Project 2

Yutong Liu N16773429 & SiyunWang N10409344

# Introduction

Here, we built a database system which is used in the Job Hunting website. When you open the website, you could choose either a Student interface or a company interface. The website is a bridge between them which helps students looking for a great employer and helps companies hiring great employees. There are some useful and interesting features could be expected from this website:

As a student, they could:

1. Sign Up their own account, and upload all the information to the website as well as their resume. The information is including their name, university, major, degree, GPA, their interest, and a resume in a pdf format. Also, they could decide whether they want all their information be viewed by all the people and companies or not. In addition, students could edit their profile as long as they want.
2. Students are welcomed to send friend requests to other students. And if other students agree with their request, they would become friends and could talk to each other, share some job and company information. All the friends request, chats, job information forwarded by their friends would be found in “Friend Notifications”.
3. Students could find a company they like by inputting the keyword of the company, they could follow the company, apply jobs posted by that company, and forward the job to their friends. They could also receive the job notifications from their followed company.
4. Besides, students can search, apply, forward jobs by using keywords.

As a recruiter from company they could:

1. Sign Up their own account by email, and upload all the information to the website, which includes their name, headquarter location, industry. They could log in to their account by either their Company name or their email.
2. They could edit new opening job information, which includes job title, location, estimate salary, desired degree, desired major, and a short description. After that, they could post their new jobs on the website so that students could find the jobs. Also, the students who satisfy the requirements of the job would be notified with this job information.

# ER Diagram

Based on these main functions, we started to design the database system and achieve it in frontend in the second project. The ER diagram is shown as below:

## Schema with Explanations and Sample queries

**Schema:**

**Student** (sid, sname, spassword)

Sid is login name.

**StudentInfo** (sid, sresumeaddr, suniversity, sdegree, smajor, sgpa, sinfo, srestriction)

Foreign Key: sid refers Student.sid

**ResumeInfo** (sid, sresumeaddr, resumecontent)

Foreign Key: (sid, sresumeaddr) refers StudentInfo.(sid, sresumeaddr)

**CompanySign** (cid, cemail, cpassword)

**Company** (cid, cname, ccity, cstate, cindustry)

Foreign Key: cid refers CompanySign.cid

**JobInfo** (jid, cid, jtitle, jcity, jstate, jsalary, jdegree, jmajor, jpostdate, jdesc)

Foreign Key: cid refers Company.cid

**Friends** (sid, fid)

Foreign Key: sid, fid refers Students.sid

**Follow** (sid, cid)

Foreign Key: sid refers Students.sid; cid refers Companys.cid

**JobApply** (sid, jid, cid)

Foreign Key: sid refers Students.sid; jid refers JobInfo.jid; cid refers Company.cid

**JobNotifications** (sid, jid, cid, notifytime, viewStatus)

Foreign Key: sid refers Students.sid; cid refers Companys.cid; jid refers JobInfo.jid

viewStatus = (‘Viewed’,’New’)

**FriendMessage** (sid, fid, message, mdate, jidmes)

Foreign Key: (sid, fid) refers (Friends.sid, fid);

**FriendRequest** (sid, rid, requesttime, frstatus)

Foreign Key: sid, rid refers Student.sid; Status = (‘Pending’,’Agreed’,’Rejected’)

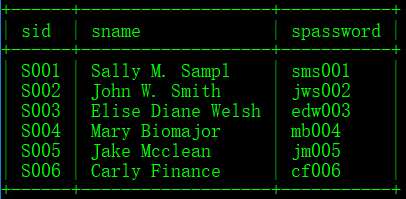
**Explanations:**

Student is for user log in. sname is the real name and sid is login name, here login name is unique. So sid is primary key for Student. After sign in, student need to fill their information, which stores in StudentInfo table. In the StudentInfo, we only store the resume address not the content. Based on this, we create a new table to store the resume content, called ResumeInfo. CompanySign is used for company sign up. And also, the information of company will be filled in Company Table. JobInfo is a list of job information, including the unique id jid. Friends table represents the relationship between two sid is friends. Follow table will record the status if the student follows the company. JobApply is for recording the apply status and JobNotifications is for store the information which student will be notificated. FriendRequest is stored the action of friend request.

**Tables and Sample Queries:**

### 1.

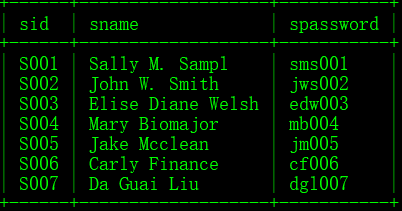
**Original Table**: (sid means loginname)



**SQL Query:**

INSERT INTO `Student` VALUES('S007','Da Guai Liu','dgl007');

**Updated Table:**



### 2.

For example, for the student which loginname = ‘S001’, list the name of all friends of him.

**SQL Query:**

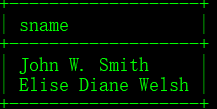
SELECT Student.sname

FROM Friends, Student

WHERE Student.sid = Friends.fid

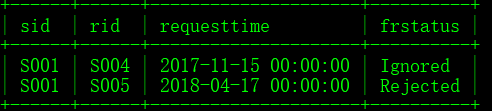
AND Friends.sid = "S001"

**Result:**



### 3.

**Original FriendRequest:**



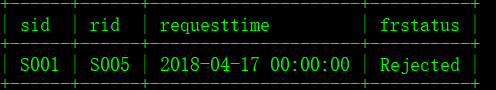
**SQL Query:**

DELETE FROM FriendRequest

WHERE TIMESTAMPDIFF(MONTH, requesttime, CURDATE()) > 1

AND frstatus != 'Agreed'

**After Delete Operation:**



### 4.

**SQL Query:**

SELECT StudentInfo.sid

FROM Follow, StudentInfo, Company

WHERE Follow.sid = StudentInfo.sid AND Follow.cid = Company.cid

AND StudentInfo.suniversity = 'New York University'

AND Company.cname = 'Microsoft'

AND Follow.followstatus = 'Followed'

**Result:**

Output the unique loginname



### 5.

**SQL Query:**

SELECT jid

FROM JobInfo

WHERE TIMESTAMPDIFF(DAY, jpostdate, CURDATE()) >= 7

AND TIMESTAMPDIFF(DAY, jpostdate, CURDATE()) < 14

AND jdegree = 'MS' AND jmajor = 'Computer Science'

**Result:**



### 6.

**First, find the student which satisfy the condition**

**SQL Query:**

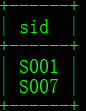
SELECT StudentInfo.sid

FROM StudentInfo, ResumeInfo

WHERE StudentInfo.sid = ResumeInfo.sid AND StudentInfo.sresumeaddr = ResumeInfo.sresumeaddr

AND sgpa > '3.5' AND resumecontent like '%database%' collate utf8\_general\_ci

**Result:**



**Second, create a notification, here for the company C05 and job J009**

**SQL Query in PHP:**

$sqlNew = "SELECT Student.sid

FROM StudentInfo, ResumeInfo, Student

WHERE StudentInfo.sid = ResumeInfo.sid AND StudentInfo.sresumeaddr = ResumeInfo.sresumeaddr AND Student.sid = StudentInfo.sid

AND sgpa > '3.5' AND resumecontent like '%database%' collate utf8\_general\_ci";

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

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()){

$sid = $row["sid"];

echo "$sid";

// insert new data to the JobNotifications Table

mysqli\_query($conn,"INSERT INTO JobNotifications (`sid`, `jid`, `cid`) VALUES ('$sid', 'J009','C05')")

or die(mysqli\_error($conn));}}

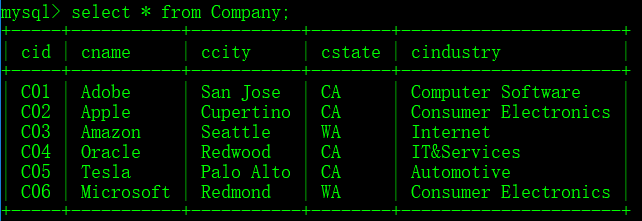
**Result:**



## (d)

**Sample Data We design:**

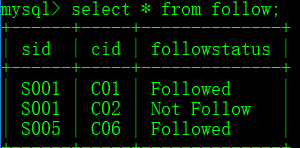
Company:



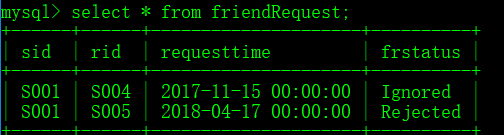
CompanySign:



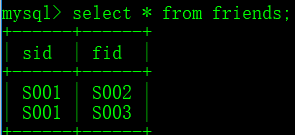
Follow:



FriendRequest:

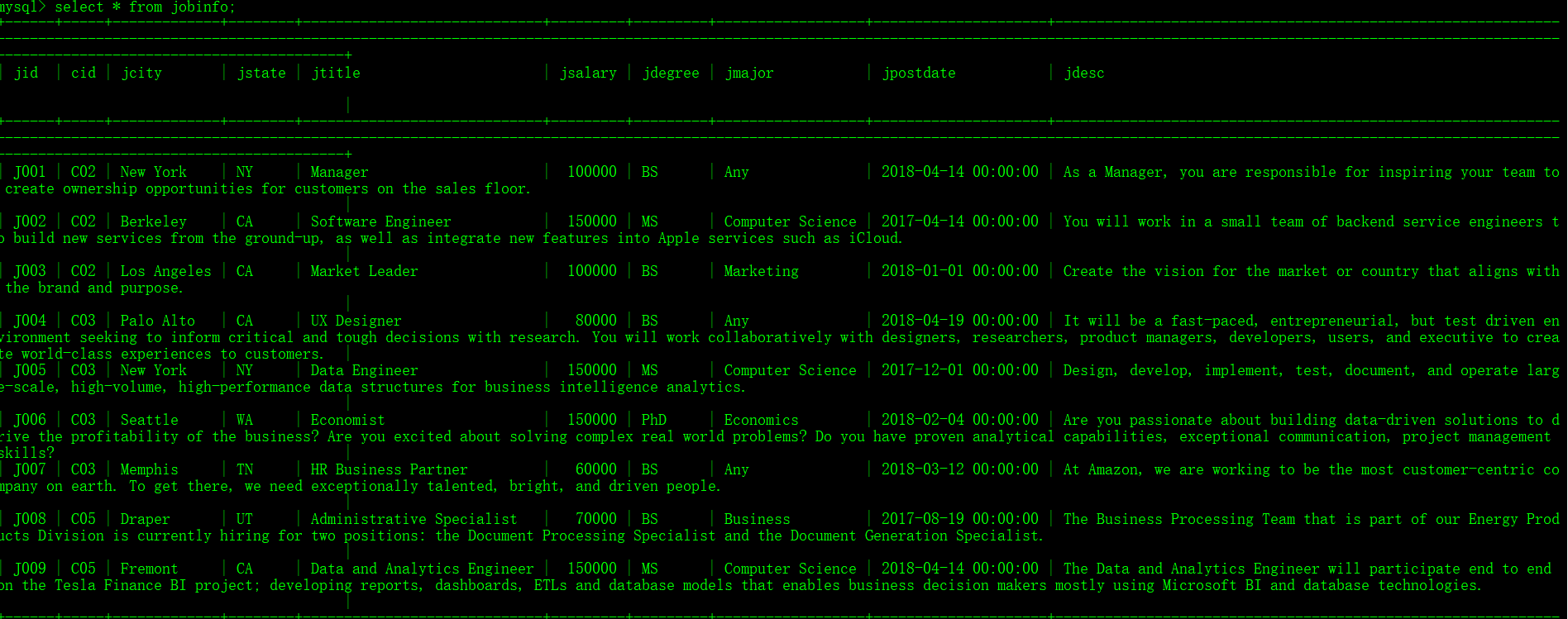


Friends

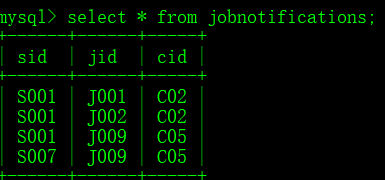


JobApply

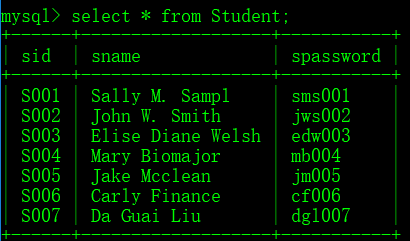


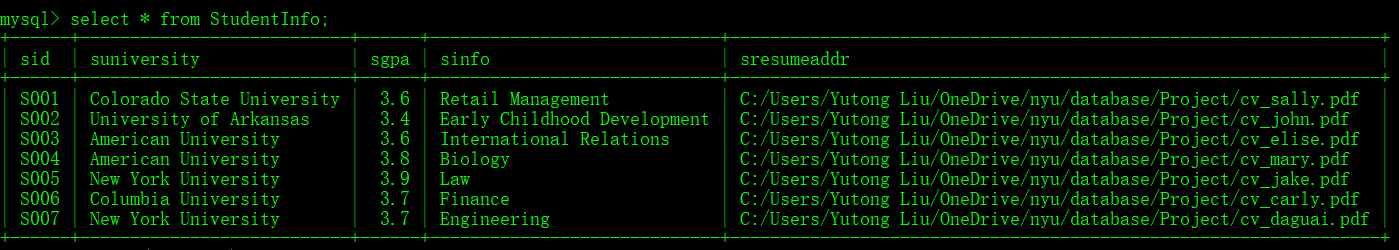
JobInfo

JobNotifications



Student



StudentInfo.

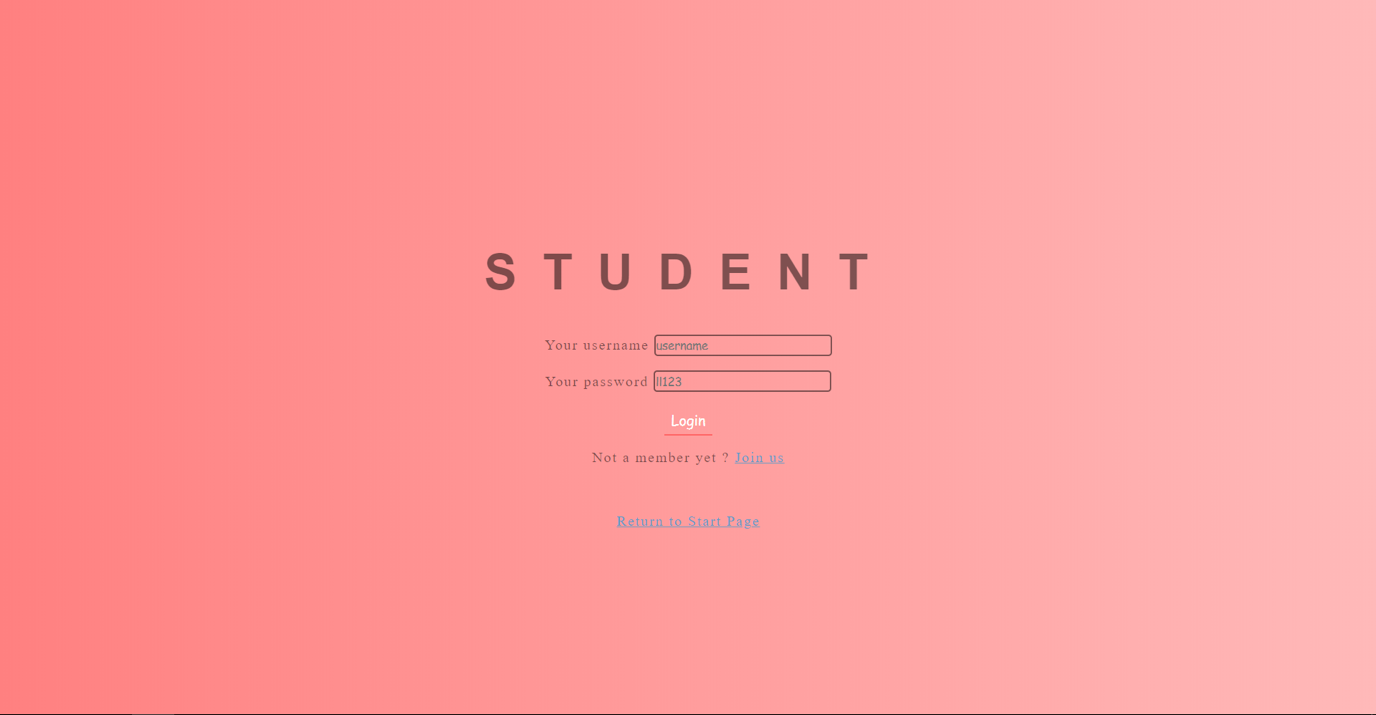
To sum up, these are data we have for now. We will add them in the second project. Also, the sql file is in the attachment, which called jonhunter.sql.

## Implementation:

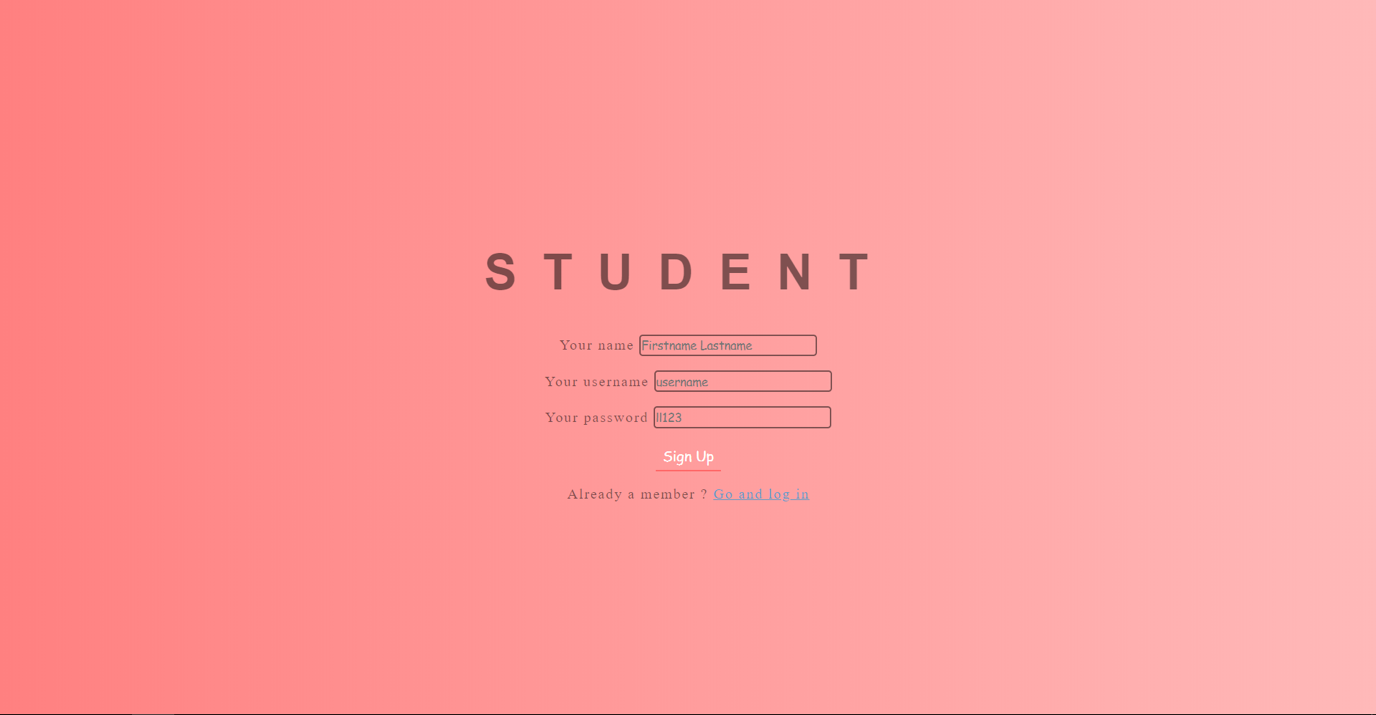
Users can choose they are a student or a company, they would be linked to different pages.



----------Student----------



**Step 1:** A new student can sign up a new account, the “login” name (sid), name, and password would be stored in “Student” table.



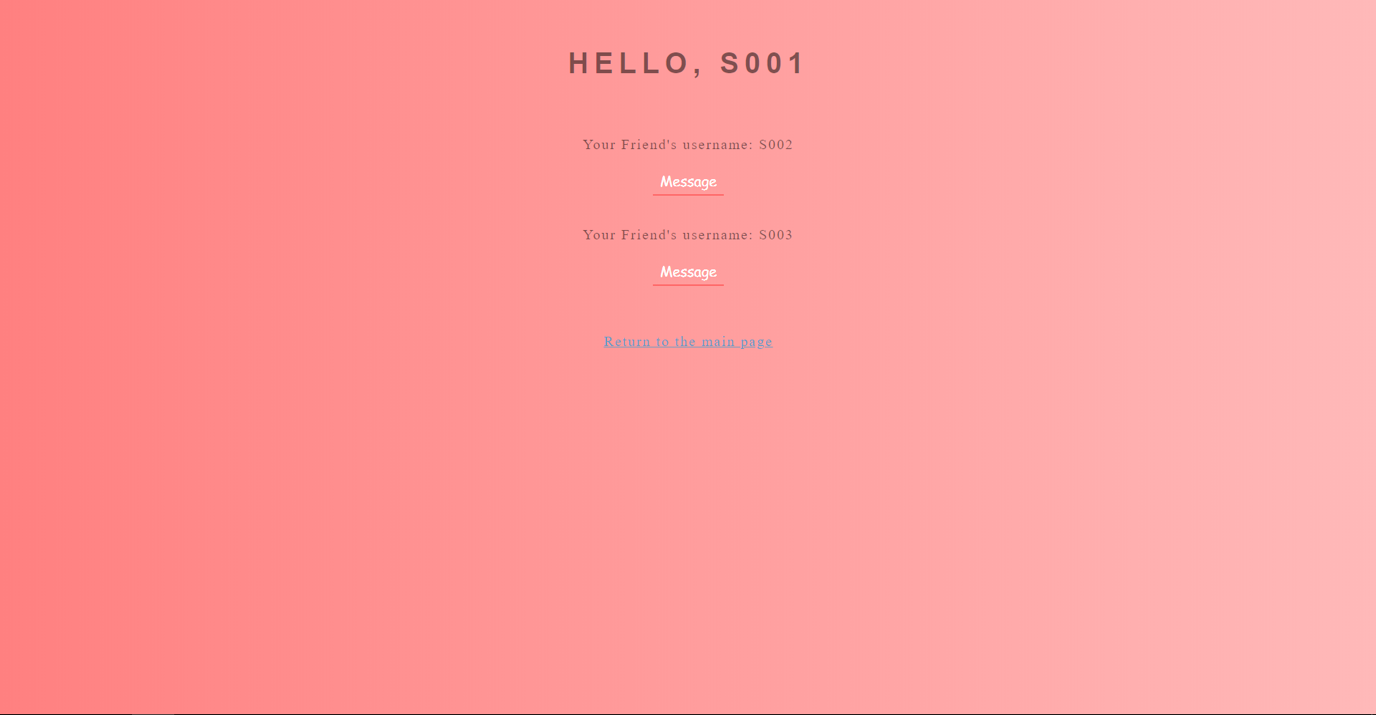
Student can also edit their profile via “View your profile” button, and link their resume. They can decide whether everyone can see their resume or only friends and company which they applied job can see their resume. The profile information includes sid, suniversity, sdegree, smajor, sgpa, sinfo (like interest), srestriction and sresumeaddr. The resume pdf file whould be stored in “ResumeInfo” table, which includes sid, sresumeaddr and resumecontent.



**Step 2:** Student can search from friend search function, and they can send friend request to other users, sid, rid (whom you request to be friend), requesttime, frstatus would be stored in “FriendRequest” table.



**Step 3:** If two students become friend, they can send message to each other and share some job information. There’s an button called ‘Talk to your friends’, which achieve this function. “Message” table need to store sid, fid, mdate, message and jidmes.



**Step 4:** Student can receive notifications when someone want to make friends with him. And he can choose Agree or Rejected. There’s an button called ‘Friend Notifications’, which achieve this function.



**Step 5:** Student can search the keyword of the Company, follow the Company they are interested in, the sid of the student and the cid of the company would be store in “Follow” table.



When you clicking the “Posted Jobs”, you can browse the jobs posted by the company.

You can apply the job, sid, jid of the job, cid would be stored in “JobApply” table.

You can forward the job to your friend, sid, fid of your friend, mdate (the date you forward the job to your friend), jidmes (this job id) would be stored in “FriendMessage” table. Your friend will see a “smile face” when there is a new notification.

**Step 6:** Students can search jobs directly by using keyword (for example “e”) in Job Search. In the next interface, you can apply the job and the information would be stored in “JobApply” table like the last step.



You can forward the job to your friend, sid, fid of your friend, mdate (the date you forward the job to your friend), jidmes (this job id) would be stored in “FriendMessage” table. Your friend will see a “smile face” when there is a new notification.

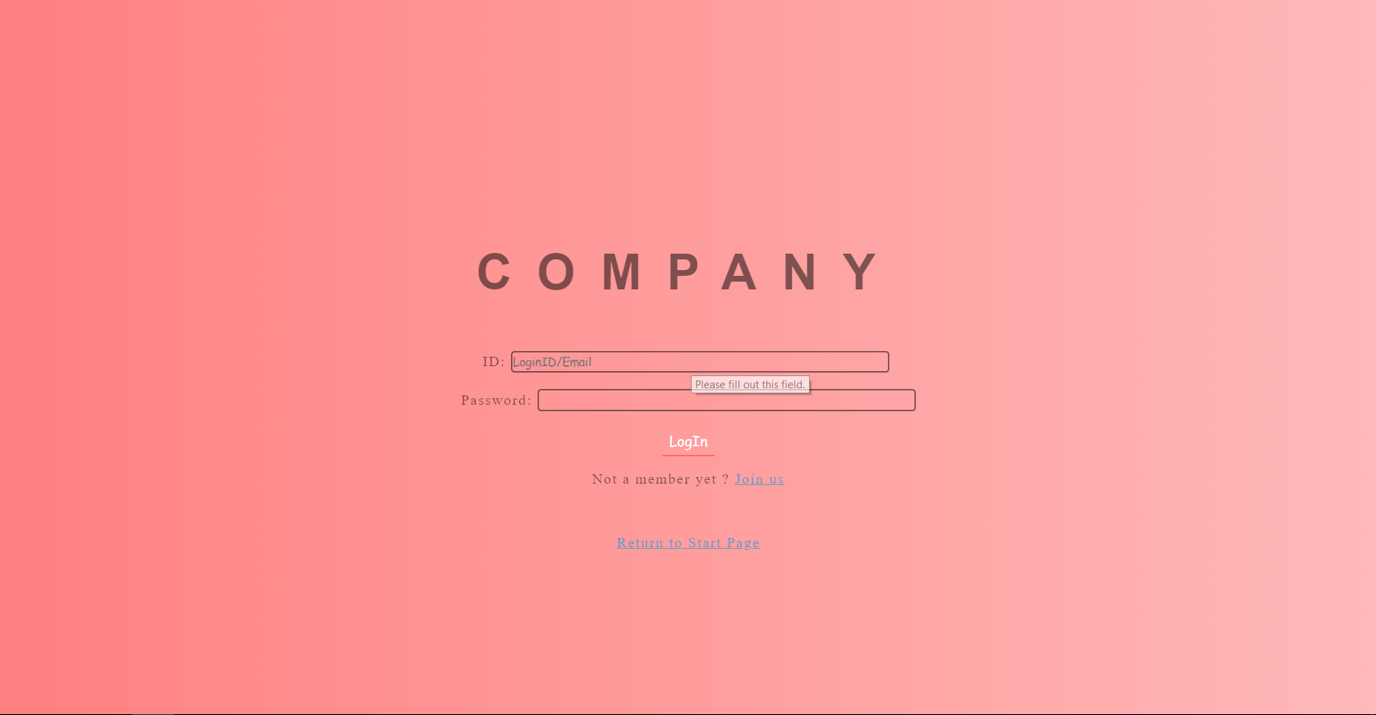
**Step 7:** For students, when their followed company post a new job or there is a matching job, they would see a “New Jobs” next to “JobNotification” button. After clicking the button, they would see a list of notified jobs, including viewed or new jobs, and they can apply or forward these jobs to their friends. When they view the job and return to the main page, the ViewStatus in the “JobNotifications” table would be update from “New” to “Viewed”.



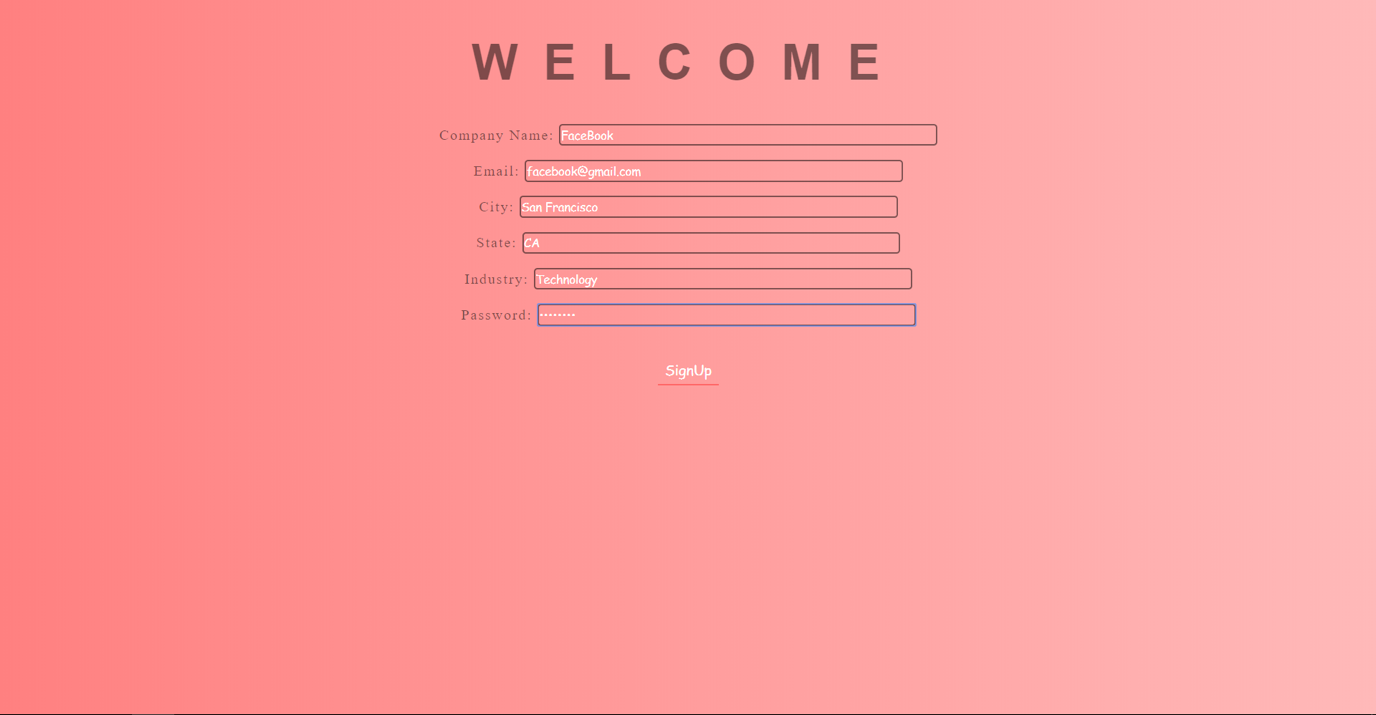
---------------------------

---------Extra Points-------

## ----------Company---------



**Step 1:** In Company\_SignUp.php, you could file out the form as the image below shows. When clicking the “Sign Up”, the information would be stored in ‘CompanySign’ table and ‘Company’ table. Clicking Return could return to sign in interface (Company\_SignIn.php).



**Step 2:** After inputting the company name/ email and password, you could enter to JobPosting.php interface. You could find your company’s general information and posted jobs on this page as below.



**Step 3:** When clicking “Add more jobs”, you could edit new job in JobEdit.php page, the new job would be stored in JobInfo. In this page, there is a query should be run. The query is designed for finding the potential students who may fit this job by matching the degree, major, keyword, or finding the students who followed this company. Then, some information would be stored into “JobNotifications” table. The information includes sid (student id), jid (job id), cid (company id), NotifyDate, ViewStatus (Set it to “New” since this is a new notification.).

