Project Report

on

Campus Placement Website

Submitted by

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

MASTER OF COMPUTER APPLICATIONS

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)

List of Chapters

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 Computerised 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

1. Student
2. Employer
3. 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 Tools and Environment

Hardware Requirements:

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

Software Requirements:

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

MODULES

The different types of modules present in this project are

1. Admin
2. Company
3. User(Candidates/ Students)

Module Description

1. Admin

a. Dashboard: In this section, admin can see all detail in brief like Total Company Registered, Total User (Candidates) Registered and Total Vacancy Listed.

b. Total Registered Company: In this section, admin can view detail of registered company.

c. Total Registered Users: In this section, admin can view detail of users.

d. Pages: In this section, the admin can manage about us and contact us pages.

e. 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.

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.

3. Company

The Company enter his user_id and Password and can login to his home page where he is having options to update the details and have a facility to post their vacancy .

Users have the privilege to see the students who are applied for that vacancy and view/download the details of accepted students.

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:

Tangible benefits:

- Saving man labor to do tedious tasks saves time.
-

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.

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

Student SignUp DB:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**UserName**	varchar(25)	latin1_swedish_ci		No	*None*			Change Drop More
☐	2	**Email**	varchar(56)	latin1_swedish_ci		No	*None*			Change Drop More
☐	3	**Password**	varchar(10)	latin1_swedish_ci		No	*None*			Change Drop More

Company Login DB:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**UserName**	varchar(10)	latin1_swedish_ci		No	*None*			Change Drop More
☐	2	**Password**	varchar(24)	latin1_swedish_ci		No	*None*			Change Drop More

Student Registration DB:

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

Admin Login DB:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**UserName**	varchar(10)	latin1_swedish_ci		No	*None*			Change Drop More
☐	2	**Password**	varchar(24)	latin1_swedish_ci		No	*None*			Change Drop More

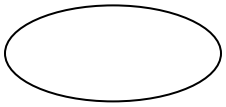
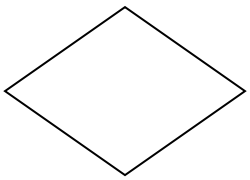
Company Detail DB:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**Details**	varchar(500)	latin1_swedish_ci		No	*None*			Change Drop More
☐	2	**Eligibilty**	varchar(150)	latin1_swedish_ci		No	*None*			Change Drop More
☐	3	**Form**	varchar(56)	latin1_swedish_ci		No	*None*			Change Drop More
☐	4	**Date**	datetime(5)			No	*None*			Change Drop More

3.2 ENTITY-RELATIONSHIP Diagrams

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

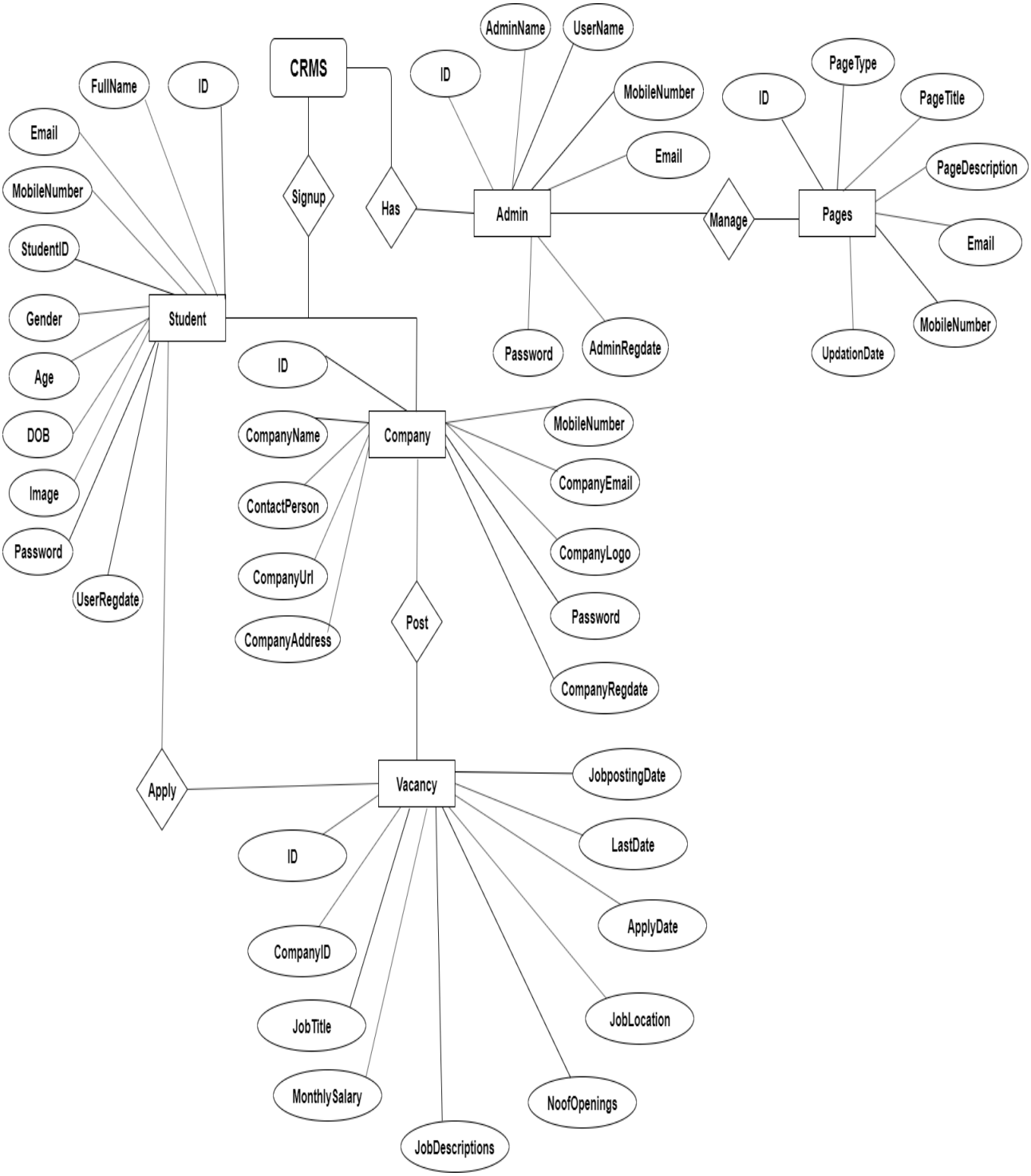
The symbols used in E-R diagrams are:

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.

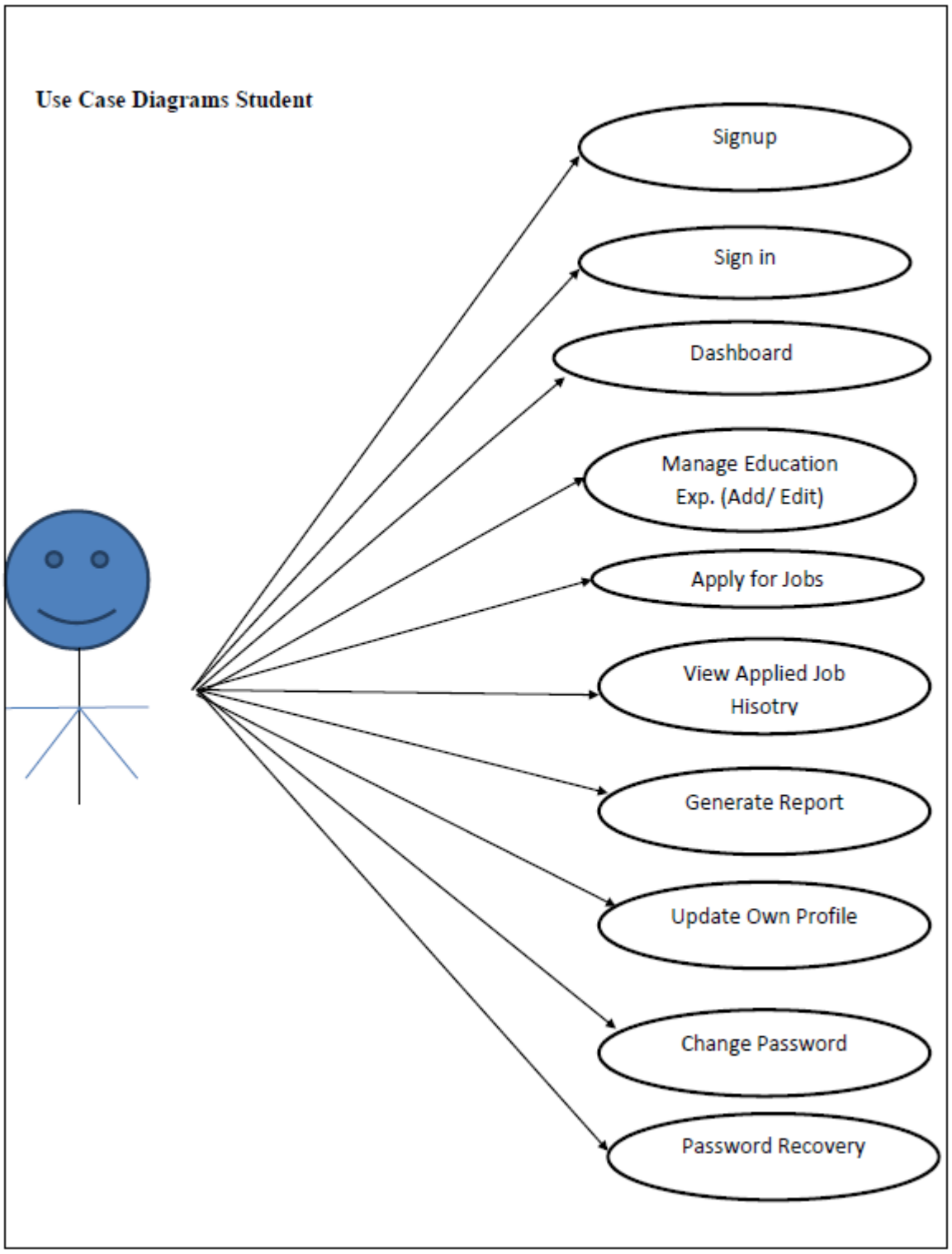
To develop a new kind of a system:

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

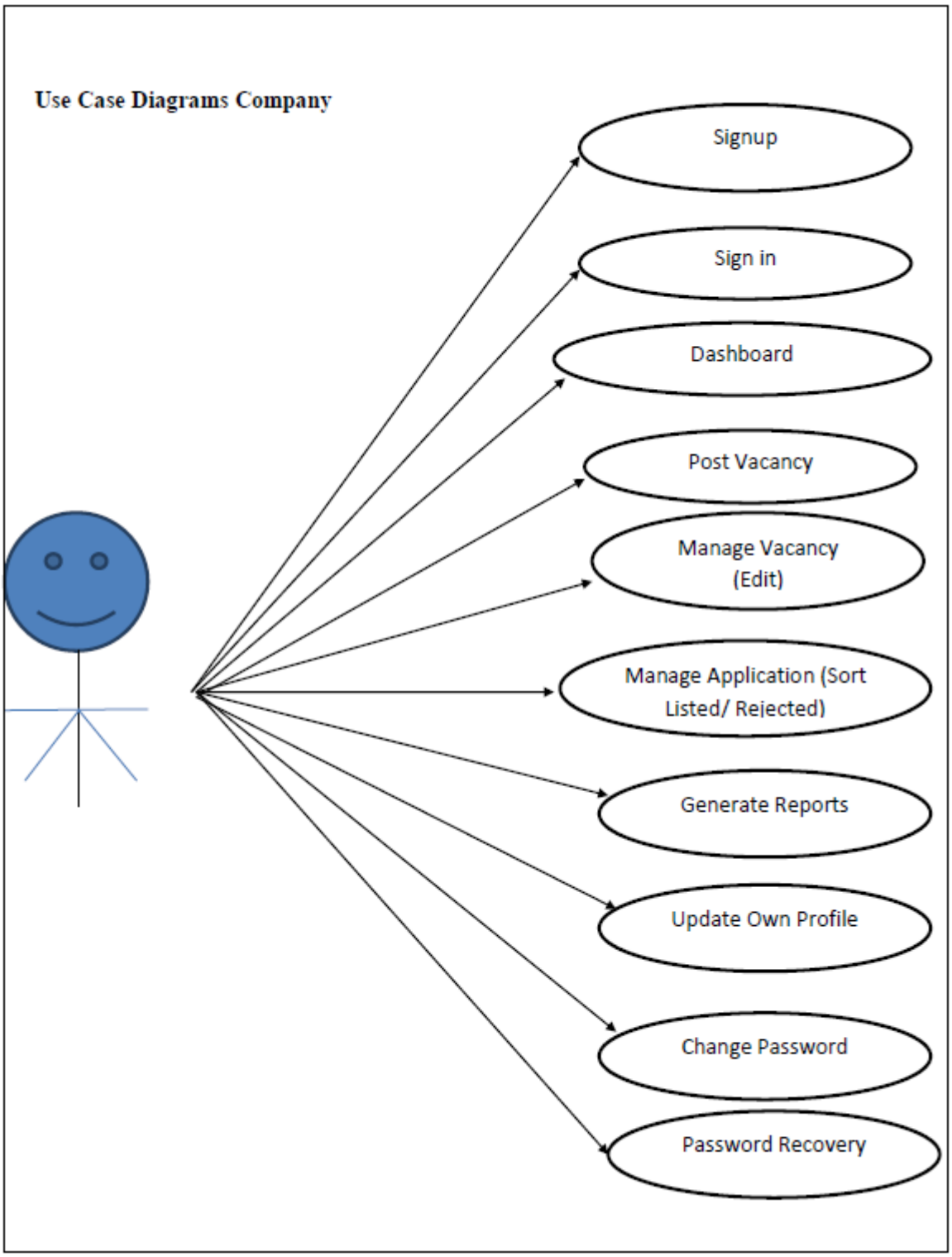

CRMS
Signup
Has
Admin
Manage
Pages
Student
Company
Post
Apply
Vacancy
AdminName
UserName
ID
MobileNumber
Email
Password
AdminRegdate
PageType
PageTitle
PageDescription
UpdationDate
FullName
StudentID
Gender
Age
DOB
Image
UserRegdate
CompanyName
ContactPerson
CompanyUrl
CompanyAddress
CompanyEmail
CompanyLogo
CompanyRegdate
JobpostingDate
LastDate
ApplyDate
JobLocation
NoofOpenings
JobDescriptions
MonthlySalary
JobTitle
CompanyID

3.3 USE CASE Diagrams

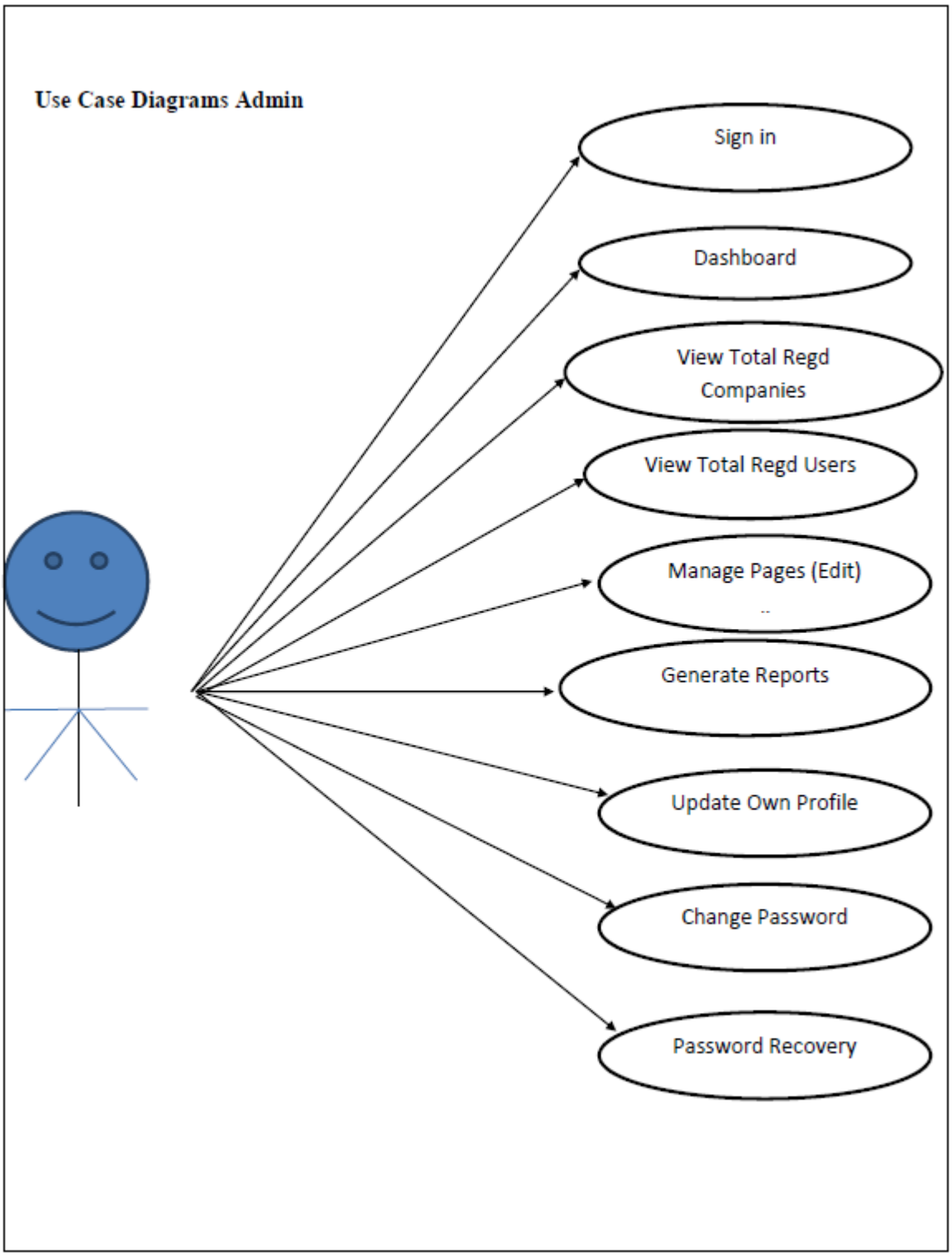
STUDENT



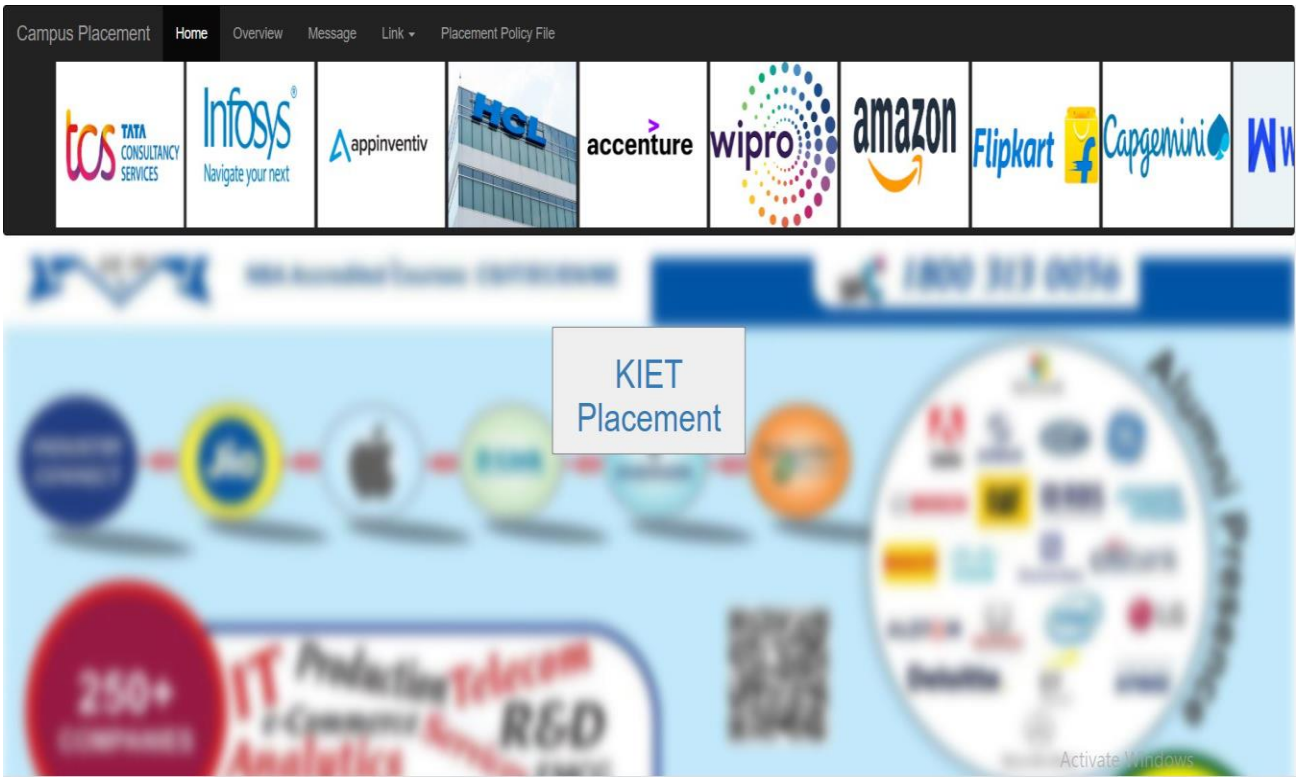
COMPANY

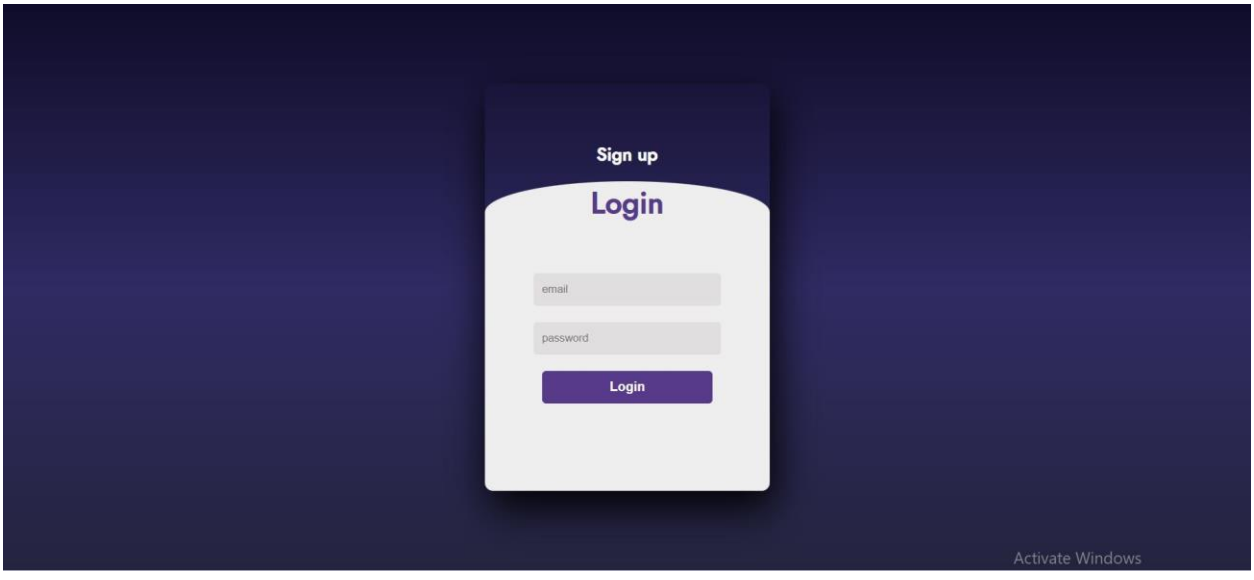

Use Case Diagrams Company
Signup
Sign in
Dashboard
Post Vacancy
Manage Vacancy (Edit)
Manage Application (Sort Listed/ Rejected)
Generate Reports
Update Own Profile
Change Password
Password Recovery

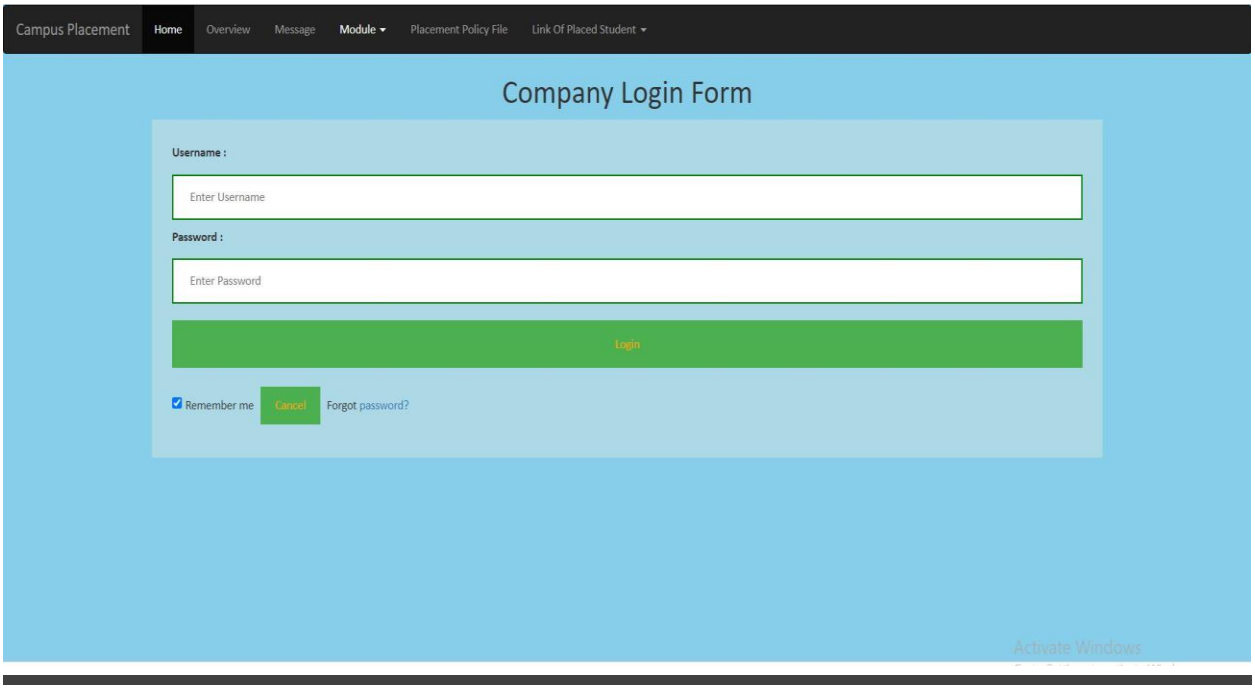
ADMIN

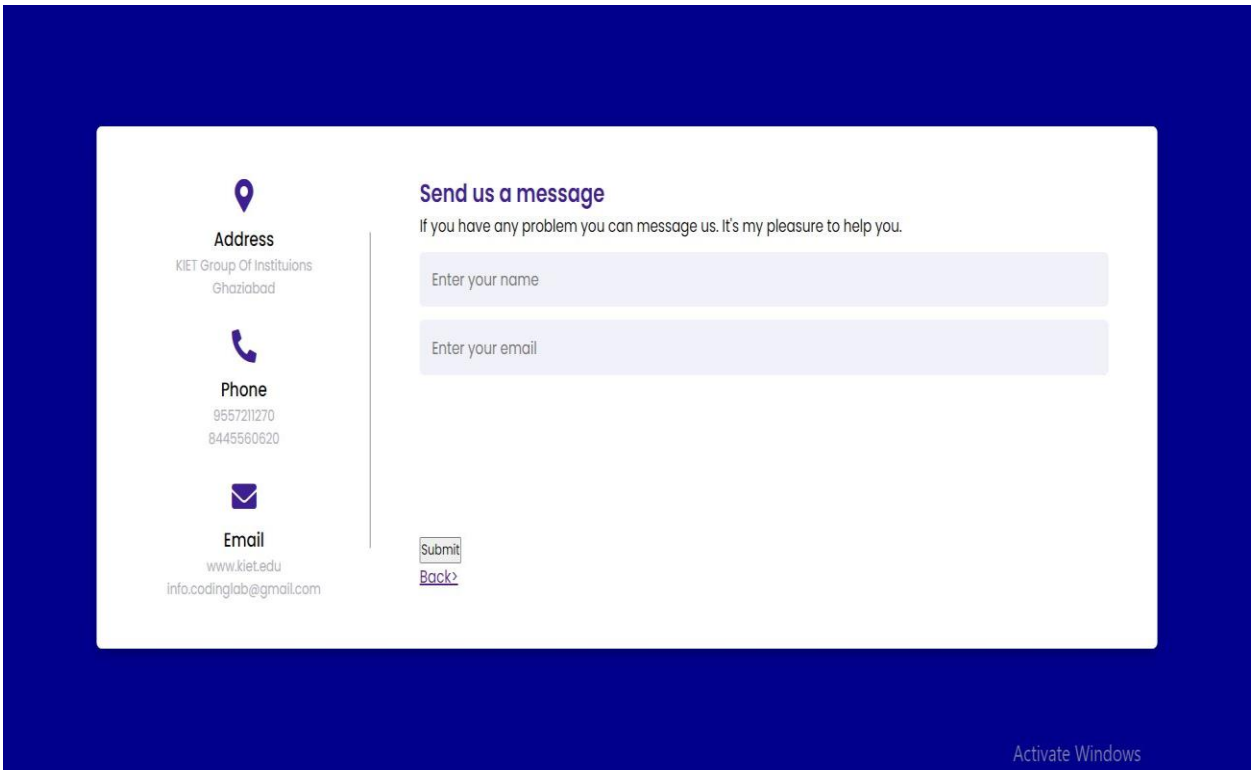

Use Case Diagrams Admin
Sign in
Dashboard
View Total Regd Companies
View Total Regd Users
Manage Pages (Edit)
Generate Reports
Update Own Profile
Change Password
Password Recovery

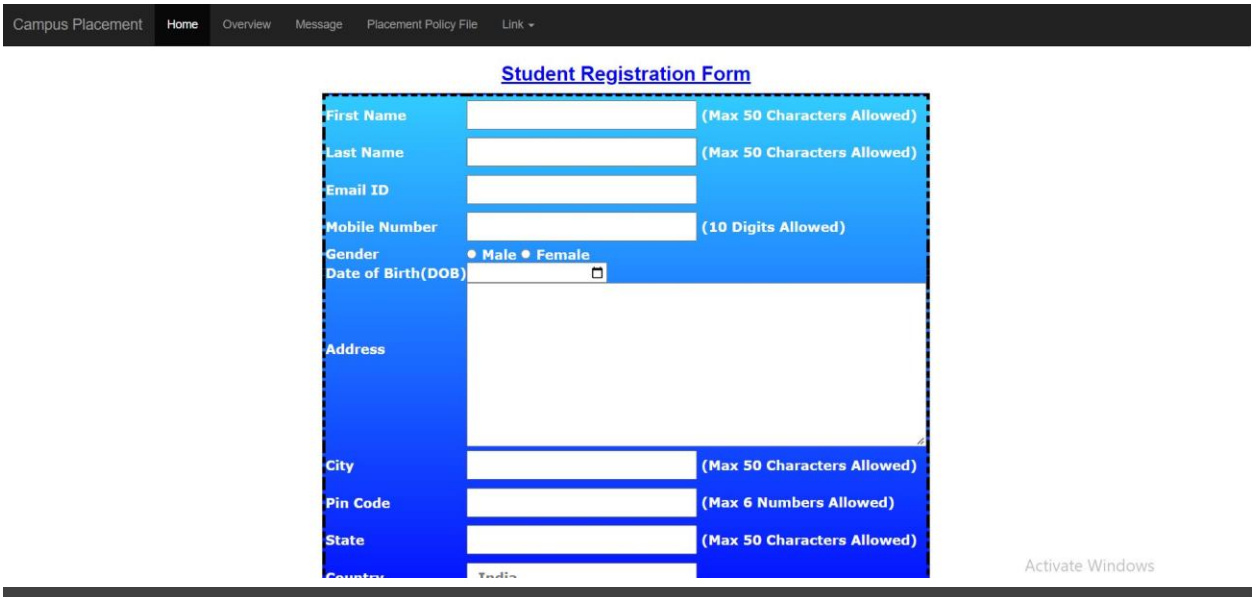
Output Screen of our Project






Campus Placement
Home
Overview
Message
Module
Placement Policy File
Link Of Placed Student
Company Login Form
Username :
Enter Username
Password :
Enter Password
Login
Remember me
Cancel
Forgot password?
Activate Windows


Address
KIET Group Of Instituions
Ghaziabad
Phone
9557211270
8445560620
Email
www.kiet.edu
info.codinglab@gmail.com
Send us a message
If you have any problem you can message us. It's my pleasure to help you.
Enter your name
Enter your email
Submit
Back>
Activate Windows


Campus Placement
Home
Overview
Message
Placement Policy File
Link
Student Registration Form
First Name
(Max 50 Characters Allowed)
Last Name
(Max 50 Characters Allowed)
Email ID
Mobile Number
(10 Digits Allowed)
Gender
Male
Female
Date of Birth(DOB)
Address
City
(Max 50 Characters Allowed)
Pin Code
(Max 6 Numbers Allowed)
State
(Max 50 Characters Allowed)
Activate Windows


Hiring Company
Logout
Exit
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
mtech
https://forms.gle/kHPip4wcVXtczCE59
capgemini
BSC, MCA
https://forms.gle/kHPip4wcVXtczCE59
Activate Windows

CODING : Module Wise

- ***Admin Module***

HTML

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


</style>
<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

```
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;
    }
```

- ***Company Module***

HTML

```
<!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

```
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;
    }
```

- ***Student Module***

HTML

```
<!DOCTYPE html>
<html>
<head>
    <title>Slide Navbar</title>
    <link rel="stylesheet" type="text/css" href="som.css">
<link href="https://fonts.googleapis.com/css2?family=Jost:wght@500&display=swap"
rel="stylesheet">
</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

```
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/68c90smiglihng9534mvqmq1946dmis5/fo0pics
p1nhiucmc0l25s29respgpr4j/1631524275000/03522360960922298374/03522360960922298374
/1Sx0jhdpEpnNIydS4rnN4kHSJtU1EyWka?e=view&authuser=0&nonce=gcrocepgbb17m&user=035
22360960922298374&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);
}
```

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:

1. Testing is process of executing a program with the intent of finding an error.

2. A good test case design is one that has a probability of finding an as yet undiscovered error.

3. 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

1. Unit test
2. Integration test

3. 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.