varchar(56)	latin1_swedish_ci		No	None		 Change  Drop  More
<input type="checkbox"/>	4	Date	datetime(5)			No	None		 Change  Drop  More

Fig:5

3.2 Entity-Relationship Model

E-R (Entity-Relationship) Diagram is used to represents the relationship between entities in the table.

The symbols used in E-R diagrams are:

SYMBOL	PURPOSE
	Represents Entity sets.
	Represent attributes.
	Represent Relationship Sets.
	Line represents flow

Structured analysis is a set of tools and techniques that the analyst.

To develop a new kind of a system:

The traditional approach focuses on the cost benefit and feasibility analysis, Project management, and hardware and software selection a personal considerations.

3.2.1 E-R Diagram of Project

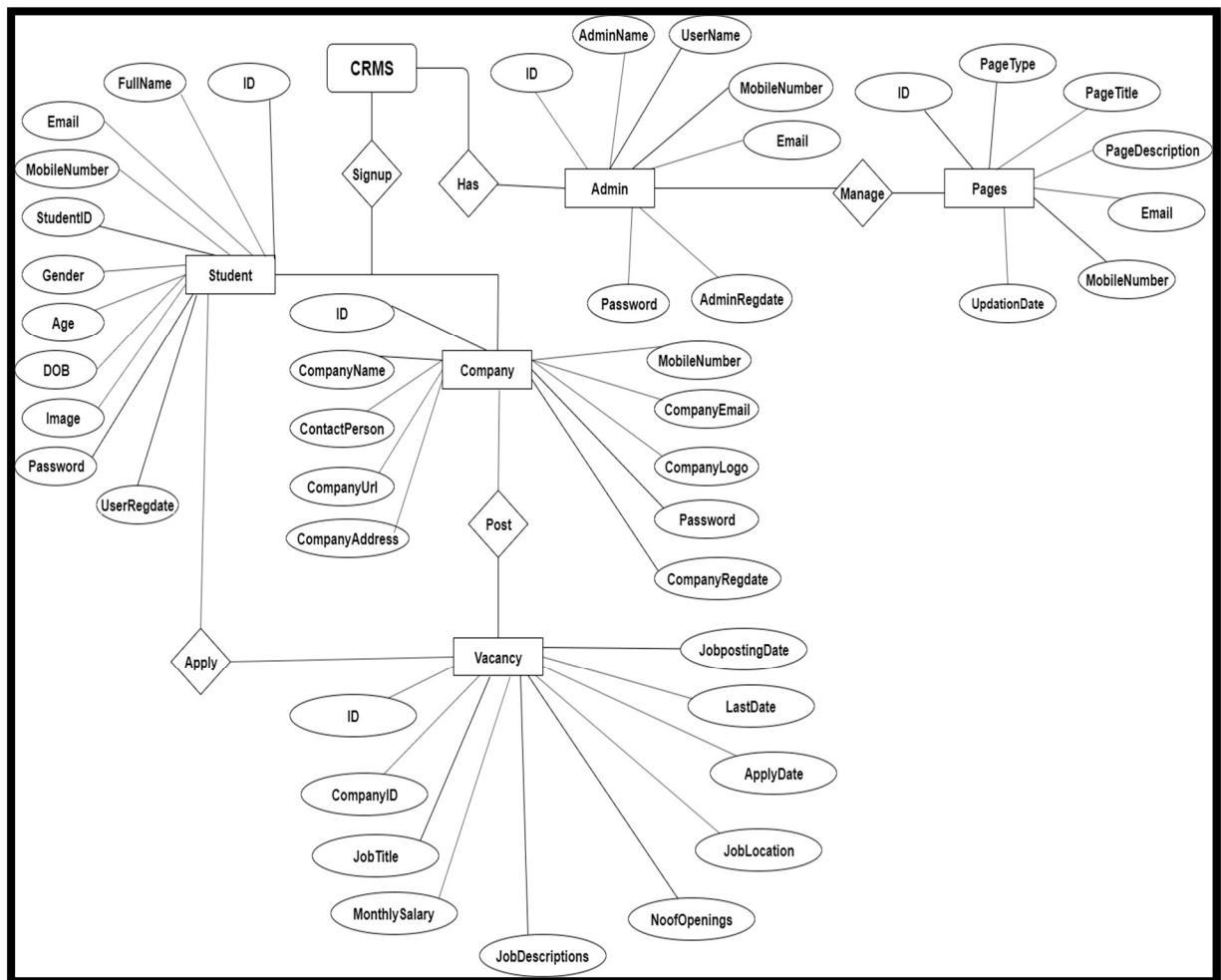


Fig:6

3.3 Use Case Diagrams

3.3.1 Student Module

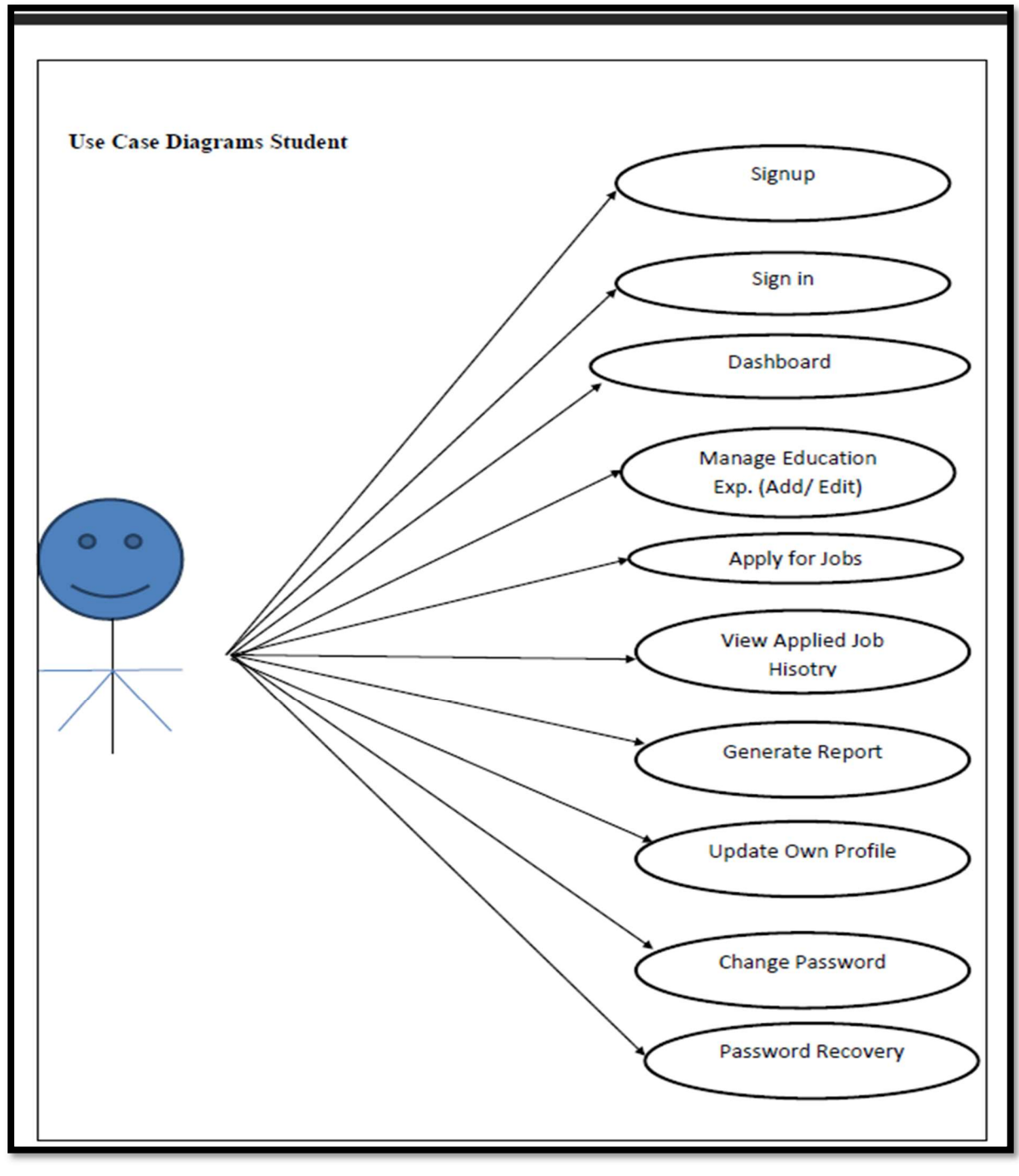


Fig:7

3.3.2 Company Module

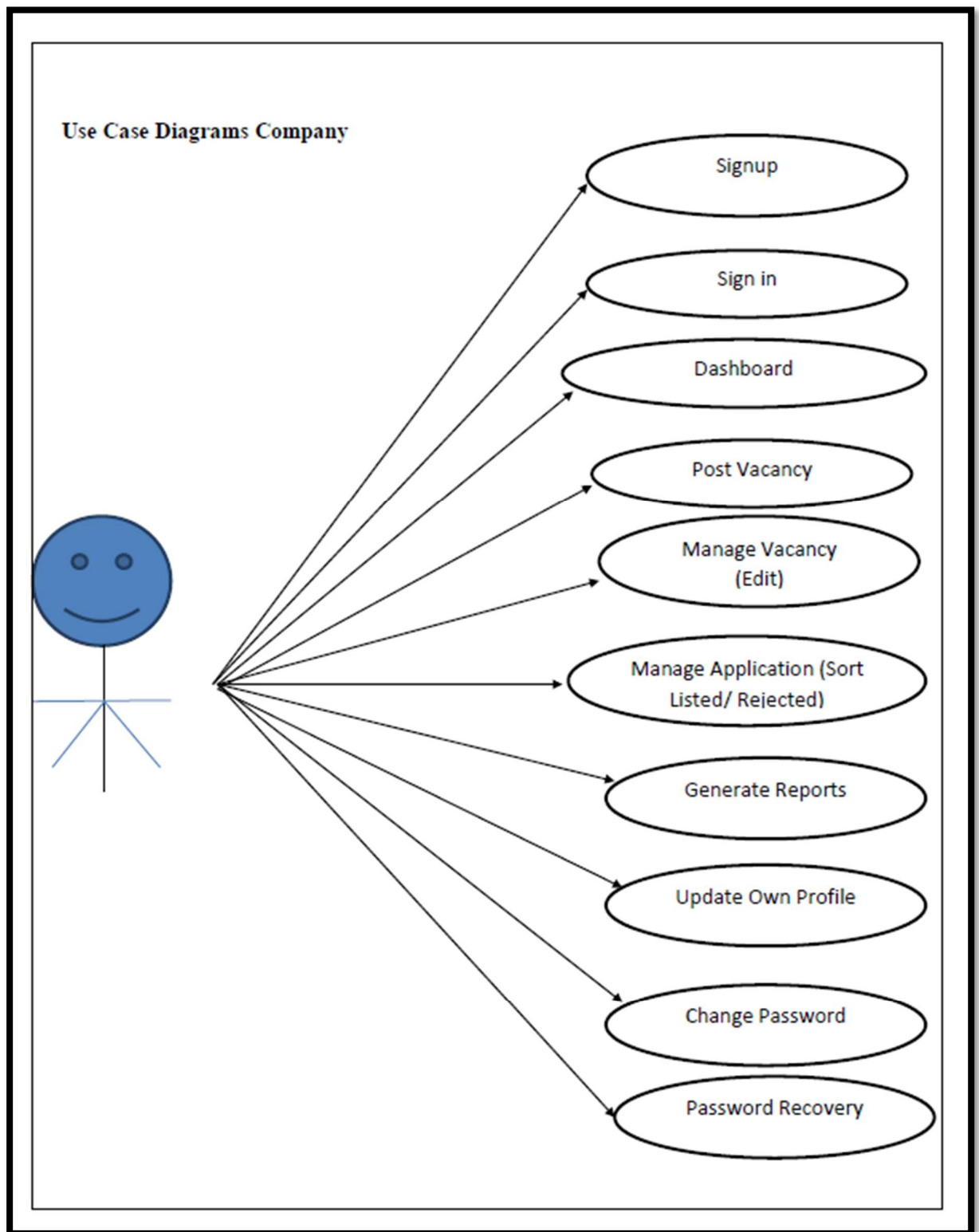


Fig:8

3.3.3 Admin Module

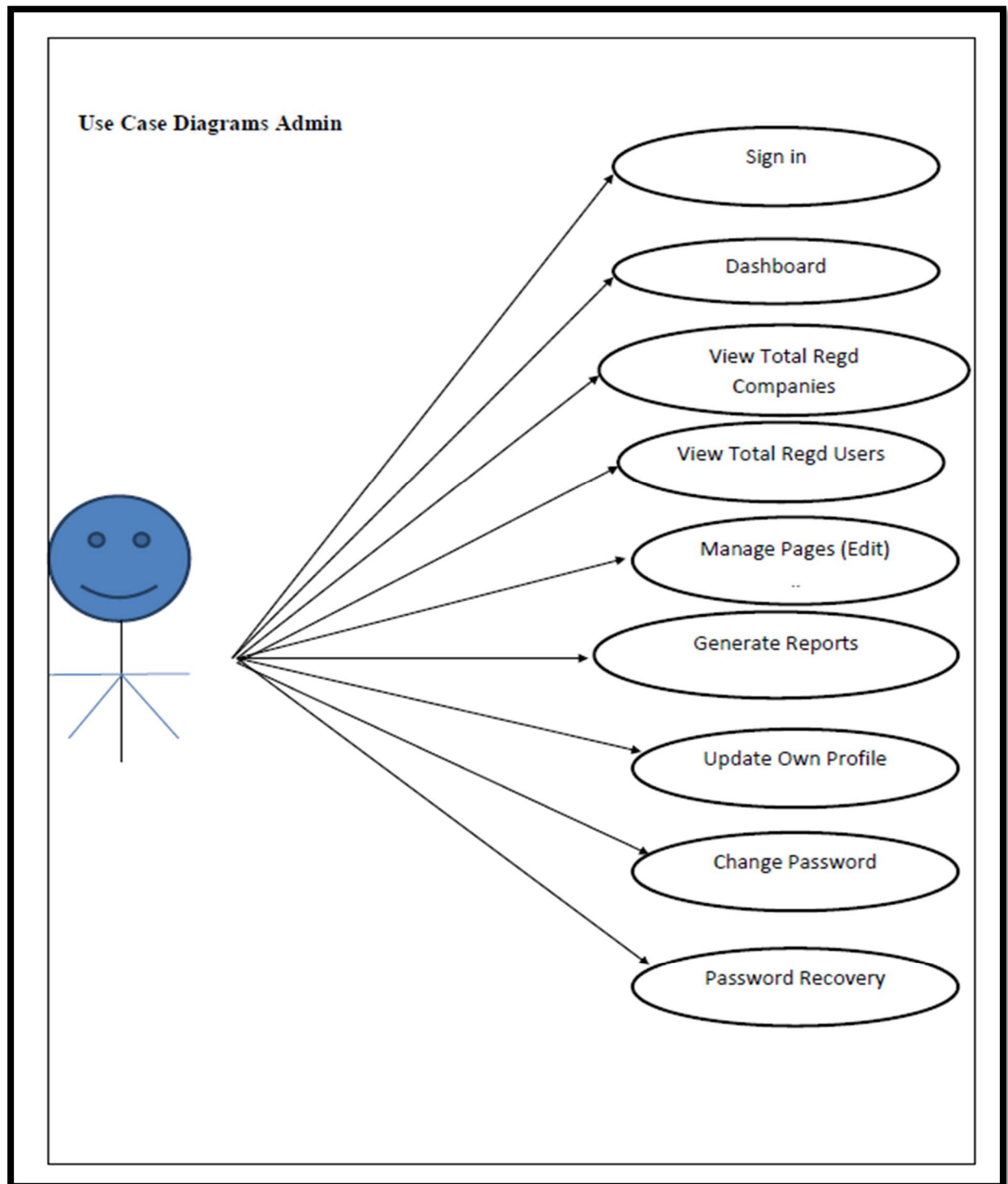


Fig:9

CHAPTER - 4

FORM DESIGN

4.1 Home Page



Fig.10

4.2 Login Page

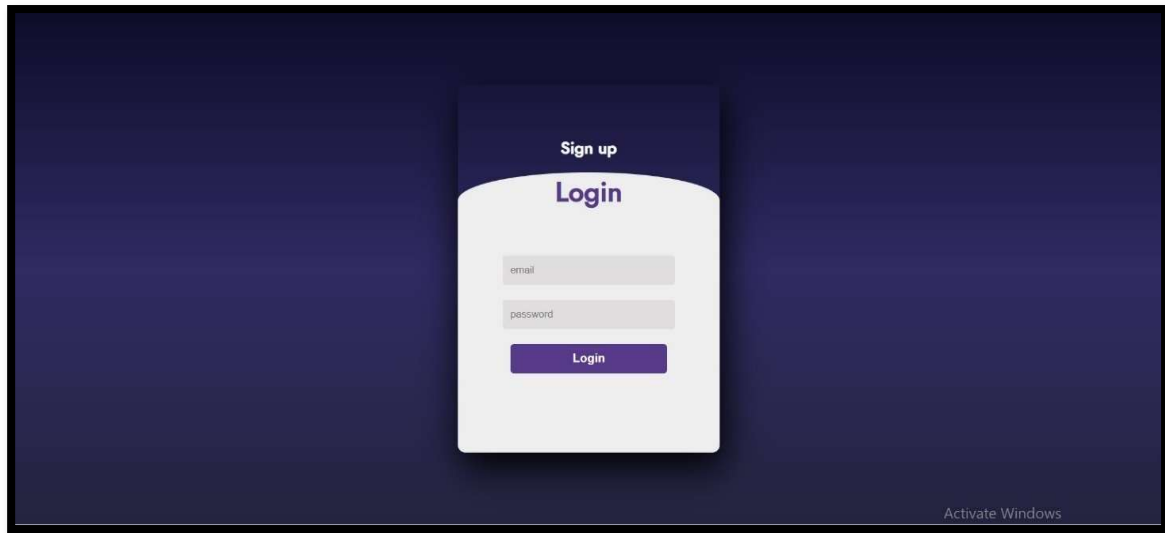


Fig.11

4.3 Company Dashboard

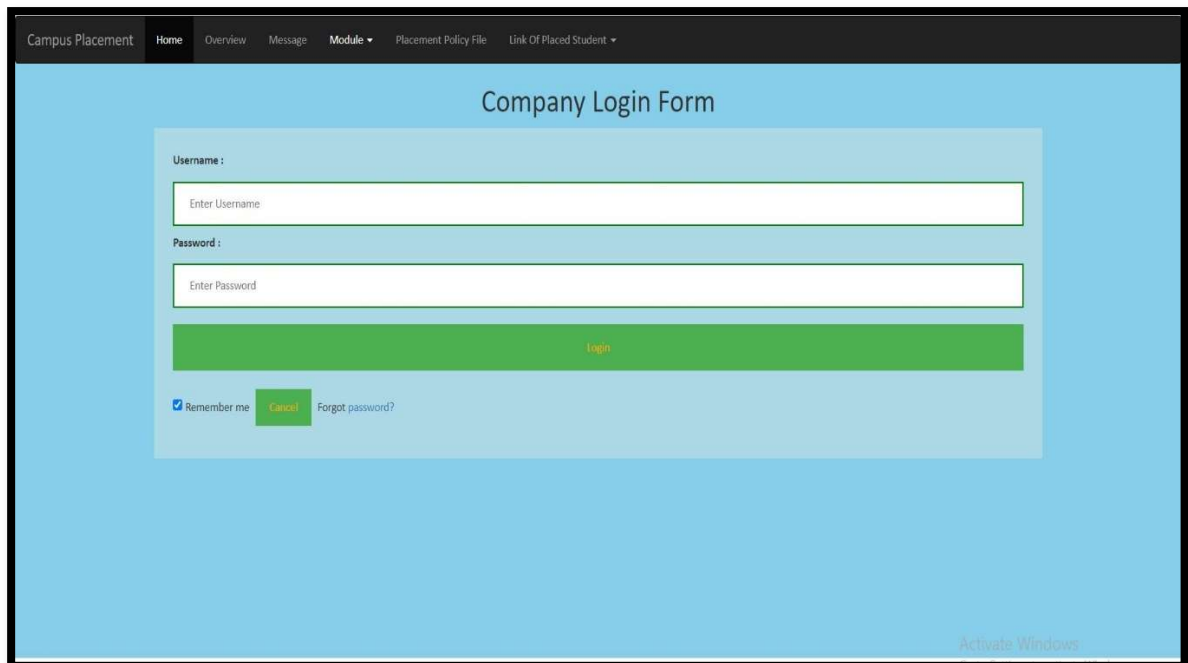
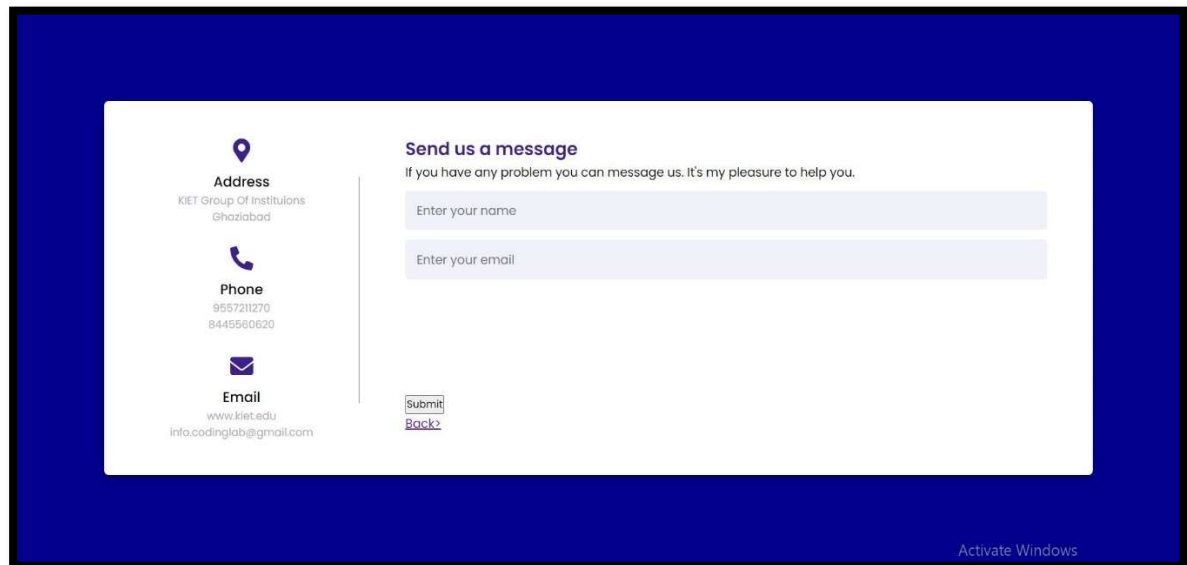


Fig.12

4.4 Feedback Page



The screenshot shows a feedback page with a dark blue background. On the left, there is contact information for KIET Group of Institutions, Ghaziabad, including a phone number (955721270, 8445560620) and an email address (www.kiet.edu, info.codinglab@gmail.com). On the right, there is a 'Send us a message' section with a text area for the name, a text area for the email, and a 'Submit' button. A 'Back' link is also present. An 'Activate Windows' watermark is visible in the bottom right corner.

Fig.13

4.5 Student Dashboard



The screenshot shows a student registration form titled 'Student Registration Form'. The form is set against a blue gradient background. It includes fields for First Name, Last Name, Email ID, Mobile Number, Gender (Male/Female), Date of Birth (DOB), Address, City, Pin Code, State, and Country. Each field has a corresponding label and a text input area. The form is framed by a blue border. An 'Activate Windows' watermark is visible in the bottom right corner.

Fig.14

4.6 Company Details

Hiring Company

Details	Eligibility	Url Form
Tcs	MCA	
wipro	MCA,Btech	https://forms.gle/kHPip4wcVXtczCE59
Accenture	MTECH	https://forms.gle/kHPip4wcVXtczCE59
Kiwitech	70	https://forms.gle/kHPip4wcVXtczCE59
appinventi	mttech	https://forms.gle/kHPip4wcVXtczCE59
capgemini	BSC, MCA	https://forms.gle/kHPip4wcVXtczCE59

logout Exit

Activate Windows

Fig.15

CHAPTER – 5

CODING

- **PHP Code(All Module)**

```
<?php
$servername = "localhost";
$username = "root";
$password = "";

// Create connection
$conn = new mysqli($servername, $username, $password, "test");

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>
<?php

include "Registration.php";

$sql = "SELECT * FROM detailsofcompany";

$result = $conn->query($sql);

?>

<!DOCTYPE html>

<html>

<head>

    <title>View Page</title>
```

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.cs
s">
```

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.cs
s">
```

```
<link rel="stylesheet" href="Style.css">
```

```
<link rel="stylesheet" href="style1.css">
<style>
  footer {
    text-align: center;
    padding: 3px;
    background-color: white;
    color: white;
  }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div>
  <a href="LandingPage.html" style="width:10%!important; height:10%;
background-color: #FFFFFF; justify-content:
end!important;"><button>Logout</button></a>
</div>
```

```
<div class="container">
<h2>Hiring Company</h2>
  <form>
<table class="table" border="10px">
```

```
<thead>
```

```
<tr>
```

```

        <th>Details</th>

        <th>Eligibility</th>
        <th>Url Form</th>

    </tr>

</thead>

<tbody>

    <?php
        if ($result->num_rows > 0) {
            while ($row = $result->fetch_assoc()) {
?>

                <tr>

<td><?php echo $row['Details']; ?></td>

<td><?php echo $row['Eligibility']; ?></td>

<td><?php echo $row['Form']; ?></td>

</tr>

                <?php    }

            }

        ?>
    </tbody>

</table>
</form>
</div>
<section>

```

```

        <footer>
        <a href="LandingPage.html">Exit</a>
    </footer>
</section>

</body>
</html>

<?php

@include "Registration.php";

if (isset($_POST['done'])) {
    $Email=$_POST['Email'];
    $Password=$_POST['Password'];
    $sql="select * from student where Email='".$_$Email.'"AND
Password='".$_$Password.'"
    limit 1";
    $query=mysqli_query($conn,$sql);
    if(mysqli_num_rows($query)>0){
        $error="user already exist";
        echo "login sucessfully";
        header("location: ./ShowDetails.php");
    }else{
        $sql="select * from student where Email='".$_$Email.'"limit 1";
        $query=mysqli_query($conn,$sql);
        if(mysqli_num_rows($query)>0){
            echo '<script>alert("Enter correct password")</script>';
        }
        else{
            echo '<script>alert("Enter correct username")</script>';
        }
    }
}
}
}
<?php

include "Registration.php";

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

```

```

$FirstName = $_POST['FirstName'];

$LastName = $_POST['LastName'];

$EmailId = $_POST['EmailId'];

$MobileNo = $_POST['MobileNo'];

$Gender = $_POST['Gender'];
$DOB = $_POST['DOB'];
$Address = $_POST['Address'];
$City = $_POST['City'];
$Pincode = $_POST['Pincode'];
$State = $_POST['State'];
$Country = $_POST['Country'];

$sql = "INSERT INTO `studentregistrationform` (FirstName, LastName,
EmailId, MobileNo, Gender,DOB,Address,City,Pincode,State,Country)
VALUES
('$FirstName','$LastName','$EmailId','$MobileNo','$Gender','$DOB','$Address','$City','$Pincode','$State','$Country')";

$result = $conn->query($sql);

if ($result == TRUE) {

    echo '<script>alert("Record Successfully Entered")</script>';

}else{

    echo "not inserted";

}

$conn->close();

}

?>

```

```

<?php
include "Registration.php";

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

    $UserName = $_POST['UserName'];

    $Email = $_POST['Email'];

    $Password = $_POST['Password'];

    $sql = "INSERT INTO `student`(`UserName`, `Email`, `Password`)
VALUES ('$UserName','$Email','$Password')";

    $result = $conn->query($sql);

    if ($result == TRUE) {

        echo '<script>alert("Record Successfully Entered")</script>';
        header("location: ./Students_Registration_Form.html");

    }else{

echo "Error:". $sql . "<br>". $conn->error;

    }

    $conn->close();

}

?>
<?php
include "Registration.php";
$sql = "SELECT * FROM detailsofcompany";

$result = $conn->query($sql);
?>

```


5.1 Admin Module

- **HTML Code**

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>

<link rel="stylesheet" href="Style.css">
</head>
<body>
<center> <h1> Admin Login Form </h1> </center>
<form>
<div class="container">
<label>Username : </label>
<input type="text" placeholder="Enter Username" name="username"
required>
<label>Password : </label>
<input type="password" placeholder="Enter Password" name="password"
required>
<button type="submit">Login</button>
<input type="checkbox" checked="checked"> Remember me
<button type="button" class="cancelbtn"
onclick="window.location.href='LandingPage.html'"> Cancel</button>
Forgot <a href="Forget.html"> password? </a>
</div>
</form>
</body>
</html>
```

- **CSS Code**

```
Body {  
    font-family: Calibri, Helvetica, sans-serif;  
    background-color: white;  
}  
Html,body{  
    height:50%;  
}  
.container{  
    display:table;  
    height:100px;  
    margin:0 auto;  
}  
button {  
    background-color: #4CAF50;  
    width: 100%;  
    color: orange;  
    padding: 15px;  
    margin: 10px 0px;  
    border: none;  
    cursor: pointer;  
}  
  
input[type=text], input[type=password] {  
    width: 100%;  
    margin: 8px 0;  
    padding: 12px 20px;  
    display: inline-block;  
    border: 2px solid green;  
    box-sizing: border-box;  
}  
button:hover {  
    opacity: 0.7;  
}  
.cancelbtn {  
    width: auto;  
    padding: 10px 18px;  
    margin: 10px 5px;
```

```

}
.container {
    padding: 25px;
    background-color: lightblue;
}

```

5.2 Company Module

- **HTML Code**

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>
<link rel="stylesheet" href="Style.css">
</head>
<body>
    <center> <h1> Company Login Form </h1> </center>
    <form>
        <div class="container">
            <label>Username : </label>
            <input type="text" placeholder="Enter Username" name="username"
required>
            <label>Password : </label>
                <input type="password" placeholder="Enter Password"
name="password" required>
            <button type="submit">Login</button>
            <input type="checkbox" checked="checked"> Remember me
                <button type="button" class="cancelbtn"
onclick="window.location.href='LandingPage.html'"> Cancel</button>
            Forgot <a href="Forget.html"> password? </a>
        </div>
    </form>
</body>
</html>

```

- **CSS Code**

```
Body {
    font-family: Calibri, Helvetica, sans-serif;
    background-color: white;
}
Html,body{
    height:50%;
}
.container{
    display:table;
    height:100px;
    margin:0 auto;
}
button {
    background-color: #4CAF50;
    width: 100%;
    color: orange;
    padding: 15px;
    margin: 10px 0px;
    border: none;
    cursor: pointer;
}
input[type=text], input[type=password] {
    width: 100%;
    margin: 8px 0;
    padding: 12px 20px;
    display: inline-block;
    border: 2px solid green;
    box-sizing: border-box;
}
button:hover {
    opacity: 0.7;
}
.cancelbtn {
    width: auto;
    padding: 10px 18px;
    margin: 10px 5px;
}
```

```
.container {
    padding: 25px;
    background-color: lightblue;
}
```

5.3 Student Module

- **HTML Code**

```
<!DOCTYPE html>
<html>
<head>
    <title>Slide Navbar</title>
    <link rel="stylesheet" type="text/css" href="som.css">
</head>
<body>
    <div class="main">
        <input type="checkbox" id="chk" aria-hidden="true">

        <div class="signup">
            <form>
                <label for="chk" aria-hidden="true">Sign up</label>
                <input type="text" name="txt" placeholder="User name"
required="">
                <input type="email" name="email" placeholder="Email"
required="">
                <input type="password" name="pswd" placeholder="Password"
required="">
                <button>Sign up</button>
            </form>
        </div>

        <div class="login">
            <form>
```

```

        <label for="chk" aria-hidden="true">Login</label>
        <input type="email" name="email" placeholder="Email"
required="">
        <input type="password" name="pswd" placeholder="Password"
required="">
        <button>Login</button>

        Forgot <a href="Forget.html"> password? </a>

    </form>
</div>
</div>
</body>
</html>

```

- **CSS Code**

```

body{
    margin: 0;
    padding: 0;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
    font-family: 'Jost', sans-serif;
    background: linear-gradient(to bottom, #0f0c29, #302b63, #24243e);
}
.main{
    width: 350px;
    height: 500px;
    background: red;
    overflow: hidden;
    background: url("https://doc-08-2c-
docs.googleusercontent.com/docs/securesc/68c90smiglihng9534mvqmq194
6dmis5/fo0picsp1nhiucmc0l25s29respgpr4j/1631524275000/035223609609
22298374/03522360960922298374/1Sx0jhdpEpnNIydS4rnN4kHSJtU1Ey
Wka?e=view&authuser=0&nonce=gcrocepgbb17m&user=03522360960922

```

```

298374&hash=tfhgbs86ka6divo3llbvp93mg4csvb38") no-repeat center/
cover;
  border-radius: 10px;
  box-shadow: 5px 20px 50px #000;
}
#chk{
  display: none;
}
.signup{
  position: relative;
  width:100%;
  height: 100%;
}
label{
  color: #fff;
  font-size: 2.3em;
  justify-content: center;
  display: flex;
  margin: 60px;
  font-weight: bold;
  cursor: pointer;
  transition: .5s ease-in-out;
}
input{
  width: 60%;
  height: 20px;
  background: #e0dede;
  justify-content: center;
  display: flex;
  margin: 20px auto;
  padding: 10px;
  border: none;
  outline: none;
  border-radius: 5px;
}
button{
  width: 60%;
  height: 40px;
  margin: 10px auto;
  justify-content: center;

```

```
display: block;
color: #fff;
background: #573b8a;
font-size: 1em;
font-weight: bold;
margin-top: 20px;
outline: none;
border: none;
border-radius: 5px;
transition: .2s ease-in;
cursor: pointer;
}
button:hover{
    background: #6d44b8;
}
.login{
    height: 460px;
    background: #eee;
    border-radius: 60% / 10%;
    transform: translateY(-180px);
    transition: .8s ease-in-out;
}
.login label{
    color: #573b8a;
    transform: scale(.6);
}
#chk:checked ~ .login{
    transform: translateY(-500px);
}
#chk:checked ~ .login label{
    transform: scale(1);
}
#chk:checked ~ .signup label{
    transform: scale(.6);
}
```


CHAPTER – 6

SYSTEM TESTING

Software Testing Techniques:

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, designing and coding.

Testing Objectives:

- Testing is process of executing a program with the intent of finding an error.
- A good test case design is one that has a probability of finding an as yet undiscovered error.
- A successful test is one that uncovers an as yet undiscovered error.

These above objectives imply a dramatic change in view port.

Testing cannot show the absence of defects, it can only show that software errors are present.

There are three types of testing strategies

- Unit test
- Integration test
- Performance test

6.1 Unit Testing:

Unit testing focuses verification efforts on the smallest unit of software design module. The unit test is always white box oriented. The tests that occur as part of unit testing are testing the module interface, examining the local data structures, testing the boundary conditions, execution all the independent paths and testing error-handling paths.

6.2 Integration Testing:

Integration testing is a systematic technique or construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing summarizes the specific functional, performance, and internal design characteristics that are to be tested. It employs top-down testing and bottom-up testing methods for this case.

6.3 Performance Testing:

Timing for both read and update transactions should be gathered to determine whether system functions are being performed in an acceptable timeframe.

CHAPTER – 7

CONCLUSION

Presently we designed our Training & Placement Cell to be very User Friendly. Many features are enhanced to the present Training & Placement Cell. With this Training & Placement Cell, most of the TPO's time is saved. The features of the system can be further enhanced in many ways.

The documentation that has enclosed can enable even a person with minimum knowledge to understand it well. It can successfully login the authorized person to a system and register them.

In our system Admin can check the Student list those eligible according to criteria given by the Company and notify them instantly and update the information anytime successfully.

Our system is Secure and User-friendly for all three modules.

CHAPTER – 8

FUTURE SCOPE

- The project is easily extensible and can be improved by further incremental releases of the same.
- We can extend this project by sending emails to companies and student information regarding placement dates , other eligibility criteria also.
- New modules can easily added as it require only an addition of new package on button click.
- Further it is intended in the future to develop a separate module named training that would make use for student for their six months industrial training.

CHAPTER – 9

BIBLIOGRAPHY

- Santhosh Kumar H,” Online Training and Placement Management system” ,International Journal of Engineering Research Technology (IJERT),ICACT - 2016 ConferenceProceedings.
- Mythili M, Aishwarya R, Shenbagam P, Sandhiya C,” E Placement Management”, International Jour- nal of Pure and Applied Mathematics(IJPAM), Volume 119 No. 10 2018,1823-1834.
- Anjali v, Jeyalakshmi PR, Anubala R,Sri Mathura devi, Ranjini.V, ”Web Based Placement Management System”, International Journal of Computer Science and Information Technologies(IJCSIT), Vol. 7 (2) , 2016, 760-763.
- Rajnesh Tripathi, Raghvendra Singh, Jaweria Usmani, ”Campus Recruitment and Placement Sys- tem”, International Conference on Recent Innovations in Science and Engineering(Icrise-18), April, 2018