Project 1

Yutong Liu N16773429 & SiyunWang N10409344

Introduction

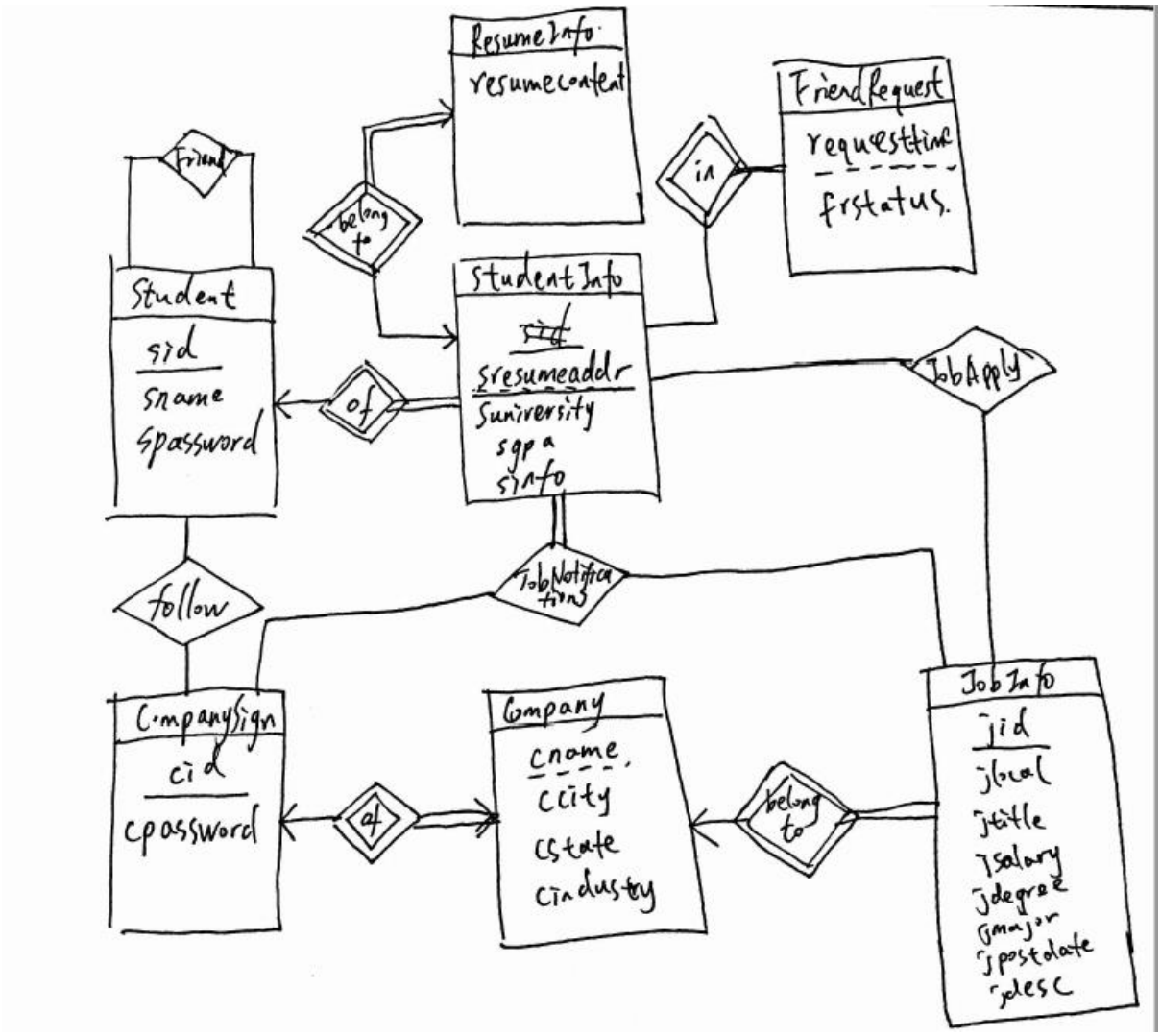
Here, we want to build a database system which is used in the Job Hunting website. There are two user group: Student and Company. The website is a bridge between them which helps student find the great employee and helps company find the great employers. There are some main functions we want to achieve:

1. Sign Up and Sign In for Student and Company. Different Sign Up methods for Different Groups.
2. Allow Student and Company submit their information and allow the information can be searched. For Students, information including their name, university, major, degree, and a resume in a pdf format. For Company, information including their name, address, industry, and Job information they posted.
3. Student can send friend request to the other students. And if the other one agreed, they become friends and can share some job or company information.
4. Student can follow their favorite company and receive the job notifications from their followed company.
5. Student can apply the job they like.
6. Company can post their job announcement to the specific student group even that they are not followed by these students.
7. For the Student and Company, they both can search some information by keywords.

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

(a)

ER diagram


ResumeInfo
resumecontent
FriendRequest
requesttime
frstatus
Friend
belong to
in
Student
sid
sname
spassword
StudentInfo
sresumeaddr
suniversity
sgpa
sinfo
of
JobApply
follow
JobNotification
CompanySign
cid
cpassword
Company
cname
ccity
cstate
cindustry
belong to
JobInfo
jid
jlocal
jtitle
jsalary
jdegree
jmajor
jpostdate
jdesc

(b)

Student (sid, sname, spassword)
Sid is login name.
StudentInfo (sid, sresumeaddr, suniversity, sdegree, smajor, sgpa, sinfo)
Foreign Key: sid refers Student.sid
ResumeInfo (sid, sresumeaddr, resumecontent)
Foreign Key: (sid, sresumeaddr) refers StudentInfo.(sid, sresumeaddr)
CompanySign (cid, cpassword)
Company (cid, cname, ccity, cstate, cindustry)
Foreign Key: cid refers CompanySign.cid
JobInfo (jid, cid, jlocal, jtitle, jsalary, jdegree, jmajor, jpostdate, jdesc)
Friends (sid, fid)
Foreign Key: sid, fid refers Students.sid
Follow (sid, cid, followStatus)
Foreign Key: sid refers Students.sid; cid refers Companys.cid
JobApply (sid, jid, applyStatus)
Foreign Key: sid refers Students.sid; jid refers JobInfo.jid
JobNotifications (sid, jid, cid)
Foreign Key: sid refers Students.sid; cid refers Companys.cid; jid refers JobInfo.jid
FriendRequest (sid, rid, requesttime, frstatus)
Foreign Key: sid, rid refers Student.sid; Status = ('Ignored','Agreed','Rejected')

Explanation:
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 follow 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.

(c)

In the third question, we write the query and use sample data test them. Below is our SQL query and Result in MySQL.

1.

Original Table: (sid means loginname)

sid	sname	spassword
S001	Sally M. Sampl	sms001
S002	John W. Smith	jws002
S003	Elise Diane Welsh	edw003
S004	Mary Biomajor	mb004
S005	Jake Mcclean	jm005
S006	Carly Finance	cf006

SQL Query:

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

Updated Table:

sid	sname	spassword
S001	Sally M. Sampl	sms001
S002	John W. Smith	jws002
S003	Elise Diane Welsh	edw003
S004	Mary Biomajor	mb004
S005	Jake Mcclean	jm005
S006	Carly Finance	cf006
S007	Da Guai Liu	dgl007

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:

sname
John W. Smith
Elise Diane Welsh

3.

Original FriendRequest:

sid	rid	requesttime	frstatus
S001	S004	2017-11-15 00:00:00	Ignored
S001	S005	2018-04-17 00:00:00	Rejected

SQL Query:

```
DELETE FROM FriendRequest
WHERE TIMESTAMPDIFF(MONTH, requesttime, CURDATE()) > 1
AND frstatus != 'Agreed'
```

After Delete Operation:

sid	rid	requesttime	frstatus
S001	S005	2018-04-17 00:00:00	Rejected

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

sid
S005

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:

sid	jid	cid
S001	J001	C02
S001	J002	C02
S001	J009	C05
S007	J009	C05

(d)

Sample Data We design:

Company:

```
mysql> select * from Company;
+-----+-----------+-----------+--------+---------------------+
| cid | cname     | ccity     | cstate | cindustry           |
+-----+-----------+-----------+--------+---------------------+
| C01 | Adobe     | San Jose  | CA     | Computer Software   |
| C02 | Apple     | Cupertino | CA     | Consumer Electronics|
| C03 | Amazon    | Seattle   | WA     | Internet            |
| C04 | Oracle    | Redwood   | CA     | IT&Services         |
| C05 | Tesla     | Palo Alto | CA     | Automotive          |
| C06 | Microsoft | Redmond   | WA     | Consumer Electronics|
+-----+-----------+-----------+--------+---------------------+
```

CompanySign:

```
mysql> select * from CompanySign;
+-----+-----------+
| cid | cpassword |
+-----+-----------+
| C01 | adobe     |
| C02 | apple     |
| C03 | amazon    |
| C04 | oracle    |
| C05 | tesla     |
| C06 | microsoft |
```

Follow:

```
mysql> select * from follow;
+------+-----+--------------+
| sid  | cid | followstatus |
+------+-----+--------------+
| S001 | C01 | Followed     |
| S001 | C02 | Not Follow   |
| S005 | C06 | Followed     |
```

FriendRequest:

```
mysql> select * from friendRequest;
```

sid	rid	requesttime	frstatus
S001	S004	2017-11-15 00:00:00	Ignored
S001	S005	2018-04-17 00:00:00	Rejected

Friends

```
mysql> select * from friends;
```

sid	fid
S001	S002
S001	S003

JobApply

sid	jid	applystatus
S001	J001	Applied
S001	J002	Not Apply

JobInfo

```
mysql> select * from jobinfo;
```

jid	cid	jcity	jstate	jtitle	jsalary	jdegree	jmajor	jpostdate	jdesc
J001	C02	New York	NY	Manager	100000	BS	Any	2018-04-14 00:00:00	As a Manager, you are responsible for inspiring your team to create ownership opportunities for customers on the sales floor.
J002	C02	Berkeley	CA	Software Engineer	150000	MS	Computer Science	2017-04-14 00:00:00	You will work in a small team of backend service engineers t o build new services from the ground-up, as well as integrate new features into Apple services such as iCloud.
J003	C02	Los Angeles	CA	Market Leader	100000	BS	Marketing	2018-01-01 00:00:00	Create the vision for the market or country that aligns with the brand and purpose.
J004	C03	Palo Alto	CA	UX Designer	80000	BS	Any	2018-04-19 00:00:00	It will be a fast-paced, entrepreneurial, but test driven en vironment seeking to inform critical and tough decisions with research. You will work collaboratively with designers, researchers, product managers, developers, users, and executive to crea te world-class experiences to customers.
J005	C03	New York	NY	Data Engineer	150000	MS	Computer Science	2017-12-01 00:00:00	Design, develop, implement, test, document, and operate larg e-scale, high-volume, high-performance data structures for business intelligence analytics.
J006	C03	Seattle	WA	Economist	150000	PhD	Economics	2018-02-04 00:00:00	Are you passionate about building data-driven solutions to d rive the profitability of the business? Are you excited about solving complex real world problems? Do you have proven analytical capabilities, exceptional communication, project management skills?
J007	C03	Memphis	TN	HR Business Partner	60000	BS	Any	2018-03-12 00:00:00	At Amazon, we are working to be the most customer-centric co mpany on earth. To get there, we need exceptionally talented, bright, and driven people.
J008	C05	Draper	UT	Administrative Specialist	70000	BS	Business	2017-08-19 00:00:00	The Business Processing Team that is part of our Energy Prod ucts Division is currently hiring for two positions: the Document Processing Specialist and the Document Generation Specialist.
J009	C05	Fremont	CA	Data and Analytics Engineer	150000	MS	Computer Science	2018-04-14 00:00:00	The Data and Analytics Engineer will participate end to end on the Tesla Finance BI project: developing reports, dashboards, ETLs and database models that enables business decision makers mostly using Microsoft BI and database technologies.

JobNotifications

```
mysql> select * from jobnotifications;
```

sid	jid	cid
S001	J001	C02
S001	J002	C02
S001	J009	C05
S007	J009	C05

Student

mysql> select * from Student;

sid	sname	spassword
S001	Sally M. Sampl	sms001
S002	John W. Smith	jws002
S003	Elise Diane Welsh	edw003
S004	Mary Biomajor	mb004
S005	Jake Mcclean	jm005
S006	Carly Finance	cf006
S007	Da Guai Liu	dgl007

StudentInfo

mysql> select * from StudentInfo;

sid	suniversity	sgpa	sinfo	sresumeaddr
S001	Colorado State University	3.6	Retail Management	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_sally.pdf
S002	University of Arkansas	3.4	Early Childhood Development	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_john.pdf
S003	American University	3.6	International Relations	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_elise.pdf
S004	American University	3.8	Biology	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_mary.pdf
S005	New York University	3.9	Law	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_jake.pdf
S006	Columbia University	3.7	Finance	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_carly.pdf
S007	New York University	3.7	Engineering	C:/Users/Yutong Liu/OneDrive/nyu/database/Project/cv_daguai.pdf

.

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.