

Online Employment Platform

1 The stakeholders and requirements

There are three types of stakeholders in this system: student, company (employer) and administrator.

The platform is designed specifically for the employers and employees so the operations of CVs and jobs are necessary (S). To measure the degree of attraction of the jobs the companies must browse the information of the applicants (M). We have all of the knowledge which is needed to complete this project so it is achievable (A). To link with the real employment cases, the relationships of entities are designed correctly (R) and these requirements are all done as planned (T). The requirements for each type of users are expanded below.

For students, they can upload their CVs and apply jobs offered by companies. After uploading them, they can apply jobs by the CVs. If they want, they can download and view their CVs or delete them. They can see which jobs are being applied using each CV and they can also browse for available jobs that have not been applied by them yet.

In term of companies, they can list their vacancies of jobs on the platform so that students can apply the jobs. Similar to the student, companied can edit and delete their jobs. Additional, they can view the applicants' CVs.

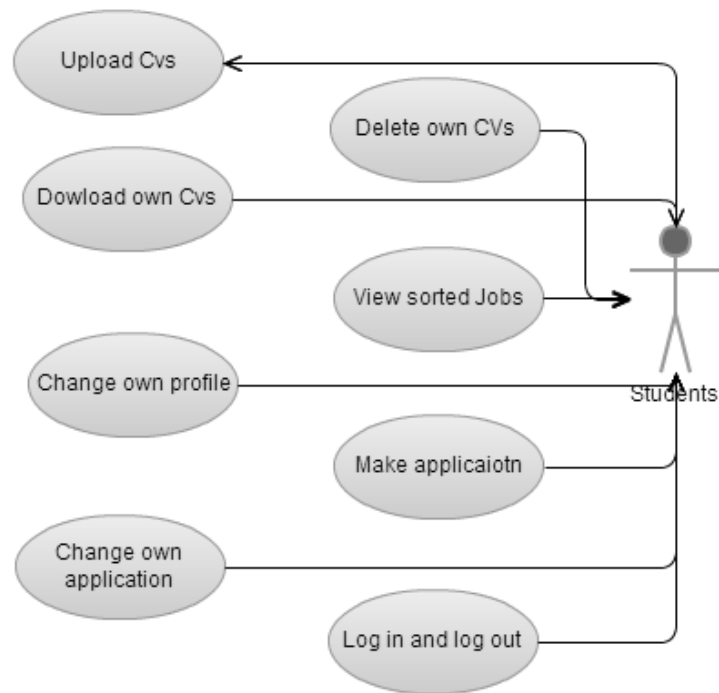
Both of these two types of users can leave their names and contact numbers.

The last type of users is administrator obviously. They can manage the data of students and companies, including disable and delete it. All of the operations are implemented according to the logic of the relationships of classes.

2 Use-case diagrams

Students' use case diagrams:

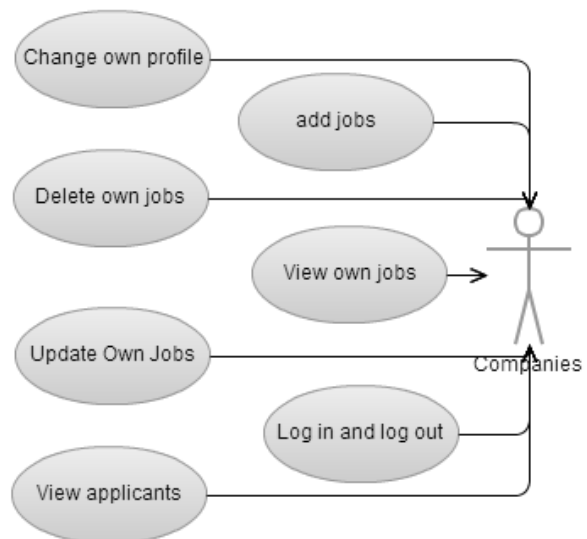
Use Case



As a Student, you can upload your CVs and download or delete them, change your own profile, including name, email and telephone number. Also, you can make applications by your CVs and change them. Obviously, the two functions -- log-in and log-out are necessary.

Companies' use case diagram

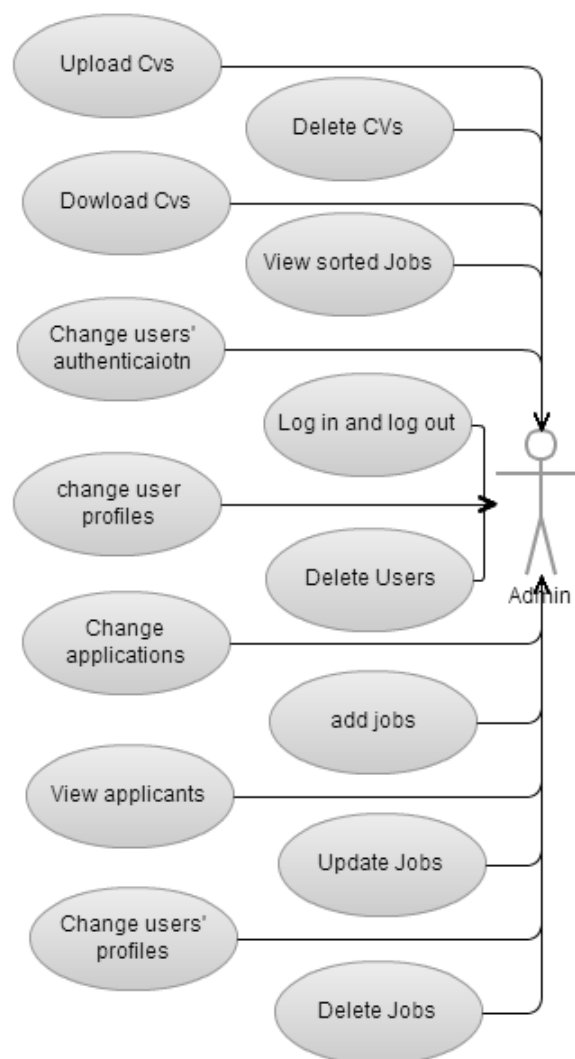
Use Case



As a company, you can add jobs. You can also view, update and delete the jobs added by you. Furthermore you can change your profile, including your company's name, address telephone number and contact name. After adding jobs you can wait and view the applicants' information. Of course, you must log in firstly and you can log out.

Administrator's use diagram

Use Case

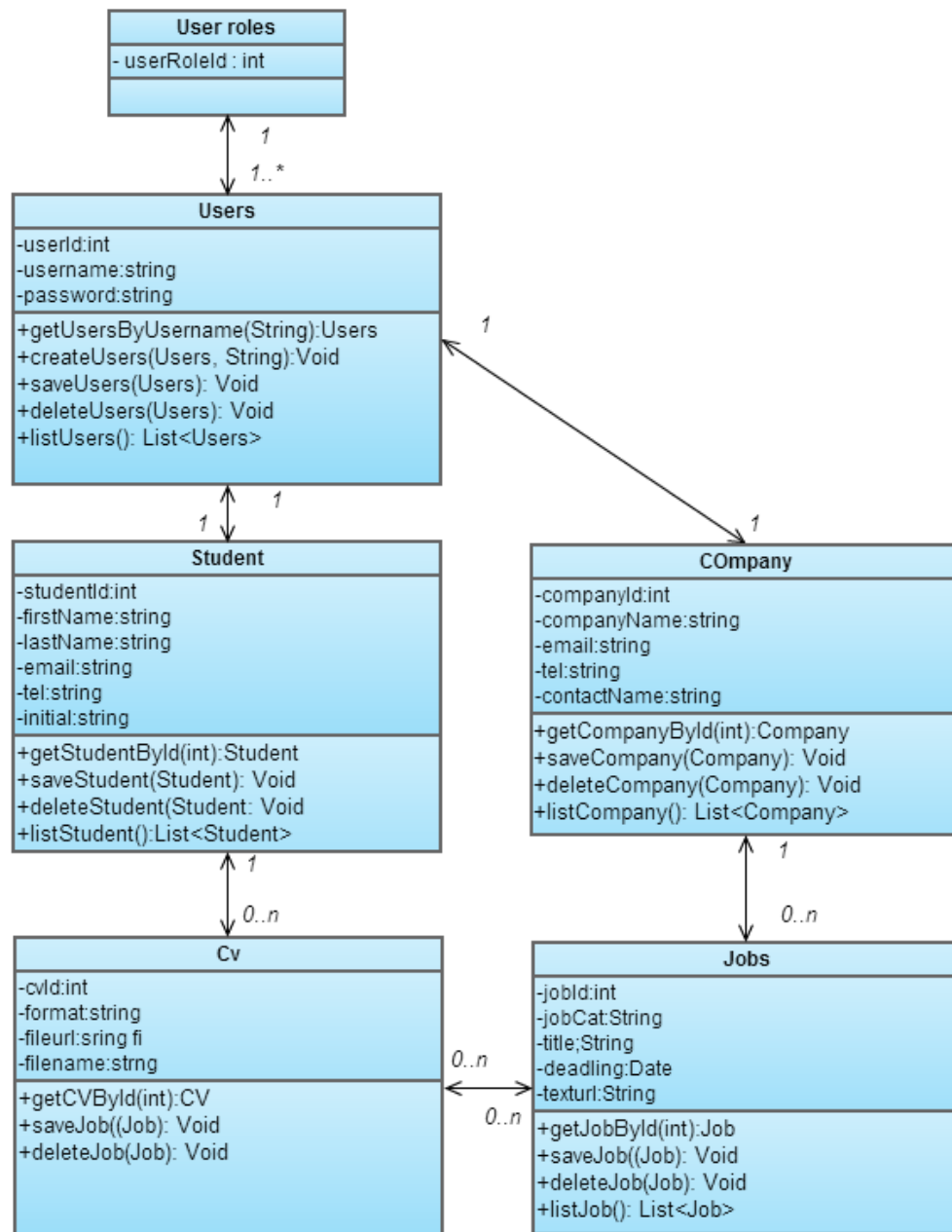


A administrator can almost do every operation to every object, you can do create, update and delete operations to users, CVs, Jobs and application. The operation is designed according to the real logical relationship. For example, to delete a student, you will delete his/her applications and CVs firstly.

3 Structural diagrams

Class diagrams

Class Diagram

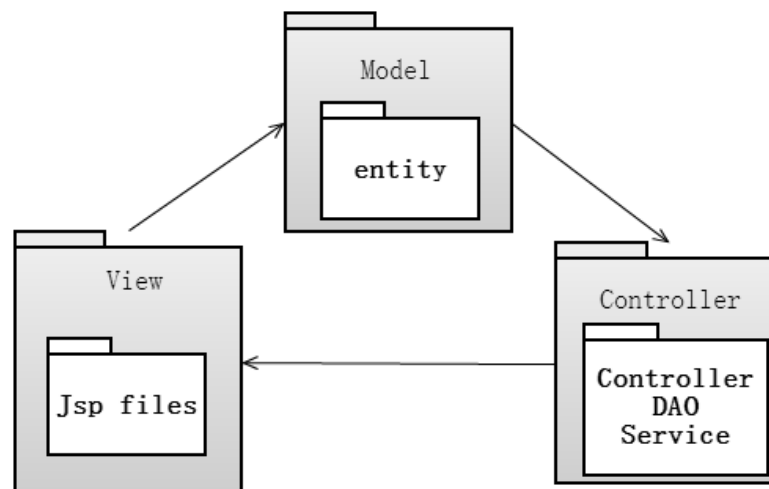


Getter and Setter methods are omitted in the diagram. You can see the property names and methods for each class.

By the annotations beside the arrows, we can see, User has one-to-one relationship to Student and Company. Student is one-to-many with Cv. Company is one-to-many with Jobs. The relationship between Cv and Jobs is many-to-many.

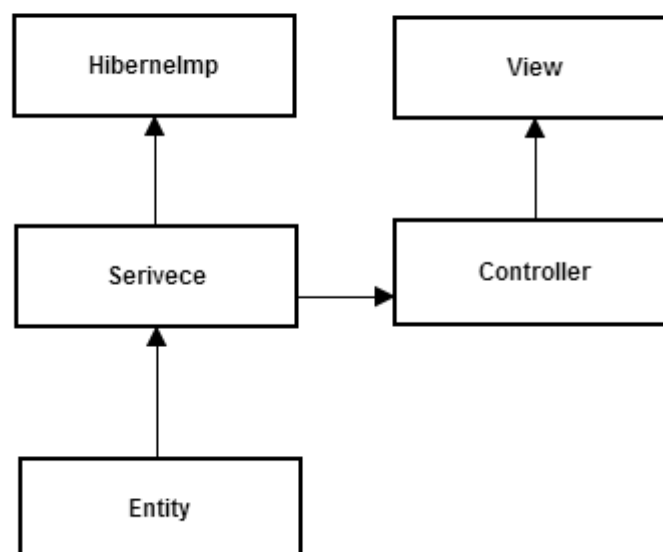
Package Diagram

Package Structure



Basically, there are four packages. In the term of MVC, the entity is the property and methods for entity objects for model layer. DAO and service is the operation for the entity objects. DAO and service are called in controller for control layer. The jsp files are view layer.

Composite structure diagrams



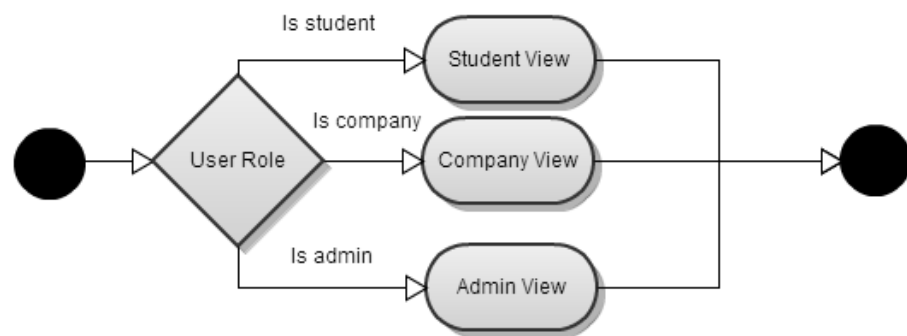
Entity provide interface to service and service provides interface to hibernateImplement to operate the database.

Service also provides interface to controller and controller provides interface to view, so user can browse and do operation,

4 Behavioural view diagrams

Authentication

Activity Diagram



When you login, you will choose your role.

If you are student, you will be redirected to the view of students.

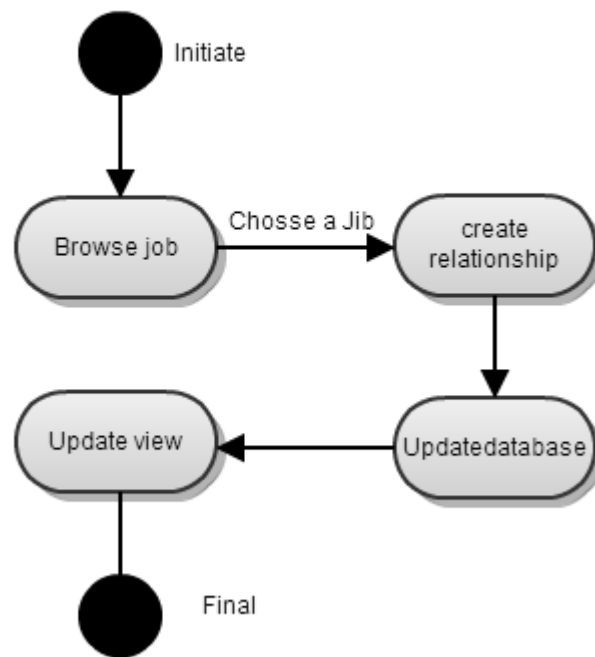
If you are company, you will be redirected to the view of companies.

If you are administrator, you will be redirected to the view of administrators.

Then the authentication activity is over

Make application by a CV

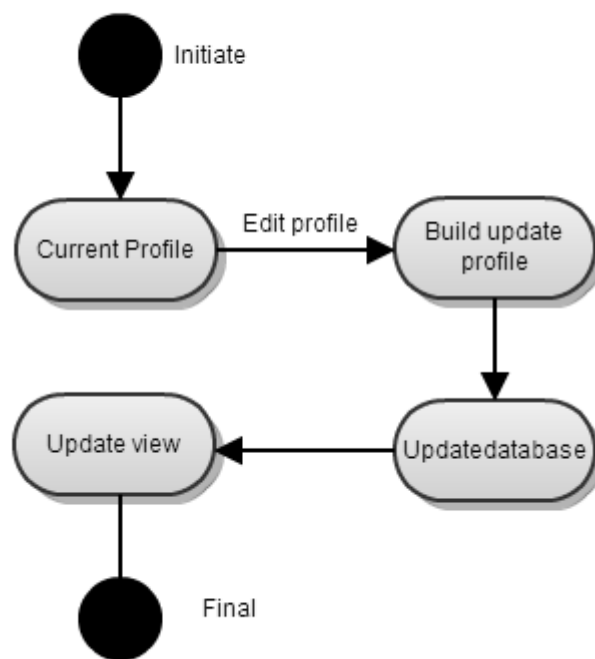
State diagrams



When you choose a CV to initiate a application, you would choose a job firstly. After that, the system will update the database to link the CV and the job. Then a updaed view page will display.

Change Profile

State diagrams



When you want to update your profile, click the link and the operation initiate, the view for current profile will display. After editing the profile, system will build it and update the database. Then you will receive the new view page and this operation is over.

5 Deployment

- Load the Netbeans project, we use standard Hibernate and Spring framework came with Netbeans. All extra jar files has been included and linked in the project properties
- We use mysql as the database server. You have to create the database then connect to it and run mySQLdatabaseCreation.sql to create the tables. You can fetch in some data for testing using SQL commands. I suggest you add some users, user roles, students and companies because we have not implementing the sign-up function yet.
- Next step is edit jdbc.property and hibernate.cfg.xml to matching to yours server settings. There is another property in jdbc.properties named default.file.path which will be the folder where you will hold all the uploaded files. The developer machines all use Windows, so we are using “\\” as the file separator
- After all that set up, build and run the application. We are using Tomcat as the Server and have not test against other servers

6 User manual

After running the project, you will see the index page. Choose your role and you will be redirected to your starting page. You will have to fill in the login page and have the correct credential for the role to access that section. The sign-up function is not available, so you need to fetch the data into the database for testing this. The logout button is always located in the top right corner.

For Student role, your page will have three tabs:

- One tab for viewing the CVs you have uploaded and which jobs is being applied using that CV. You can download the CV, edit (re-upload) it or delete it. You can also download the description of the jobs being applied
- Another tab is for browsing the job that you have not applied yet. You can download the description, and sort the job base on some criteria.
- The last tab is for uploading a new CV. You will be lead to a page where u can upload the file. If no error returns, you will be returned to your CV list tab with the new CV shows up.
- There is a link for updating your profile on the top right corner, next to your name which will go to a form where u can edit your details and save it, or cancel to return to your start page.

For Company role, your page will have two tabs:

- One tab for viewing the Jobs you are providing and which jobs is being applied using which CV. You can download the CV and edit the job you are offering.
- The other tab is for creating a new Job vacancy. You will be lead to a page where u can upload the description file and type in all other details. If no error returns, you will be returned to your CV list tab with the new CV shows up.
- There is a link for updating your profile on the top right corner, next to your name which will go to a form where u can edit your details and save it, or cancel to return to your start page.

For Admin role, your page will have two tabs:

- One tab for viewing the Students in the system. You can disable/enable the account and delete it.
- The other tab for viewing the Companies in the system. The functions are the same as above.

7 Contribution:

a) Giang Tran Hoang – URN 6073931:

- Set-up the projects and dependencies
- Design and create the database,
- Create the Entity classes using reverse engineering, with some customisation on the relationships handling mechanism (lazy initiation, many-to-many cascading without extra entity, etc...)
- Create all the DAO and Service classes.
- Code, test and debug most of the controller classes and all the transactions between three M-V-C layers.
- Write the user manual in the documentation.

b) Yan Huang – URN6219669

I coded the jsp files, some parts of controllers and document. Giang is really kind and helpful. I think we contributed half to half (50:50).