



PLACEMENT PORTAL

DATABASE MANAGEMENT PROJECT

MAY 2021

Bhagavathi Ravikrishnan, 2019103009

Kavya Sridhar, 2019103027

Ishwarya Sriram, 2019103528

CSE, R-Batch

College Of Engineering, Guindy



Contents

SNo.	Title of Contents	PageNo
1	Introduction	1
2	Abstract	1
3	Unique Idea Integration	2
4	The Application	2
5	ER Diagram	3
6	Relational Model	4
7	DDL Commands	5
8	DML Commands	8
9	List of Tables	10
10	Procedures	13
11	Transactions	14
12	Triggers	15
13	Front-end Screenshots	16
14	Contributions	27



Introduction

Every year, during the placement season, students are faced with a daunting challenge of making sure they don't miss out on any opportunities and the lack of a proper mechanism to help students with the process, leads to a lot of hurdles, uncertainties and missed opportunities. Therefore this project has identified this key issue and aims to resolve it through a Placement Portal by establishing a link between companies, students and the college.



Abstract

The Placement Portal is an application that allows students to sign up, fill out their resume and regularly update it, and view the details regarding the opportunities that are available to them. They will also be able to view the criteria for selection, salary packages, etc., and filter companies based on the parameters mentioned. Further, this portal will also have a channel for the companies that come to the campus to hire students, where they can list their requirements and the descriptions, and view the students' application details. All of these operations are spearheaded by the Admin who oversees the process and can add companies and optionally remove students and companies.



Unique Idea Integration

Providing Alumni Details

The details of the alumni who have gotten placed via this portal is made available to anyone who visits the application. This will enable visitors to view their details and gain guidance regarding the same,

The implementation of this idea is carried out through a trigger that moves the details of the applicant who has been accepted for a job from the students table to the alumni table, while deleting the records of their other applications to ensure that they are placed only in one job.

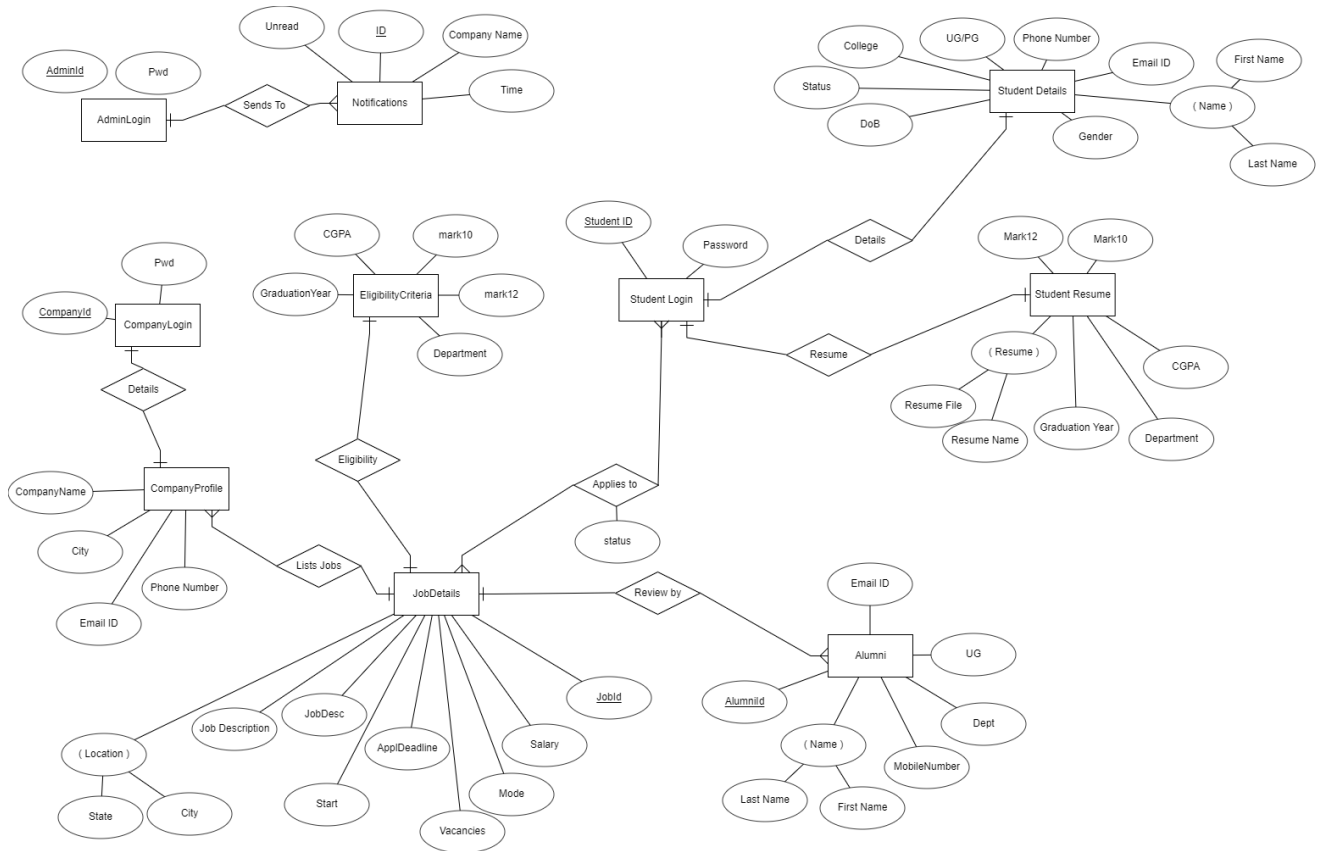


The Application

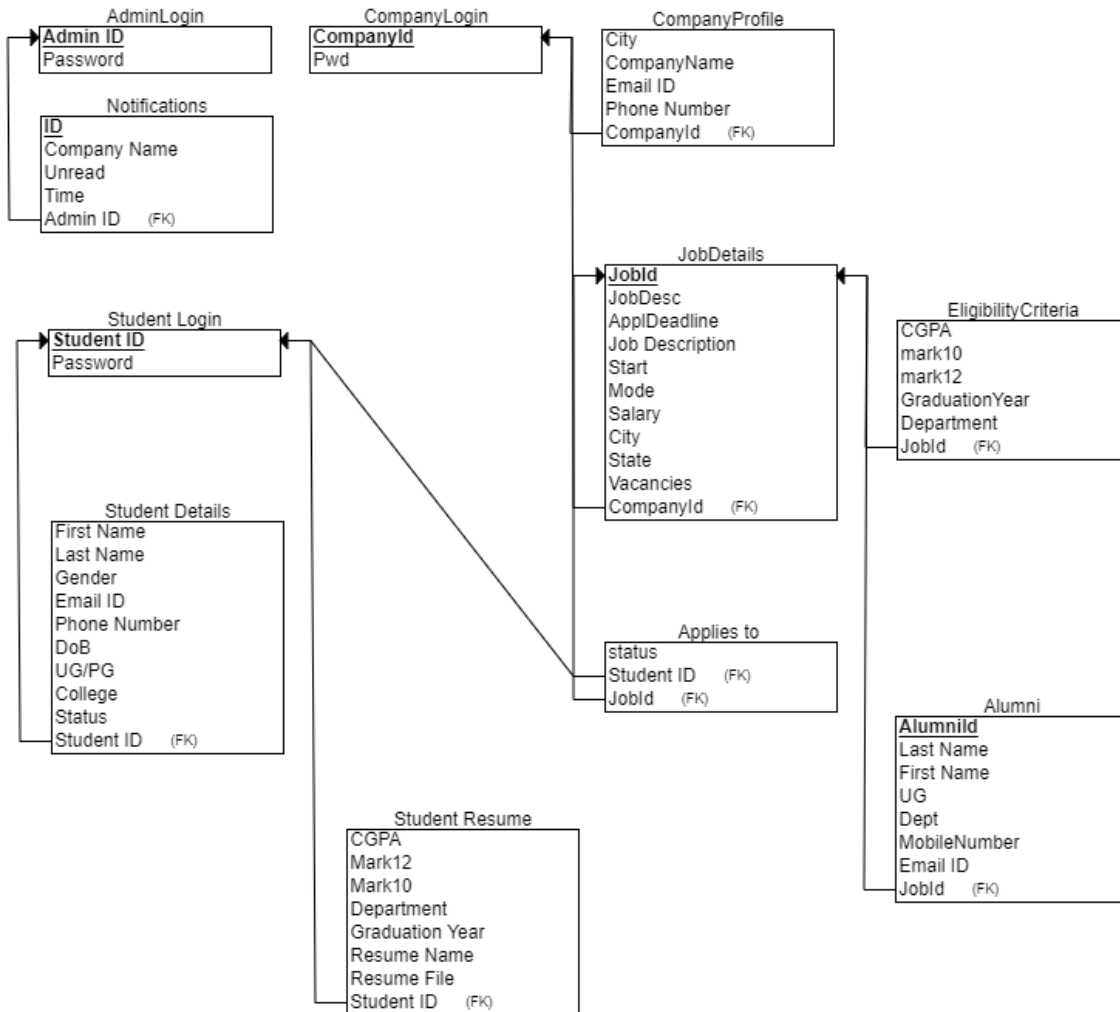
The following are the tools that have been utilised to implement this project

1. Front End - HTML/CSS/Javascript/jQuery
2. Backend - PHP, MySQL
3. Server - XAMPP

ER Diagram



Relational Model



The following database ensures maximum utilisation and minimal redundancy.

DDL Commands

CREATING DATABASE 'placement_portal'

```
Command Prompt - mysql -u root -p -h 127.0.0.1
Microsoft Windows [Version 10.0.19041.985]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Sridhar G>cd C:\xampp\mysql\bin

C:\xampp\mysql\bin>mysql -u root -p -h 127.0.0.1
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.25 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database placement_portal;
Query OK, 1 row affected (0.01 sec)

mysql> CREATE TABLE AdminLogin
-> (
->   AdminId INT NOT NULL,
->   Pwd VARCHAR(255) NOT NULL,
->   PRIMARY KEY (AdminId)
-> );
ERROR 1046 (3D000): No database selected
mysql> use placement_portal;
Database changed
mysql>
```

CREATING TABLES

```
Command Prompt - mysql -u root -p -h 127.0.0.1
Query OK, 1 row affected (0.01 sec)

mysql> CREATE TABLE AdminLogin
-> (
->   AdminId INT NOT NULL,
->   Pwd VARCHAR(255) NOT NULL,
->   PRIMARY KEY (AdminId)
-> );
ERROR 1046 (3D000): No database selected
mysql> use placement_portal;
Database changed
mysql> CREATE TABLE AdminLogin
-> (
->   AdminId INT NOT NULL,
->   Pwd VARCHAR(255) NOT NULL,
->   PRIMARY KEY (AdminId)
-> );
Query OK, 0 rows affected (0.05 sec)

mysql> CREATE TABLE CompanyLogin
-> (
->   Pwd VARCHAR(255) NOT NULL,
->   CompanyId INT NOT NULL,
->   PRIMARY KEY (CompanyId)
-> );
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE TABLE CompanyProfile
-> (
->   VaccanciesCount INT NOT NULL,
->   CompanyName INT NOT NULL,
->   CompanyId INT NOT NULL,
->   PRIMARY KEY (CompanyId),
->   FOREIGN KEY (CompanyId) REFERENCES CompanyLogin(CompanyId)
-> );
```

adminlogin, companylogin and companyprofile

```

C:\ Command Prompt - mysql -u root -p -h 127.0.0.1
mysql> CREATE TABLE JobDetails
-> (
-> JobId INT NOT NULL,
-> JobDesc VARCHAR(255) NOT NULL,
-> ApplDeadline DATE NOT NULL,
-> CompanyId INT NOT NULL,
-> PRIMARY KEY (JobId),
-> FOREIGN KEY (CompanyId) REFERENCES CompanyProfile(CompanyId)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE TABLE EligibilityCriteria
-> (
-> CGPA FLOAT NOT NULL,
-> mark10 INT NOT NULL,
-> mark12 INT NOT NULL,
-> JobId INT NOT NULL,
-> CompanyId INT NOT NULL,
-> FOREIGN KEY (JobId) REFERENCES JobDetails(JobId),
-> FOREIGN KEY (CompanyId) REFERENCES CompanyProfile(CompanyId)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE TABLE Alumni
-> (
-> AlumniId INT NOT NULL,
-> FirstName VARCHAR(255) NOT NULL,
-> LastName VARCHAR(255) NOT NULL,
-> MobileNumber NUMERIC(10) NOT NULL,
-> UG VARCHAR(255) NOT NULL,
-> Department VARCHAR(255) NOT NULL,
-> Review VARCHAR(255) NOT NULL,
-> JobId INT NOT NULL,
-> CompanyId INT NOT NULL,
-> PRIMARY KEY (AlumniId),

```

jobdetails, eligibilitycriteria and alumni

```

C:\ Command Prompt - mysql -u root -p -h 127.0.0.1
mysql>
mysql> CREATE TABLE StudentLogin
-> (
-> StudentId INT NOT NULL,
-> pwd VARCHAR(255) NOT NULL,
-> PRIMARY KEY (StudentId)
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE TABLE StudentDetails
-> (
-> FirstName VARCHAR(255) NOT NULL,
-> LastName VARCHAR(255) NOT NULL,
-> DoB DATE NOT NULL,
-> EmailId VARCHAR(255) NOT NULL,
-> MobileNumber NUMERIC(10) NOT NULL,
-> UG VARCHAR(255) NOT NULL,
-> Department VARCHAR(255) NOT NULL,
-> StudentId INT NOT NULL,
-> PRIMARY KEY (StudentId),
-> FOREIGN KEY (StudentId) REFERENCES StudentLogin(StudentId)
-> );
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE TABLE StudentResume
-> (
-> ResumeID INT NOT NULL,
-> CGPA INT NOT NULL,
-> mark10 INT NOT NULL,
-> mark12 INT NOT NULL,
-> StudentId INT NOT NULL,
-> PRIMARY KEY (ResumeID),
-> FOREIGN KEY (StudentId) REFERENCES StudentLogin(StudentId)
-> );
Query OK, 0 rows affected (0.02 sec)

mysql>

```

studentlogin , studentdetails and studentresume


```
mysql> CREATE TABLE `jobappl` (
  -> `Status` char(1) NOT NULL,
  -> `studentid` int(11) DEFAULT NULL,
  -> `JobId` varchar(10) DEFAULT NULL
  -> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected, 2 warnings (0.08 sec)

mysql> CREATE TABLE `notifications` (
  -> `id` int(10) UNSIGNED NOT NULL,
  -> `companyName` varchar(100) NOT NULL,
  -> `unread` tinyint(1) NOT NULL DEFAULT 1,
  -> `time` datetime DEFAULT current_timestamp(),
  -> `AdminId` varchar(10) NOT NULL DEFAULT 'admin'
  -> ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
Query OK, 0 rows affected, 3 warnings (0.19 sec)

mysql>
```

jobapply and notifications

ALTERING TABLES

```
Command Prompt - mysql -u root -p -h 127.0.0.1
mysql> desc adminlogin ;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| AdminId | int           | NO   | PRI | NULL    |       |
| Pwd     | varchar(255)  | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> ALTER TABLE adminlogin MODIFY AdminId varchar(10);
Query OK, 1 row affected (0.07 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> desc adminlogin;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| AdminId | varchar(10)   | NO   | PRI | NULL    |       |
| Pwd     | varchar(255)  | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Modifying AdminId from int to varchar

```
mysql> alter table studentresume add (Department varchar(255));
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table studentresume add (GradYear int(4));
Query OK, 0 rows affected, 1 warning (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 1
```

Adding column Department to the studentresume table

```
mysql> desc studentdetails;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| FirstName | varchar(255) | NO | | NULL | |
| LastName | varchar(255) | NO | | NULL | |
| DoB | date | NO | | NULL | |
| EmailId | varchar(255) | NO | | NULL | |
| MobileNumber | decimal(10,0) | NO | | NULL | |
| UG | varchar(255) | NO | | NULL | |
| Department | varchar(255) | NO | | NULL | |
| StudentId | int | NO | PRI | NULL | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> alter table studentdetails drop Department;
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table studentdetails add College varchar(255);
Query OK, 0 rows affected (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> desc studentdetails;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| FirstName | varchar(255) | NO | | NULL | |
| LastName | varchar(255) | NO | | NULL | |
| DoB | date | NO | | NULL | |
| EmailId | varchar(255) | NO | | NULL | |
| MobileNumber | decimal(10,0) | NO | | NULL | |
| UG | varchar(255) | NO | | NULL | |
| StudentId | int | NO | PRI | NULL | |
| College | varchar(255) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)

mysql>
```

Dropping Department and adding College to the table studentdetails

DML Commands

INSERT/UPDATE/SELECT/DELETE

INSERTIONS

Inserting admin username/password via command line interface

```
Command Prompt - mysql -u root -p -h 127.0.0.1

mysql> desc adminlogin;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| AdminId | varchar(10) | NO | PRI | NULL | |
| Pwd | varchar(255) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> insert into adminlogin values ('admin','root');
Query OK, 1 row affected (0.00 sec)

mysql> select * from adminlogin;
+-----+-----+
| AdminId | Pwd |
+-----+-----+
| admin | root |
+-----+-----+
1 row in set (0.00 sec)
```

The rest of the insertions are done through the User Interface in the application.

```

$stmt = Database::$conn->prepare("INSERT INTO jobdetails
    (CompanyId, JobId, JobDesc, ApplDeadline, Start,Mode, Salary, City, State, Vacancies)
    VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)");
$stmt->bind_param("sssssssss",$cid,$jid,$desc,$ad,$start,$mode,$salary,$city, $state,$vac);
$res= $stmt->execute();
if(!$res)
    echo("Error: ".$stmt->error);
else
    echo "<script>alert('Job upload success')</script>";

$stmt = Database::$conn->prepare("INSERT INTO eligibilitycriteria
    (JobId, cgpa,mark12,mark10,department,gradyear)
    VALUES (?, ?, ?, ?, ?, ?)");
$stmt->bind_param("ssssss",$jid,$gpa,$mark12,$mark10,$chk,$gradyear);
$res= $stmt->execute();
if(!$res)
    echo("Error: ".$stmt->error);
else
{
    echo "<script>alert('Criteria upload success')</script>";
    echo "<script>location.replace('../dashboardpage/mainindex.php');</script>";
}
}

```

sample screenshot of how the insert commands are executed in PHP

UPDATES

Updates are performed through PHP and Stored Procedures

```

if(count($_POST)>0){
    // var_dump($_POST);
    if(mysqli_query(Database::$conn,"update companyprofile set
        Email='".$_POST['EmailId']."' , PhoneNumber='".$_POST['MobileNumber']."'
        , City='".$_POST['City']."' WHERE CompanyId='".$_GET['CompanyId']."' . ""))
    {
        $message="Success";
        echo "<script>alert('Updated Successfully')</script>";
        echo "<script>location.replace('mainindex.php');</script>";
    }
    else{
        echo mysqli_error(Database::$conn);
    }
}

```

Sample screenshot showing updation through PHP

SELECTIONS

Select queries are executed via PHP to display details from the database.

```
<?php
    $sql = "SELECT * FROM jobappl inner join jobdetails on jobappl.JobId = jobdetails.JobId
    where jobdetails.CompanyId ='" . $_SESSION['username'] . "' and jobappl.Status='y'";
    $result = mysqli_query(Database::$conn,$sql);
    $rowCount = mysqli_num_rows($result);
    echo($rowCount);
?>
```

```
<?php
    require_once 'dbconnect.php';
    $id=$_SESSION['username'];
    //var_dump($_SESSION);
    if(!mysqli_query(Database::$conn,"SELECT * FROM studentdetails where studentid=$id")){
        echo mysqli_error(Database::$conn);
    }
    $result=mysqli_query(Database::$conn,"SELECT * FROM studentdetails where studentid=$id");
?>
```

List of all tables

```
Command Prompt - mysql -u root -p -h 127.0.0.1
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 403
Server version: 8.0.25 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use placement_portal;
Database changed
mysql> show tables;
+-----+
| Tables_in_placement_portal |
+-----+
| adminlogin                  |
| alumni                      |
| companylogin                |
| companyprofile              |
| eligibilitycriteria          |
| jobappl                     |
| jobdetails                  |
| notifications                |
| studentdetails              |
| studentlogin                |
| studentresume                |
+-----+
11 rows in set (0.00 sec)

mysql>
```

```
Command Prompt - mysql -u root -p -h 127.0.0.1
| studentdetails |
| studentlogin   |
| studentresume  |
+-----+
11 rows in set (0.00 sec)

mysql> select * from adminlogin;
+-----+
| AdminId | Pwd |
+-----+
| admin   | newroot |
+-----+
1 row in set (0.00 sec)

mysql> select * from alumni;
+-----+
| AlumniId | FirstName | LastName | Email | MobileNumber | UG |
+-----+
| 2019103006 | Athirai | S | athirai@gmail.com | 5634556673 | UG |
| 2019103009 | bhargav | ravi | baggy@gmail.com | 8734646777 | UG |
| 2019103027 | Kavya | Sridhar | kavya@gmail.com | 6437733776 | PG |
+-----+
3 rows in set (0.00 sec)

mysql> select * from companylogin;
+-----+
| Pwd | CompanyId |
+-----+
| password | 4 |
| password | 5 |
| password | 6 |
| password | 7 |
| password | 8 |
+-----+
5 rows in set (0.00 sec)

mysql> \
mysql>
```

Adminlogin, alumni and companylogin

```
Command Prompt - mysql -u root -p -h 127.0.0.1
Database changed
mysql> select * from companyprofile;
+-----+
| CompanyName | CompanyId | Email | PhoneNumber | City |
+-----+
| Samsung | 4 | s@gmail.com | 564546446 | Hyderabad |
| Infosys | 5 | inf@gmail.com | 2147483647 | Chennai |
| Amazon | 6 | a@gmail.com | 676767565 | Delhi |
| Microsoft | 7 | mic@gmail.com | 2147483647 | Chennai |
| test_comp | 8 | testcomp@gmail.com | 2147483647 | Mumbai |
+-----+
5 rows in set (0.00 sec)

mysql> select * from eligibilitycriteria;
+-----+
| CGPA | mark10 | mark12 | GradYear | Department | JobId |
+-----+
| 7 | 70 | 70 | 2022 | CSE,IT, | 4Sam0 |
| 8 | 75 | 70 | 2023 | CSE,ECE, | 4Sam1 |
| 9 | 90 | 90 | 2023 | CSE,IT,ECE, | 5Inf2 |
| 9 | 90 | 90 | 2022 | IT,ECE, | 6Ama3 |
| 8 | 80 | 80 | 2023 | CSE,IT,Mech, | 7Mic4 |
| 7.5 | 94 | 95 | 2023 | CSE,IT, | 8tes5 |
+-----+
6 rows in set (0.00 sec)

mysql> select * from jobappl;
+-----+
| Status | studentid | JobId |
+-----+
| y | 2019103006 | 6Ama3 |
| n | 2019103015 | 4Sam0 |
| n | 2019103015 | 6Ama3 |
| n | 2019103015 | 7Mic4 |
| y | 2019103027 | 8tes5 |
+-----+
5 rows in set (0.00 sec)

mysql>
```

companyprofile, eligibilitycriteria and jobappl

```
Command Prompt - mysql -u root -p -h 127.0.0.1
5 rows in set (0.00 sec)

mysql> select * from jobdetails;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| JobId | JobDesc | ApplDeadline | CompanyId | Start | Mode | Salary | City | State | Vacancies |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 4Sam0 | Software Engineer required | 2021-06-05 | 4 | 10/June | Online | 100000 | Hyderabad | Telangana | 4 |
| 4Sam1 | Software Engineer Front end | 2021-06-06 | 4 | 15/June | Offline | 120000 | Hyderabad | Telangana | 8 |
| 5Inf2 | Software backend | 2021-06-06 | 5 | 8/June | Mixed | 50000 | Chennai | Tamil Nadu | 6 |
| 6Ana2 | Manager | 2021-07-07 | 6 | 8/November | Offline | 70000 | Delhi | Delhi | 7 |
| 7M4c4 | Backend Engineer | 2021-06-05 | 7 | 7/October | Offline | 67000 | Chennai | Tamil Nadu | 7 |
| 8tes5 | test description | 2021-07-13 | 8 | July 2022 | Online | 25630 | Chennai | Tamil Nadu | 2 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> select * from notifications;
+-----+-----+-----+-----+-----+
| Id | companyName | unread | time | AdminId |
+-----+-----+-----+-----+-----+
| 4 | Samsung | 0 | 2021-05-30 19:50:00 | admin |
| 5 | Infosys | 0 | 2021-05-30 19:50:17 | admin |
| 6 | Wells Fargo | 0 | 2021-05-30 19:50:28 | admin |
| 7 | Microsoft | 0 | 2021-05-30 19:50:38 | admin |
| 8 | test_comp | 0 | 2021-05-31 10:01:33 | admin |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> select * from studentdetails;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| FirstName | LastName | DoB | EmailId | MobileNumber | UG | StudentId | College | Gender | Status |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Max | doe | 2000-06-06 | v@gmail.com | 7364767676 | PG | 2019103000 | CEG | other | b |
| Athirai | S | 2001-11-30 | athirai@gmail.com | 5634556673 | UG | 2019103006 | MIT | female | y |
| Bhargav | ravi | 2002-01-15 | baggy@gmail.com | 8734646777 | UG | 2019103009 | CEG | male | y |
| Dhivya | Ramesh | 2001-02-14 | dhivya@gmail.com | 7637738748 | PG | 2019103015 | ACTech | female | n |
| Kavya | Sridhar | 2001-05-15 | kavya@gmail.com | 6437733776 | PG | 2019103027 | CEG | female | y |
| Neha | Susan | 2001-12-18 | neha@gmail.com | 7364746476 | PG | 2019103042 | MIT | female | n |
| Samhit | Mahadevan | 2001-06-18 | sammy@gmail.com | 6748363787 | PG | 2019103054 | MIT | male | n |
| Shrishti | Padmanabhan | 2001-06-10 | shrishti@gmail.com | 8476545677 | UG | 2019103060 | CEG | female | n |
| eeshwaran | sriram | 2001-06-01 | eesh@gmail.com | 6776768788 | UG | 2019103528 | CEG | male | n |
| Kavya | Gopal | 2021-05-15 | kavya@gmail.com | 7845124596 | UG | 2147483647 | CEG | female | n |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

jobdetails, notifications and studentdetails

```
Command Prompt - mysql -u root -p -h 127.0.0.1

mysql>
mysql> select * from studentlogin;
+-----+-----+
| StudentId | pwd |
+-----+-----+
| 2019103000 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103006 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103009 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103015 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103027 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103042 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103054 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103060 | 5d41402abc4b2a76b9719d911017c592 |
| 2019103528 | 5d41402abc4b2a76b9719d911017c592 |
| 2147483647 | 5d41402abc4b2a76b9719d911017c592 |
+-----+-----+
10 rows in set (0.00 sec)

mysql> select * from studentresume;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| CGPA | mark12 | StudentId | Department | GradYear | file | filename | mark10 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 10 | 95 | 2019103006 | EEE | 2023 | 0x443A5C785C746D705C706870454632432E746D70 | new_resume_athirai.pdf | 95 |
| 10 | 90 | 2019103015 | Chemical Technology | 2024 | 0x443A5C785C746D705C706870354246332E746D70 | resume_dhivyaR.pdf | 90 |
| 9.8 | 90 | 2019103027 | CS | 2022 | 0x443A5C785C746D705C706870454333342E746D70 | Resume_KavyaSridhar.pdf | 95 |
| 9.7 | 90 | 2019103042 | CS | 2023 | 0x443A5C785C746D705C706870454531382E746D70 | resumr.pdf | 90 |
| 7.7 | 79 | 2019103054 | ECE | 2022 | 0x443A5C785C746D705C706870314142382E746D70 | resumr.pdf | 89 |
| 9 | 89 | 2019103060 | CS | 2023 | 0x443A5C785C746D705C706870414345392E746D70 | resume_shrishti.pdf | 90 |
| 8 | 67 | 2019103528 | CS | 2022 | 0x443A5C785C746D705C706870314230412E746D70 | resume_eesh.pdf | 60 |
| 9 | 94 | 2147483647 | CSE | 2023 | 0x433A5C78616D70705C746D705C7068703242432E746D70 | SAMPLE_RESUME.pdf | 95 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

Studentresume and studentlogin

SQL Procedures

Stored Procedures are used in the application for updations/insertions

Creating Stored Procedure and calling Stored Procedure for updating Admin Password:

```
mysql> CREATE PROCEDURE AdmPwdChange(IN replpwd VARCHAR(255))
-> BEGIN
-> UPDATE adminlogin SET pwd = replpwd WHERE AdminId = 'admin';
-> COMMIT;
-> END
-> //
```

Query OK, 0 rows affected (0.02 sec)

```
mysql>
```

```
if(mysql_query(Database::$conn,"SELECT * from adminlogin where AdminId='".$$_GET['AdminId']."'")){
    if($newPwd == $confirmPwd){
        $pass = $newPwd;
        if(mysql_query(Database::$conn,"CALL AdmPwdChange('$pass')")){
            $message="Success";
            echo "<script>alert('Updated Successfully');</script>";
            echo "<script>location.replace('mainindex.php');</script>";
        }
        else{
            echo mysqli_error(Database::$conn);
        }
    }
}
```

Creating Stored Procedure and calling Stored Procedure for inserting notification

```
mysql> CREATE PROCEDURE insert_notif(IN Compname varchar(100))
-> BEGIN
-> INSERT INTO notifications(companyName,time) values (Compname,NOW());
-> END
-> //
```

Query OK, 0 rows affected (0.04 sec)

```
if(isset($_POST['reg_user']))
{
    $compname=$_POST['name'];
    if(mysql_query(Database::$conn,"CALL insert_notif('$compname')")){
        $message="Success";
        echo "<script>alert('Request Submitted Successfully. Please Wait for Admin to authorize account');</script>";
        echo "<script>location.replace('../mainpage/index.html');</script>";
    }
    else{
        echo mysqli_error(Database::$conn);
    }
}
```

Transactions

A simple SQL transaction is used in this application when the admin adds a new company to the portal

```
adminlogin > dashboardpage > managecomp.php
1  <?php
2  function apply_transaction($id,$name)
3  {
4      mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);
5      $conn=new mysqli('localhost','root','','placement_portal');
6      $conn->set_charset('utf8mb4');
7      $conn->begin_transaction();
8      try{
9          $query=$conn->query("select * from companylogin where companyId='$id'");
10         $row=$query->fetch_assoc();
11         if(!$row)
12         {
13             $query=$conn->query("UPDATE notifications set unread='0' where id='$id'");
14             if($query)
15             {
16                 $sql = "INSERT INTO companylogin (CompanyId, Pwd)
17                     VALUES ('$id','password')";
18
19                 if ($conn->query($sql) === TRUE) {
20                     $conn->commit();
21                     echo "<script>alert('New record created successfully');</script>";
22                     echo "<script>location.replace('mainindex.php');</script>";
23                 } else {
24                     echo "<script>alert('Error creating company');</script>";
25                     echo "<script>location.replace('mainindex.php');</script>";
26                 }
27             }
28         }
29         else{
30             echo "<script>alert('Error creating company');</script>";
31
32             echo "<script>location.replace('mainindex.php');</script>";
33         }
34     } else{
35         echo "<script>alert('Company already exists');</script>";
36         echo "<script>location.replace('mainindex.php');</script>";
37     }
38 }catch(mysqli_sql_exception $exception){
39     $mysqli->rollback();
40     throw $exception;
41 }
42 }
43
44
45 if($_GET['addcomp']){
46     $name=$_GET['name'];
47     $id=$_GET['id'];
48     apply_transaction($id,$name);
49 }
50
51
52 }
```

The transaction ensures that integrity of the data is maintained when the admin views the unread notifications and inserts the data into the table. If at any point in the execution an operation fails, the transaction performs a ROLLBACK, and if no problems are encountered, the transaction performs a COMMIT.

Triggers

Creating trigger 'ALUMNI' to insert student details into the alumni table when they get placed.

```
mysql> delimiter //
mysql> CREATE TRIGGER `ALUMNI` AFTER UPDATE ON `studentdetails`
-> FOR EACH ROW if (NEW.Status = 'y')
-> THEN
->     INSERT INTO `alumni`(`AlumniId`, `FirstName`, `LastName`, `Email`, `MobileNumber`, `UG`)VALUES(NEW.StudentId,NEW.FirstName,NEW.LastName,NEW.EmailId,NEW.MobileNumber,NEW.UG);
->     end if
-> //
Query OK, 0 rows affected (0.01 sec)

mysql>
```

Creating trigger 'VACANCY' to decrement the number of vacant jobs when a student gets placed.

```
mysql> delimiter //
mysql> CREATE TRIGGER `VACANCY` AFTER UPDATE ON `jobappl`
-> FOR EACH ROW if (NEW.Status = 'y')
-> THEN
->     UPDATE `jobdetails` SET `Vacancies`=`Vacancies`-1 WHERE `JobId` = NEW.`JobId`;
->     end if
-> //
Query OK, 0 rows affected (0.01 sec)

mysql> █
```

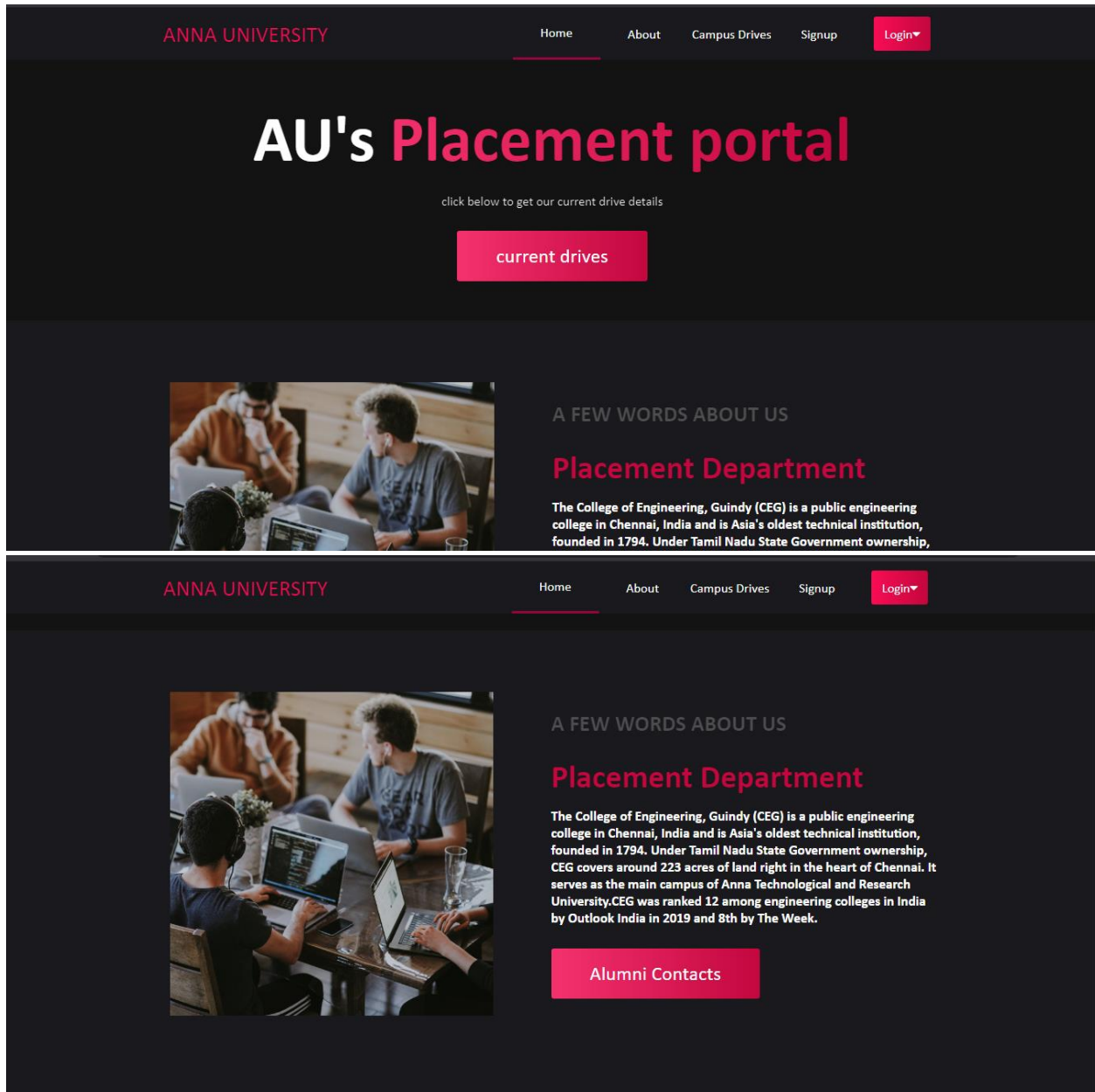
Creating trigger 'BANS_STUDENT' to delete student records from studentresume and jobappl tables when a student is removed by the admin.

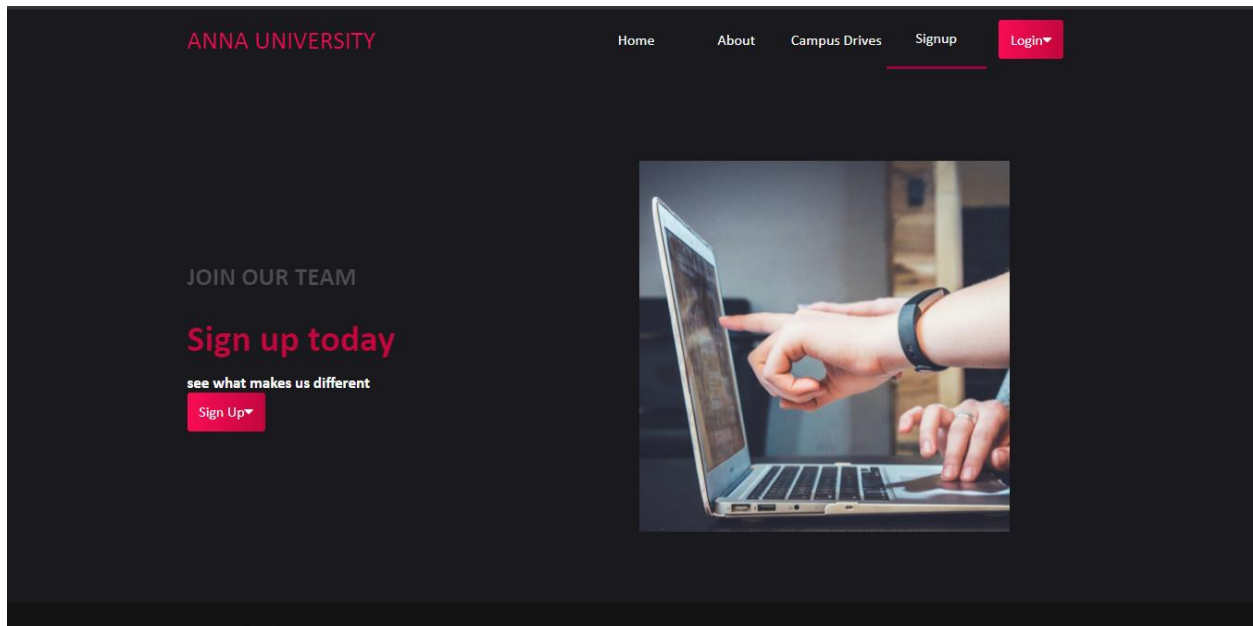
```
mysql> delimiter //
mysql> CREATE TRIGGER `BANS_STUDENT` AFTER UPDATE ON `studentdetails`
-> FOR EACH ROW BEGIN
-> DELETE FROM `jobappl` WHERE StudentId=OLD.StudentId;
->     DELETE FROM `studentresume` WHERE StudentId=OLD.StudentId;
-> END
-> //
Query OK, 0 rows affected (0.01 sec)

mysql> █
```

Front-End Screenshots

HOME PAGE





STUDENT LOGIN

Student Login Page

Student Username

Enter username

Password

Enter password

Login

Not yet a member? [Sign up](#)

STUDENT REGISTRATION

Student Registration Page

Student Username

Password

Confirm Password

Register

Already a member? [Sign in](#)

STUDENT DASHBOARD

Placement Portal

Dashboard

Resume

Apply

Settings

Sign Out

Welcome,
2019103027

4
Total Companies

5
Total Jobs

1
Jobs Applied to

Not Placed
Status

Table

View All

Job ID	Company ID	Company Name	Status
4Sam0	4	Samsung	Not Placed

STUDENT RESUME

Placement Portal

Dashboard

Resume

Apply

Settings

Sign Out

Welcome,
2019103027

Resume Details

Student ID

2019103027

Department

CS

Year of Graduation

2022

CGPA

9.8

Class 12 percentage

90

Class 10 percentage

95

Upload Resume

Upload .pdf file

Upload

Resume_KavyaSridhar.pdf

Submit

STUDENT PLACEMENT APPLICATION

Welcome,
2019103027

PLACEMENT APPLICATION

Enter








Filter

Search

Reset

JOB ID	COMPANY NAME	JOB DESC	NO. OF VACANCIES	CGPA Required	YEAR OF GRADUATION	DEPARTMENT(S)	START	MODE	SALARY	CITY NAME	DEADLINE	APPLY
4Sam0	Samsung	Software Engineer required	4	7	2022	CSE,IT,	10/June	Online	100000	Hyderabad	2021-06-05	Click To
4Sam1	Samsung	Software Engineer Front end	8	8	2023	CSE,ECE,	15/June	Offline	120000	Hyderabad	2021-06-06	Click To
5Inf2	Infosys	Software backend	6	9	2023	CSE,IT,ECE,	8/June	Mixed	50000	Chennai	2021-06-06	Click To
6Ama3	Amazon	Manager	8	9	2022	IT,ECE,	8/November	Offline	70000	Delhi	2021-07-07	Click To
7Mic4	Microsoft	Backend Engineer	7	8	2023	CSE,IT,Mech,	7/October	Offline	67000	Chennai	2021-06-05	Click To

STUDENT SETTINGS



Welcome,
2019103027

Particular	Details
Student ID:	2019103027
First Name	Kavya
Last Name	Sridhar
Email id	kavya@gmail.com
DoB	2001-05-15
Phone Number	6437733776
Study	PG
College	CEG
Gender	female
Details	Update
Password	Change Password

COMPANY LOGIN

Company Login Page

Username

Password

Login

Not yet a member? [Sign up](#)

COMPANY REGISTRATION

Company Registration Page

Company Name

Register

Already a member? [Sign in](#)

COMPANY DASHBOARD

Placement Portal

Dashboard

Applications

Jobs

Settings

Sign Out

9
Total Students

3
Students applied

0
Students Placed

2
Total Jobs

Table

Job ID	Student ID	Name	Status
4Sam0	2019103006	Athirai	Not Placed
4Sam0	2019103015	Dhivya	Not Placed
4Sam0	2019103027	Kavya	Not Placed

View All

Jobs

Job Id	Vacancy Count
4Sam0	4
4Sam1	8

Welcome,
4

APPLICATIONS RECEIVED

Welcome,
4

VIEW STUDENT APPLICATIONS

Enter

Filter

Search

Reset

STUDENT ID	STUDENT NAME	COLLEGE	DEPARTMENT	CGPA	GRADUATION YEAR	UG/PG	JOB ID	DOB	GENDER	EMAIL ID	VIEW RESUME	APPL. STATUS
2019103006	Athirai	MIT	EEE	10	2023	UG	4Sam0	2001-11-30	female	athirai@gmail.com	Click here	Accept
2019103015	Dhivya	ACTech	Chemical Technology	10	2024	PG	4Sam0	2001-02-14	female	dhivya@gmail.com	Click here	Accept
2019103027	Kavya	CEG	CS	9.8	2022	PG	4Sam0	2001-05-15	female	kavya@gmail.com	Click here	Accept

JOB DETAILS FORM

Welcome,
4

Job Details

Company ID

4

Job ID

4Sam5

Job Description

Application Deadline

dd/mm/yyyy

Approximate Start Date

Enter Start Time

City

Enter City

State

Enter State

Mode

Online/Offline/Mixed

Salary

Enter approx salary

Vacancies

Enter Vacancies

COMPANY SETTINGS

Welcome,
4

Particular	Details
Company ID	4
Company Name	Samsung
EmailId	s@gmail.com
Contact Number	564546446
City	Hyderabad
Details	Update
Password	Change Password

ADMIN LOGIN

Admin Login Page

Admin User Id

Password

[Submit](#)

ADMIN DASHBOARD

Placement Portal

Dashboard

Students

Companies

Settings

Sign Out

9
Total Students

4
Total Companies

8
Students Active

1
Students Placed

Students

View All

Student ID	Name	Status
2019103000	Max	Not Placed
2019103006	Athirai	Not Placed
2019103009	bhargav	Placed
2019103015	Dhivya	Not Placed
2019103027	Kavya	Not Placed
2019103042	Neha	Not Placed
2019103054	Samhit	Not Placed
2019103060	Shrishti	Not Placed
2019103528	eeshwaran	Not Placed

List of Jobs

Job Id	Company Name	Vacancy Count
4Sam0	Samsung	4
4Sam1	Samsung	8
5Inf2	Infosys	6
6Ama3	Amazon	8
7Mic4	Microsoft	7

ADMIN VIEWS STUDENT DETAILS

VIEW STUDENT DETAILS

Enter

Filter Search Reset

Student Id	FIRST NAME	LAST NAME	COLLEGE	DEPARTMENT	CGPA	YEAR OF GRADUATION	UG/PG	DOB	GENDER	EMAIL ID	APPL STATUS	REMOVE
2019103000	Max	doe	CEG	CS	7.8	2023	PG	2000-06-06	other	v@gmail.com	n	Remove
2019103006	Athirai	S	MIT	EEE	10	2023	UG	2001-11-30	female	athirai@gmail.com	n	Remove
2019103015	Dhivya	Ramesh	ACTech	Chemical Technology	10	2024	PG	2001-02-14	female	dhivya@gmail.com	n	Remove
2019103027	Kavya	Sridhar	CEG	CS	9.8	2022	PG	2001-05-15	female	kavya@gmail.com	n	Remove
2019103042	Neha	Susan	MIT	CS	9.7	2023	PG	2001-12-18	female	neha@gmail.com	n	Remove
2019103054	Samhit	Mahadevan	MIT	ECE	7.7	2022	PG	2001-06-18	male	sammy@gmail.com	n	Remove
2019103060	Shrishti	Padmanabhan	CEG	CS	9	2023	UG	2001-06-10	female	shrishti@gmail.com	n	Remove
2019103528	eeshwaran	sriram	CEG	CS	8	2022	UG	2001-06-01	male	eesh@gmail.com	n	Remove

localhost/dbms project/Placement-Portal/adminlogin/dashboardpage/mainindex.php#

ADMIN VIEWS AND ADDS COMPANIES

Placement Portal

Dashboard

Students

Companies

Settings

Sign Out

Welcome,
Admin

View Companies

VIEW COMPANY DETAILS

Enter

Filter

Search

Reset

COMPANY ID	COMPANY NAME	EMAIL ID	PHONE NO.	CITY
4	Samsung	s@gmail.com	564546446	Hyderabad
5	Infosys	inf@gmail.com	2147483647	Chennai
6	Amazon	a@gmail.com	676767565	Delhi
7	Microsoft	mic@gmail.com	2147483647	Chennai

Add Companies

NOTIFICATION ID

COMPANY NAME

UNREAD

TIME

CREATE COMPANY

ADMIN SETTINGS

Welcome,
Admin

Change Password

Admin ID

admin

New Password

Current Password

Confirm Password

Submit

ALUMNI

Placement Portal

ALUMNI ID	FIRST NAME	LAST NAME	EMAIL ID	MOBILE NUMBER	UG/PG
2019103009	bhargav	ravi	baggy@gmail.com	8734646777	UG



Contributions

- Bhagavathi Ravikrishnan, 2019103009
 - Home Page (Front end)
 - Relational Schema
 - Student/Admin/Company Dashboard(Backend)
 - Student Placement Application -Applying for a job (Backend)
 - Student/Company Settings -Change Password(Backend + Front end)
 - Company Applications received - Accepting (Backend)
 - Admin view student details-remove a student (Backend +Trigger)
- Ishwarya Sriram, 2019103528
 - Student/Admin/Company page(Front end)
 - Student/Admin/Company Dashboard(Backend)
 - Student/Company Registration/Job details Registration (Frontend and Backend)
 - Implementing SQL Procedures - updating Admin Password and inserting details into Notifications table
 - Implementing SQL transaction for reading notifications and inserting company
- Kavya Sridhar, 2019103027
 - Created all tables
 - Student/Admin/Companylogin (Backend)
 - Admin's view of student and company details(Backend)
 - Student Placement Application-Viewing & Filtering(Backend)
 - Company Applications received - Viewing & Filtering(Backend)
 - Triggers for adding placed students into alumni tables and for decreasing no. of vacancies for the same.