DBMS - Mini Project
College Job Portal

Submitted By: Karthikeya R Jenni
SRN: PES1UG20CS200
V Semester Section D

TABLE OF CONTENTS

1). Short Description and Scope of the Project

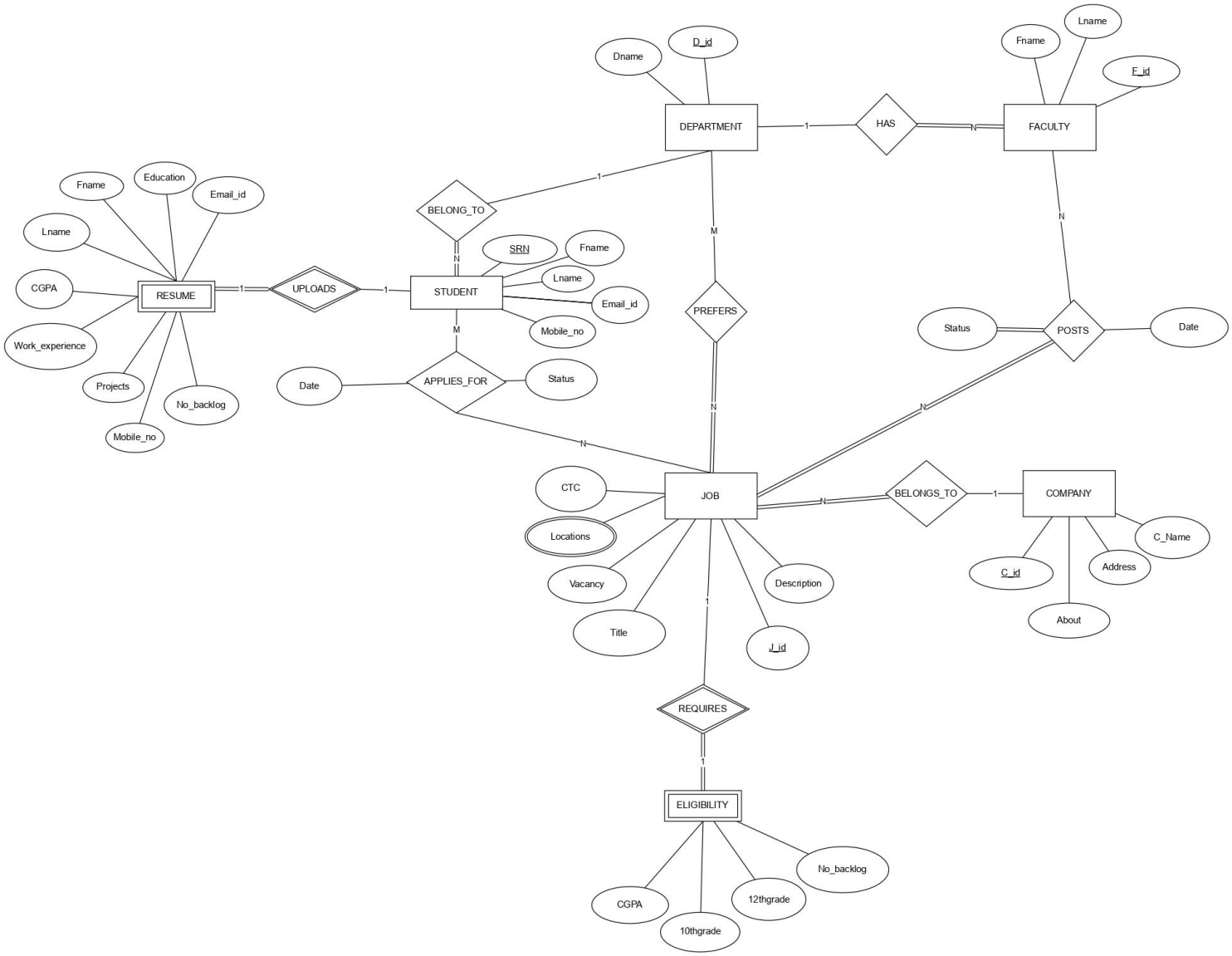
During Campus placement students have to keep track of hundreds of emails, application deadlines, and registration links. Students also find it difficult to fill in the same information in every google form sent by the college. It is also becoming increasingly tough for colleges to keep track of students applying for various jobs.

PESU Job Student's job portal is a one-stop solution for all these problems.

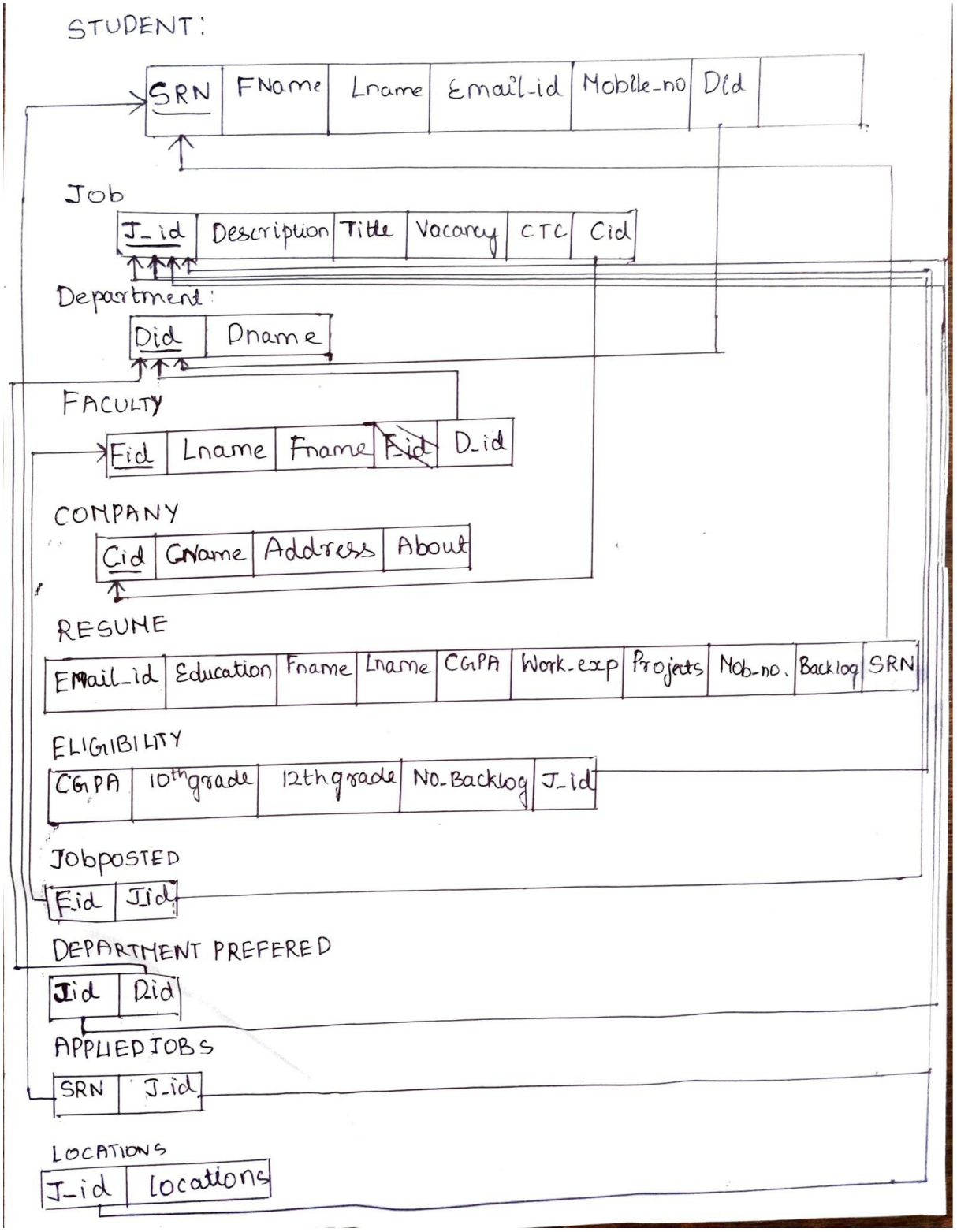
- College faculty and students can easily register themselves on this portal using their respective ids.
- Once a Company comes to campus for placements, college faculty can easily register that company on the portal and post the job that the company is offering on the portal.
- Students can view the different jobs available along with their eligibility criteria and apply for the job with a click of a button.
- No more filling out long google forms!
- Students can keep updating their profiles regularly as and when they complete new projects, certifications, etc.
- Companies can also easily view the resume of each applied student.

PESU Student’s Job Portal Front end:

2. ER Diagram:


DEPARTMENT
Dname
D_id
HAS
FACULTY
Fname
Lname
F_id
BELONG_TO
STUDENT
SRN
Fname
Lname
Email_id
Mobile_no
UPLOADS
RESUME
Fname
Education
Email_id
Lname
CGPA
Work_experience
Projects
No_backlog
Mobile_no
PREFERS
APPLIES_FOR
Date
Status
POSTS
Status
Date
JOB
CTC
Locations
Vacancy
Title
Description
J_id
BELONGS_TO
COMPANY
C_Name
C_id
Address
About
REQUIRES
ELIGIBILITY
No_backlog
CGPA
12thgrade
10thgrade

3. Relation Schema



4. DDL statements - Building the database

- **Applied_jobs table:**

```
CREATE TABLE applied_jobs (
  SRN char(13) ,
  jid int);
```

- **Company table:**

```
CREATE TABLE company (
  cid int NOT NULL,
  cname varchar(20) ,
  about text, PRIMARY KEY (cid)
);
```

- **Jobs table**

```
CREATE TABLE job (
  jid int NOT NULL,
  description text,
  title varchar(50) ,
  vacancy int,
  ctc int,
  cid int NOT NULL, PRIMARY KEY (jid), FOREIGN KEY (cid)
REFERENCES company (cid)
);
```

- **Department table**

```
CREATE TABLE department (
  did int NOT NULL,
  dname varchar(20), PRIMARY KEY (did));
```

- **Faculty table:**

```
CREATE TABLE faculty (
  fid int NOT NULL,
  fname varchar(20) ,
  lname varchar(20) ,
  did int(11) ,
  password varchar(30), PRIMARY KEY (fid), FOREIGN KEY
(did) REFERENCES department (did));
```

- **Eligibility table:**

```
CREATE TABLE eligibility (
  cgpa float,
  marks_10 float,
  marks_12 float,
  backlogs int,
  jid int NOT NULL,
);
```

- **Job_posted table**

```
CREATE TABLE job_posted (
  fid int(11) ,
```

```
  jid int(11), FOREIGN KEY (fid) REFERENCES faculty(fid),
FOREIGN KEY (jid) REFERENCES job (jid)
);
```

- **Student table**

```
CREATE TABLE student (
  SRN char(13) NOT NULL,
  fname varchar(20),
  lname varchar(20),
  email varchar(50),
  phone char(20),
  did int,
  password1 varchar(20), PRIMARY KEY (SRN), FOREIGN KEY
(did) REFERENCES department (did)
);
```

5. Populating the Database

1. Populating the student table:

Command:

```
cur.execute("INSERT INTO
student(did,email,fname,lname,password1,phone,SRN)
VALUES(%s,%s,%s,%s,%s,%s,%s)",(dno,email,fname,lname,password1,
phone,SRN)
```

2. Populating faculty table:

Command:

```
 cur.execute("INSERT INTO faculty ( did , fid , fname , lname, password)
VALUES(%s,%s,%s,%s,%s)",(did,fid,fname,lname,password))
```

3. Populating job table

Command:

```
cur.execute("INSERT INTO job(cid,ctc,description,jid,title,vacancy)
VALUES(%s,%s,%s,%s,%s,%s)",(cid,ctc,jdes,jid,jname,vacancy));
```

4. Populating eligibility table:

Command:

```
cur.execute("INSERT INTO
eligibility(backlogs,cgpa,jid,marks_10,marks_12)
VALUES(%s,%s,%s,%s,%s)",(backlogs,cgpa,jid,grade_10,grade_12))
```

5. Populating the student_resume table:

Code:

```
cur2.execute("INSERT INTO
student_resume(email,fname,lname,cgpa,work_exp,projects,phone,backlo
g,SRN,marks_10,marks_12)
VALUES(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)",(email,fname,lname,c
gpa,work_exp,project,phone,backlog,SRN,marks_10,marks_12))
```

6. Populating applied_jobs

Code:

```
cur.execute("INSERT INTO applied_jobs(SRN,jid)
VALUES(%s,%s)",(SRN,jid))
```

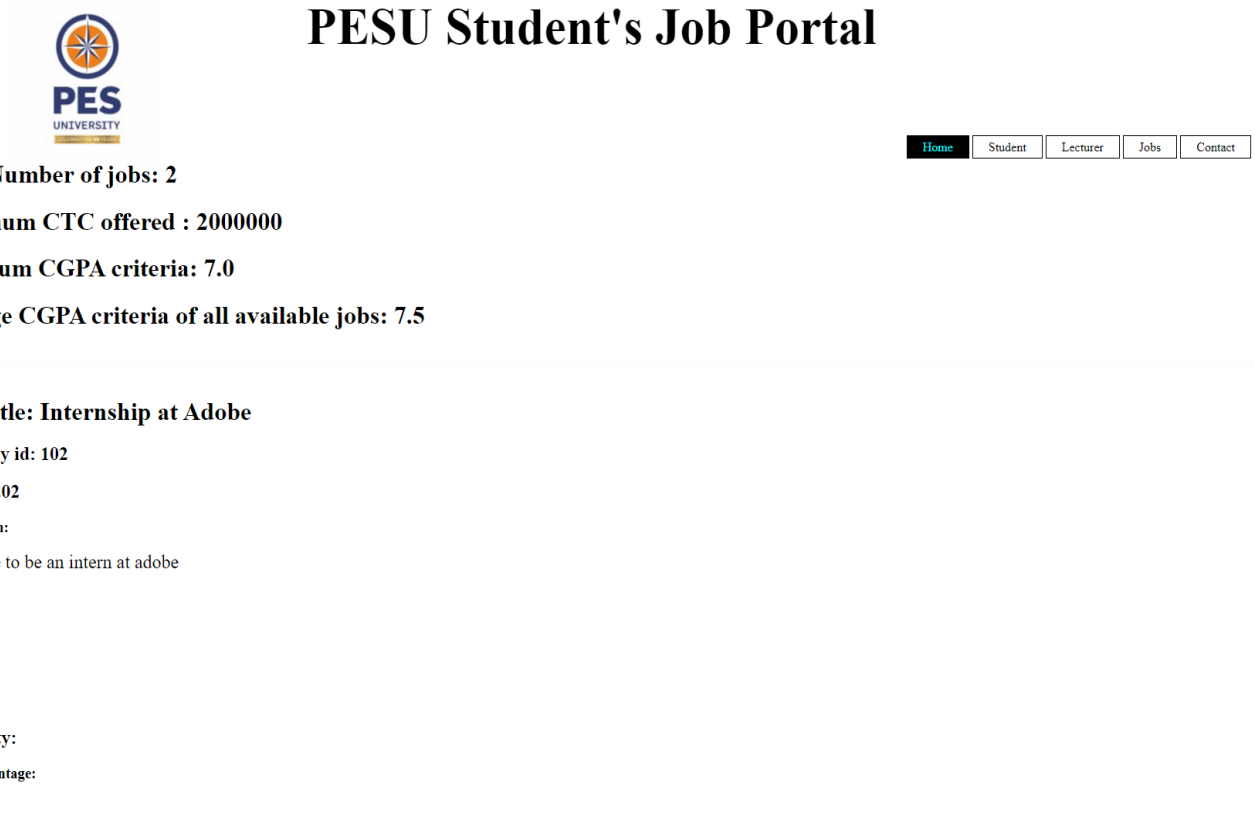
6) a. Join Queries

1. Display the jobs and their corresponding eligibility criterion.

Job information and their eligibility are in different tables, hence a join operation is done on these tables based on job id.

Command:

SELECT * FROM job JOIN eligibility WHERE job.jid = eligibility.jid;



Job Title: Internship at Adobe

Company id: 102

Job id: 202

Description:

A chance to be an intern at adobe

CTC:

1500000

Vacancy:

3

Eligibility:

10th Percentage:

90.0

12th Percentage:

90.0

CGPA:

7.0

Backlogs:

0

2. Display the information of every registered company and the jobs they are offering.

SQL Command:

SELECT * FROM company JOIN job WHERE jid = jid;

Output:

+ Options

cid	cname	about	jid	description	title	vacancy	ctc	cid
101	Cisco	A very famous networking company	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
101	Cisco	A very famous networking company	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102
102	Adobe	A multinational comapny	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
102	Adobe	A multinational comapny	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102
103	Microsoft	Microsoft Corporation is an American multinational...	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
103	Microsoft	Microsoft Corporation is an American multinational...	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102
104	Google	Google LLC is an American multinational technology...	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
104	Google	Google LLC is an American multinational technology...	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102
300	GE	New	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
300	GE	New	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102

3. Display the information of all students along with their resumes.

SQL command:

SELECT * FROM student JOIN student_resume WHERE student.SRN = student_resume.SRN;

Output:

SRN	fname	lname	email	phone	did	password1	email	fname	lname	cgpa	work_exp	projects	phone	backlog	SRN	marks_10	marks_12
PES1UG20CS200	Karthikeya	Jenni	karthikeyajenni@gmail.com	8088743562	1	*NULL*	karthikeyajenni@gmail.com	Karthikeya	Jenni	9.5	Software Intern at Microsoft	Accident response system Seminar hall booking sys...	8088743562	0	PES1UG20CS200	*NULL*	*NULL*
PES1UG20CS202	Karthikeya	R Jenni	karthikeyajenni@gmail.com	8088372235	1	123	karthikeyajenni@gmail.com	Karthikeya	R Jenni	9.8	Mane kelsa	-	8088372235	0	PES1UG20CS202	67	76
PES1UG20CS500	Karthikeya	R Jenni	karthikeyajenni@gmail.com	8088372235	1	123	karthikeyajenni@gmail.com	Karthikeya	R Jenni	9.54	Intern at google	Student job portal	8088372235	0	PES1UG20CS500	96	97

4. Display the details of every job posted along with the details of the faculty who posted it.

SQL command:

```
SELECT * FROM faculty AS f JOIN job AS j   WHERE
EXISTS (SELECT * FROM job_posted WHERE
job_posted.fid = f.fid AND job_posted.jid = j.jid);
```

Output:

+ Options

fid	fname	lname	did	password	jid	description	title	vacancy	ctc	cid
1002	Rama	P	2	*NULL*	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
1001	Teja	B	1	*NULL*	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102

6). b. Aggregate Functions

1). Display the total number of jobs posted.

Command:

SELECT COUNT(jid) INTO total FROM job;

2). Display the maximum ctc offered from the list of posted jobs.

Command:

SELECT MAX(ctc) FROM job;

3). Display the minimum cgpa requirement among the list of posted jobs.

Command:
SELECT MIN(cgpa) FROM eligibility;

4). Display the average cgpa requirement among the posted jobs.

Command:

SELECT AVG(cgpa) FROM eligibility;

Result of the commands mentioned above:

Total Number of jobs: 2

Maximum CTC offered : 2000000

Minimum CGPA criteria: 7.0

Average CGPA criteria of all available jobs: 7.5

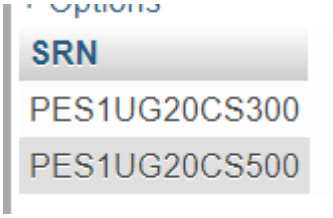
6). c. SET OPERATIONS

1). Display the SRN of all students who have not applied for any job.

SQL Command:

```
SELECT SRN FROM student
EXCEPT
SELECT SRN FROM applied_jobs;
```

Output:



SRN
PES1UG20CS300
PES1UG20CS500

2). Display the fid of all faculty who have not posted any job.

SQL Command:

```
SELECT fid FROM faculty
EXCEPT
SELECT fid FROM job_posted;
```

Output:

fid
1234
5678

3. Display the information all students along with their resume if they have one.

SQL Command:

SELECT * FROM student RIGHT JOIN student_resume ON student_resume.SRN = student.SRN;

Output:

SRN	fname	lname	email	phone	did	password1	email	fname	lname	cgpa	work_exp	projects	phone	backlog	SRN	marks_10	marks_12
PES1UG20CS200	Karthikeya	Jenni	karthikeyajenni@gmail.com	8088743562	1	*NULL*	karthikeyajenni@gmail.com	Karthikeya	Jenni	9.5	Software Intern at Microsoft	Accident response system Seminar hall booking sys...	8088743562	0	PES1UG20CS200	*NULL*	*NULL*
PES1UG20CS202	Karthikeya	R Jenni	karthikeyajenni@gmail.com	8088372235	1	123	karthikeyajenni@gmail.com	Karthikeya	R Jenni	9.8	Mane kelsa	-	8088372235	0	PES1UG20CS202	67	76
PES1UG20CS500	Karthikeya	R Jenni	karthikeyajenni@gmail.com	8088372235	1	123	karthikeyajenni@gmail.com	Karthikeya	R Jenni	9.54	Intern at google	Student job portal	8088372235	0	PES1UG20CS500	96	97

3. Display the details of every job posted along with the details of the faculty who posted it.

SQL command:

SELECT * FROM faculty AS f JOIN job AS j WHERE EXISTS (SELECT * FROM job_posted WHERE job_posted.fid = f.fid AND job_posted.jid = j.jid);

Output:

+ Options

fid	fname	lname	did	password	jid	description	title	vacancy	ctc	cid
1002	Rama	P	2	*NULL*	200	Are you curious, collaborative, and love to inspir...	SDE at Microsoft	2	2000000	103
1001	Teja	B	1	*NULL*	202	A chance to be an intern at adobe	Internship at Adobe	3	1500000	102

6). d. Functions and procedure:

1). Create a function that inputs the job id of a job and deletes the corresponding job if it is in the job table and returns 1. If a job with that id is not present, it should return 0.

Command:

```
DELIMITER $$
CREATE FUNCTION delete_jobs (new_jid INTEGER)
RETURNS INTEGER
DETERMINISTIC
BEGIN
    DECLARE done INT DEFAULT 0;
    DECLARE cur CURSOR FOR SELECT * FROM job
WHERE jid = new_jid;
  DECLARE CONTINUE HANDLER FOR NOT FOUND
SET done = 1;
  OPEN cur;
  IF done = 1 THEN
    RETURN 0;
  ELSE
```

```
        DELETE FROM job WHERE jid = new_jid;
    END IF;
    CLOSE cur;
    RETURN 1;

END $$
DELIMITER ;
```

2). Write a procedure that counts the total number of jobs posted and returns the count.

Command:

```
DELIMITER $$
CREATE PROCEDURE count_jobs (OUT total INT)
BEGIN
        SELECT COUNT(jid) INTO total FROM job;
   END $$
DELIMITER ;
```

6). e. Triggers and cursors

1). Create a trigger to delete the eligibility of a job with id jid when a delete operation has been called on the job with id jid.

Command:

```
DELIMITER $$
CREATE TRIGGER before_delete_job
   BEFORE DELETE
       ON job FOR EACH ROW
       BEGIN
      DELETE FROM eligibility WHERE jid = OLD.jid;
   END $$
DELIMITER ;
```

2). Create a trigger to keep the SRN of the student unchanged after the student updates his resume, even if the student has changed his SRN.

```
DELIMITER $$
CREATE TRIGGER after_update_profile
   AFTER UPDATE
       ON student_resume FOR EACH ROW
       BEGIN
       UPDATE student_resume SET SRN = OLD.SRN
WHERE SRN = NEW.SRN ;
       END $$
DELIMITER ;
```

3. Create a trigger to delete a student's resume if his cgpa is less than 6. (As the student is not eligible for placements)

```
DELIMITER $$
CREATE TRIGGER before_insert
    AFTER INSERT
    ON student_resume FOR EACH ROW
        BEGIN
        IF NEW.cgpa <6 THEN
        DELETE FROM student_resume WHERE SRN =
NEW.SRN;
        END IF;

    END $$
DELIMITER ;
```

4. Create a trigger to delete a student's resume if the student has more than 5 backlogs. (As the student is not eligible for placements)

```
DELIMITER $$
CREATE TRIGGER before_insert_backlog
   AFTER INSERT
   ON student_resume FOR EACH ROW
       BEGIN
       IF NEW.backlog > 5 THEN
       DELETE FROM student_resume WHERE SRN =
NEW.SRN;
       END IF;

   END $$
DELIMITER ;
```

Cursors

1). Cursor to fetch all entries from the job table.

```
cur = mysql.connection.cursor()

cur.execute("SELECT * FROM job")

details1 = cur.fetchall()

mysql.connection.commit()

cur.close()
```

2). Cursor to fetch all entries from applied_jobs table.

```
cur = mysql.connection.cursor()

cur.execute("SELECT * FROM applied_jobs")

details1 = cur.fetchall()
```

3). Cursor to insert values into student table.

```
 cur = mysql.connection.cursor()
cur.execute("INSERT INTO
student(did,email,fname,lname,password1,phone,SRN)
VALUES(%s,%s,%s,%s,%s,%s,%s)",(dno,email,fname,lname
,password1,phone,SRN))
mysql.connection.commit()
cur.close()
```

4). Cursor to join job and eligibility table.

```
cur = mysql.connection.cursor()
cur.execute("SELECT * FROM job JOIN eligibility WHERE
job.jid = eligibility.jid")
details1 = cur.fetchall()
```

5). Cursor to update student table.

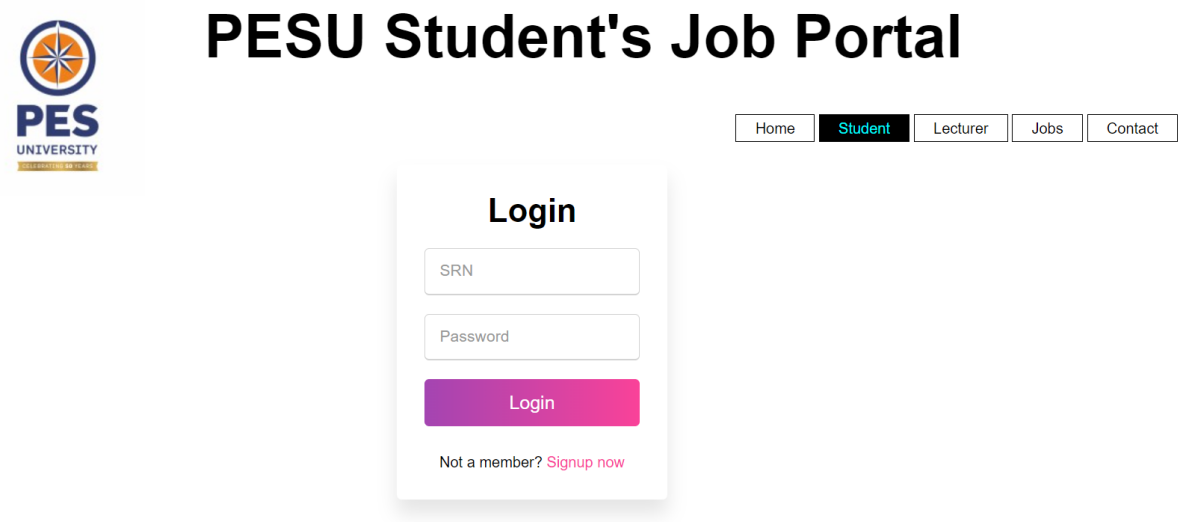
```
cur3 = mysql.connection.cursor()
cur3.execute("UPDATE student SET fname =
%s,lname=%s,email=%s,phone=%s,did=%s WHERE SRN =
%s",(fname,lname,email,phone,dno,SRN))
mysql.connection.commit()
```

6). f. Higher Level Programming - A simple frontend

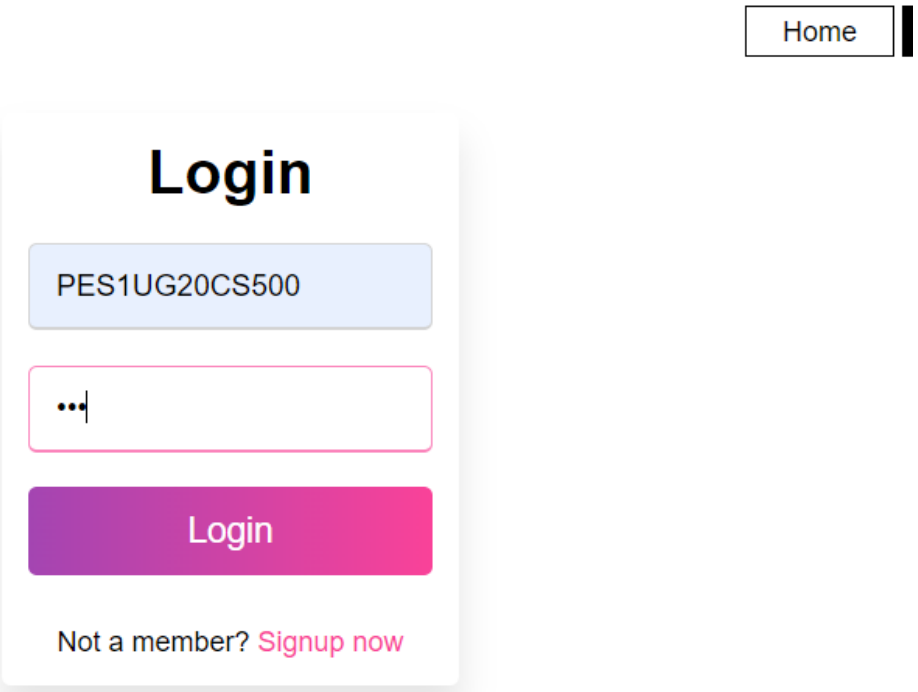
1). Home page

2). Student Login Page and Faculty login page

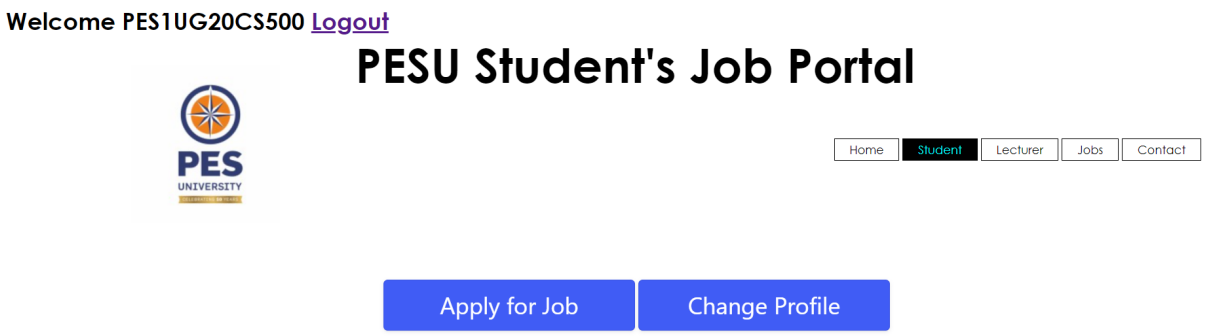
Read operation being performed



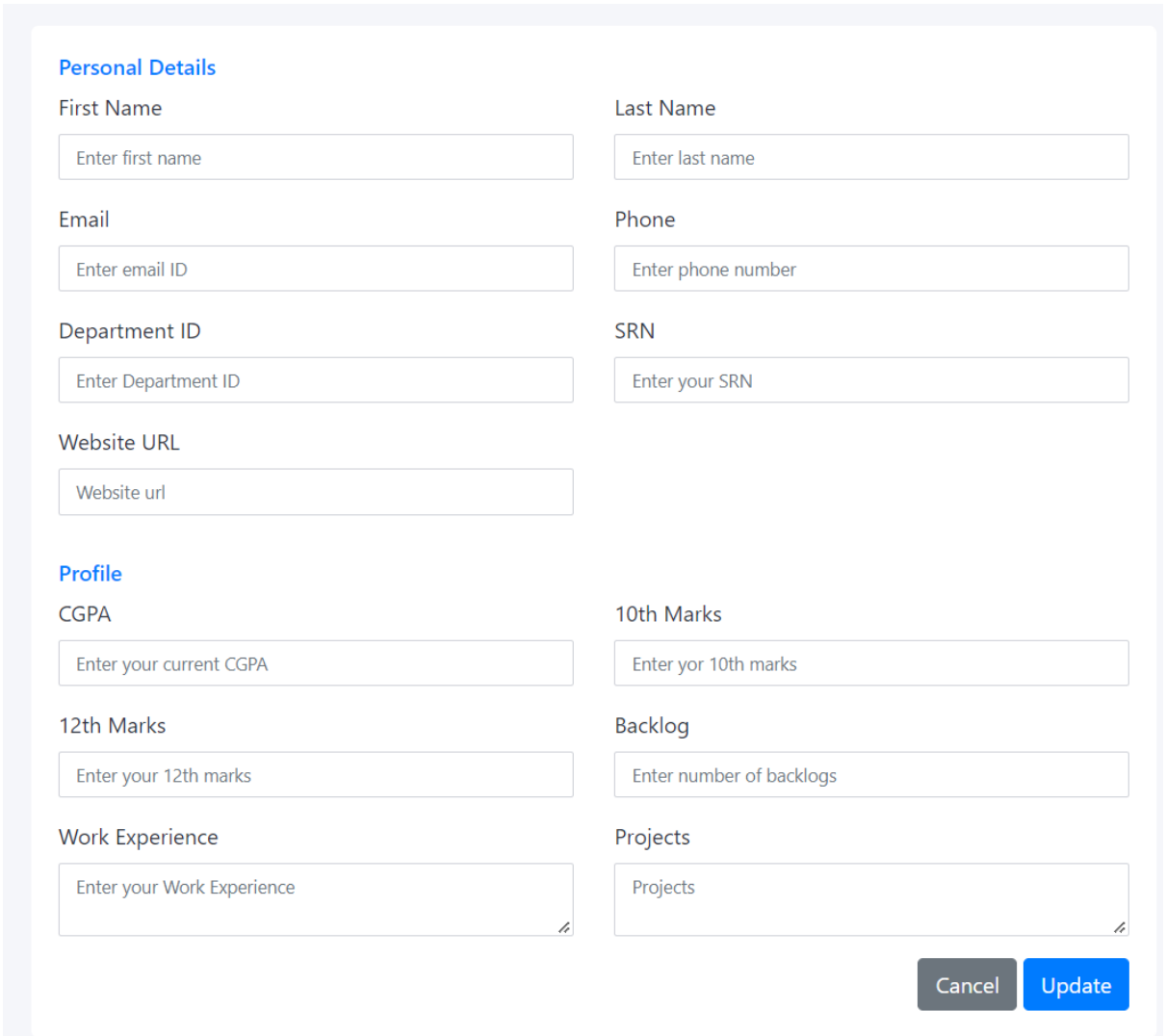
Enter details:



Login Successful:



4). Update student profile: Update operation



Personal Details

First Name: Karthikeya

Last Name: R Jenni

Email: karthikeyajenni@gmail.com

Phone: 8088372235

Department ID: 1

SRN: PES1UG20CS500

Website URL: https://www.linkedin.com/feed/

Profile

CGPA: 9.54

10th Marks: 96

12th Marks: 97

Backlog: 0

Work Experience: Intern at google

Projects: Student job portal

Cancel | Update

Welcome Logout

PESU Student's Job Portal

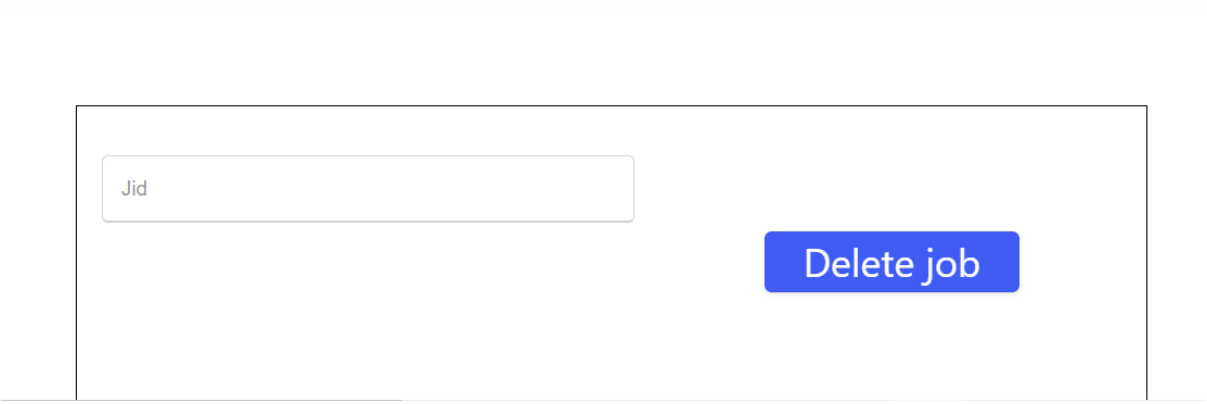
Home | Student | Lecturer | Jobs | Contact

Profile updated successfully

Apply for Job | Change Profile

3). Delete a job:

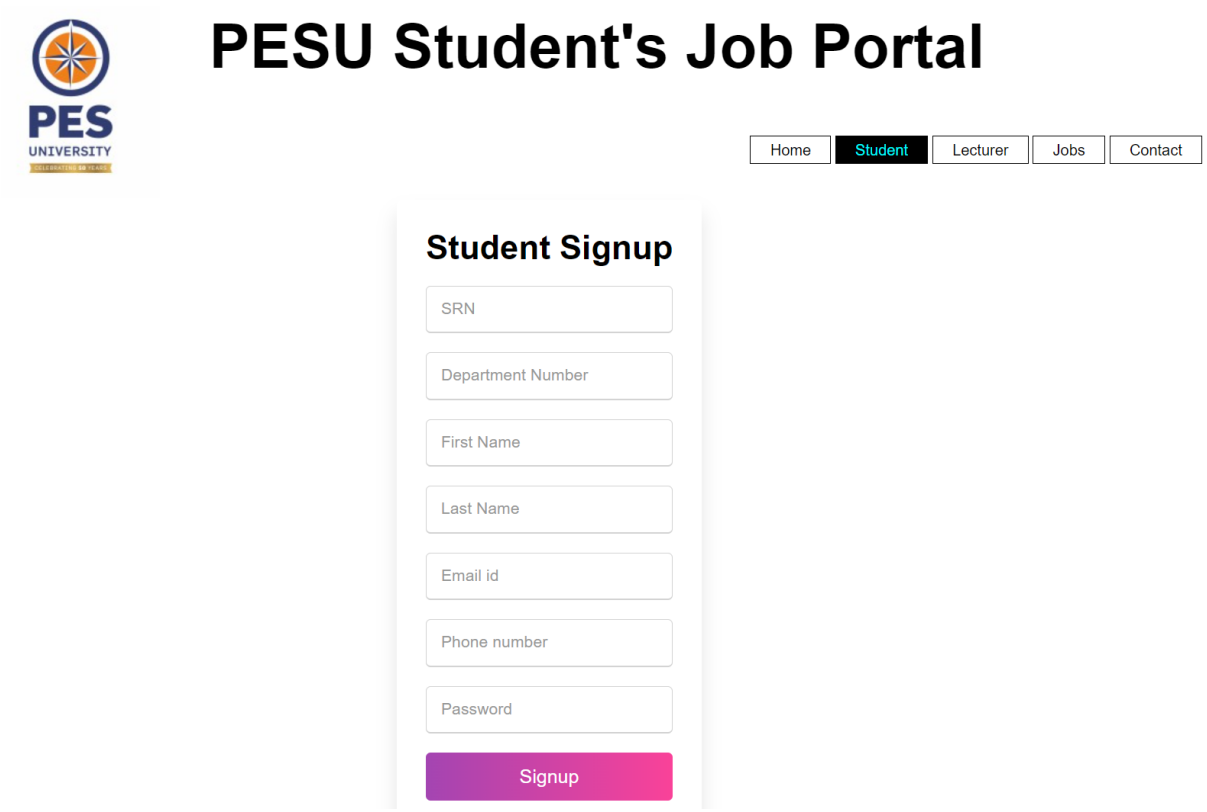
Delete operation



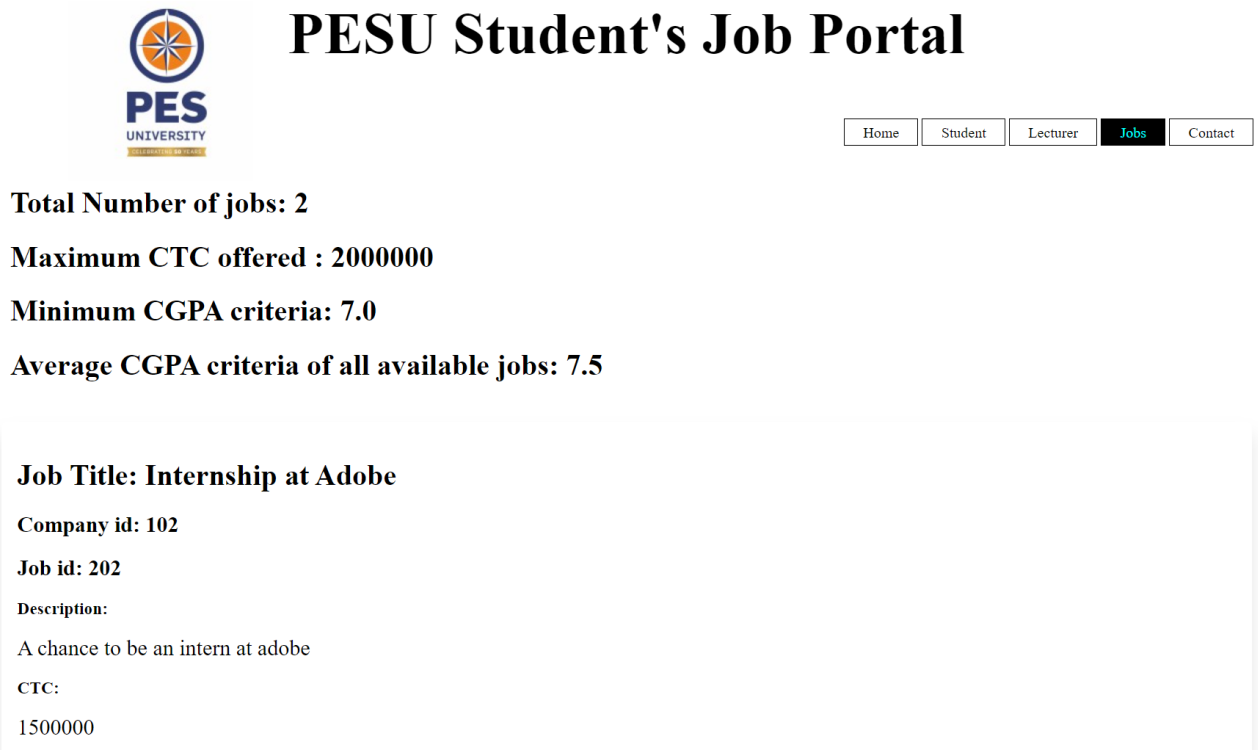
Enter job id:

Deletion successful:

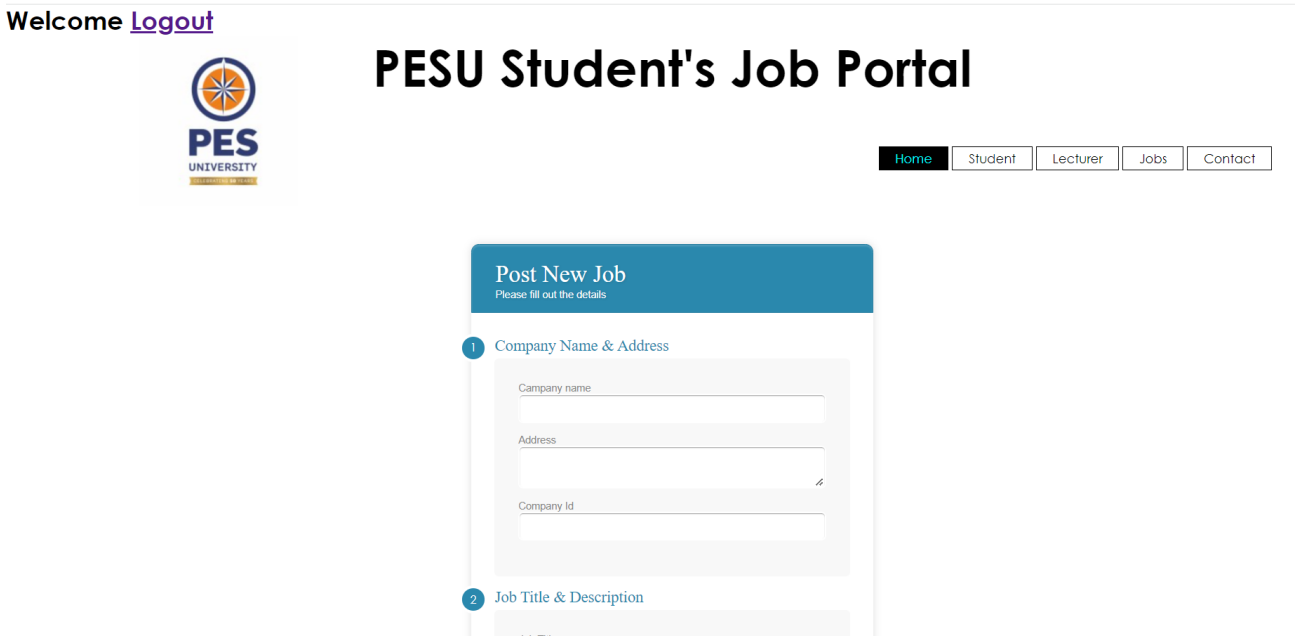
3). Student Registration Page


PESU Student's Job Portal
Home
Student
Lecturer
Jobs
Contact
Student Signup
SRN
Department Number
First Name
Last Name
Email id
Phone number
Password
Signup

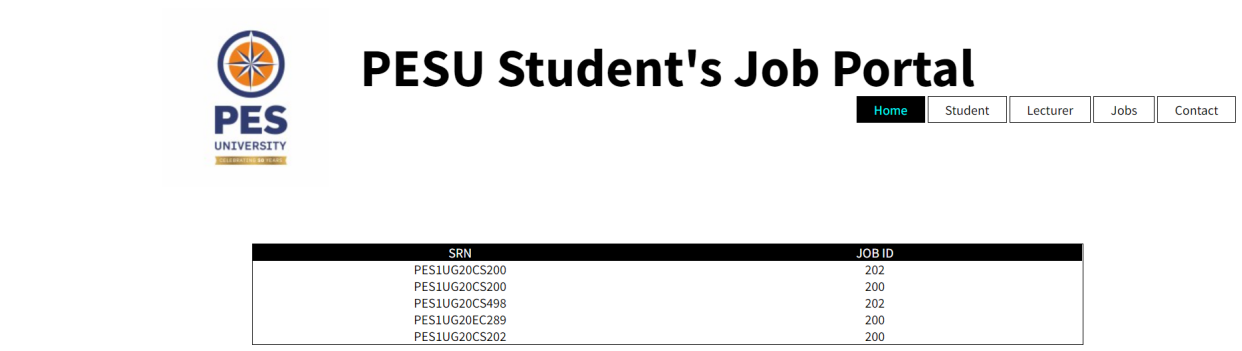
4). Jobs Page


PESU Student's Job Portal
Home
Student
Lecturer
Jobs
Contact
Total Number of jobs: 2
Maximum CTC offered : 2000000
Minimum CGPA criteria: 7.0
Average CGPA criteria of all available jobs: 7.5
Job Title: Internship at Adobe
Company id: 102
Job id: 202
Description:
A chance to be an intern at adobe
CTC:
1500000

5). Faculty can post a new job on this page:



6). Faculty can view the applied jobs:



SRN	JOB ID
PES1UG20CS200	202
PES1UG20CS200	200
PES1UG20CS498	202
PES1UG20EC289	200
PES1UG20CS202	200