

VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



WEB PROGRAMMING (CO3049)

Assignment

Design and develop a Job Portal

Advisor: Prof. Nguyễn Đức Thái
Students: Nguyễn Lê Nhật Dương - 1952638
 Nguyễn Minh Tâm - 1952968
 Nguyễn Đức Thành - 1952983
 Lê Đức Cầm - 1952588

HO CHI MINH CITY, DECEMBER 2022



Contents

1 Work assignment	2
2 Introduction	2
2.1 Context of the project	2
2.2 Scope of the project	2
2.3 Relevant stakeholders	3
2.4 Expectation on the system	3
3 Functional and non-functional requirements	3
3.1 Functional requirements	3
3.2 Non-functional requirements	3
3.3 Use-case diagram of the system	4
4 Database	5
5 System modelling - Class diagram	5
6 Architecture design	6
6.1 Introduction to MVC	6
6.2 Applying MVC into our model	8
7 Mock-up design	8
8 Final product	9
9 Conclusion	14



1 Work assignment

No.	Team member	Changes	Contribution
1	Nguyễn Lê Nhật Dương	Add section 5 Add section 6 Design all JD related features	25%
2	Nguyễn Minh Tâm	Add section 2 Add section 7 Design authentication pages	25%
3	Nguyễn Đức Thành	Add section 4 Add section 8 Design home page & layout of the web	25%
4	Lê Đức Cầm	Add section 3 Add section 9 Design all CV related features	25%

2 Introduction

2.1 Context of the project

A curriculum vitae (CV) is a detailed document that highlights your professional and academic history. It is an essential tool for anyone who looks for a job because it not only can help you outline your skills and experience to employers but also show the employers the value you will bring to their company. A well-crafted CV can grab the employer's attention and make the employer think you are a great fit for the job.

A job description (JD) is a document that gives a summary of a job or a position that the hiring company is offering. It contains multiple information such as essential job requirements, duties, responsibilities, and skills required to perform a specific role. A detailed job description will cover how success is measured in the role so it can be used in performance evaluations and help employers filter out applicants when choosing candidates for interviews.

It is understood that both CV and JD play an important role in recruiting as well as job-hunting since they lay the foundation for the company to get acquainted with candidates. These days, with the rise of job portals, job-hunting has been made to be easier for candidates and employers. They allow JD to be posted online, thus allowing them to be viewed more easily. CV can also be created using preset templates found online. Creating a website that can do both would simplify the process even further is what we are aiming for in this project.

2.2 Scope of the project

The website is hosted locally now. In near future, we expect to host the website on the web.



2.3 Relevant stakeholders

Some of the relevant stakeholders of the system:

- **Guests** that visit the website but do not create account.
- **Candidates** who have account on website and can create CV.
- **Employers** who have account on website and can create job description.
- **Admin** who have account on website and can add new users.

2.4 Expectation on the system

The website is labelled as successfully implemented if it is able to fulfil the below requirements:

- User having an account should be able to interact with the website.
 - Candidates can create CV, view & edit their own CV, view job descriptions and apply/withdraw for a position.
 - Employers can create job descriptions and view candidates' CVs.
 - Admin can view all users' CVs, job descriptions and can add new users.

⇒ In general, it is expected to have a responsive web-based job portal that can help candidate approach their employers. This further increases business intelligence and reduces wasted effort.

3 Functional and non-functional requirements

3.1 Functional requirements

Some of the functional requirements of the desired website are listed below:

- The website shall allow guests to create account.
- The website shall allow candidates to create, edit and view CV.
- The website shall allow candidates to view and apply for a job using one of their CVs.
- The website shall allow employers to view the candidates' CVs.
- The website shall allow employers to create and edit job description.
- The website shall allow admin to add new users.

3.2 Non-functional requirements

Some of the non-functional requirements of the desired website are listed below:

- The website should allow accidental downtime for at most 5 minutes.
- The uptime rate of the website should be above 90% of the time.
- The website should allow only authenticated users to use its features.
- The website should be responsive and usable on different browsers.

3.3 Use-case diagram of the system

In order to have a more vivid visualization on the website, the use-case diagram in Figure 1 is produced to give us an illustration of interaction between the website - job portal - and its surroundings or actors.

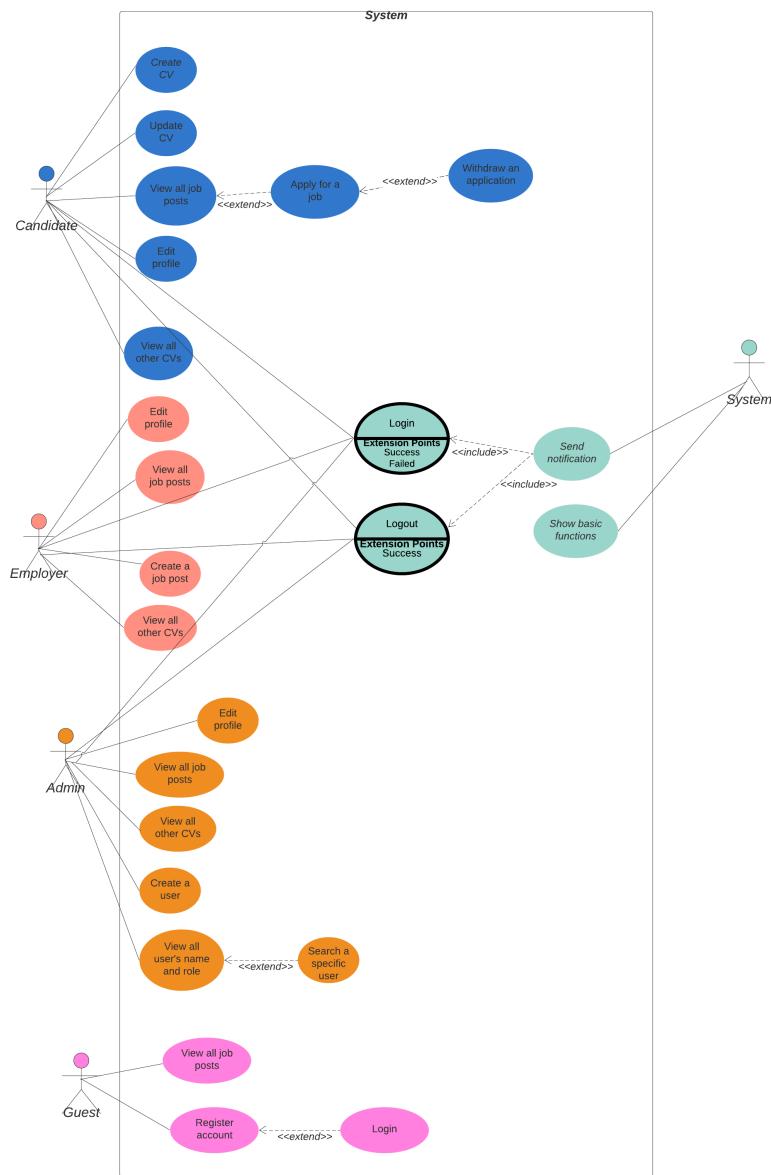
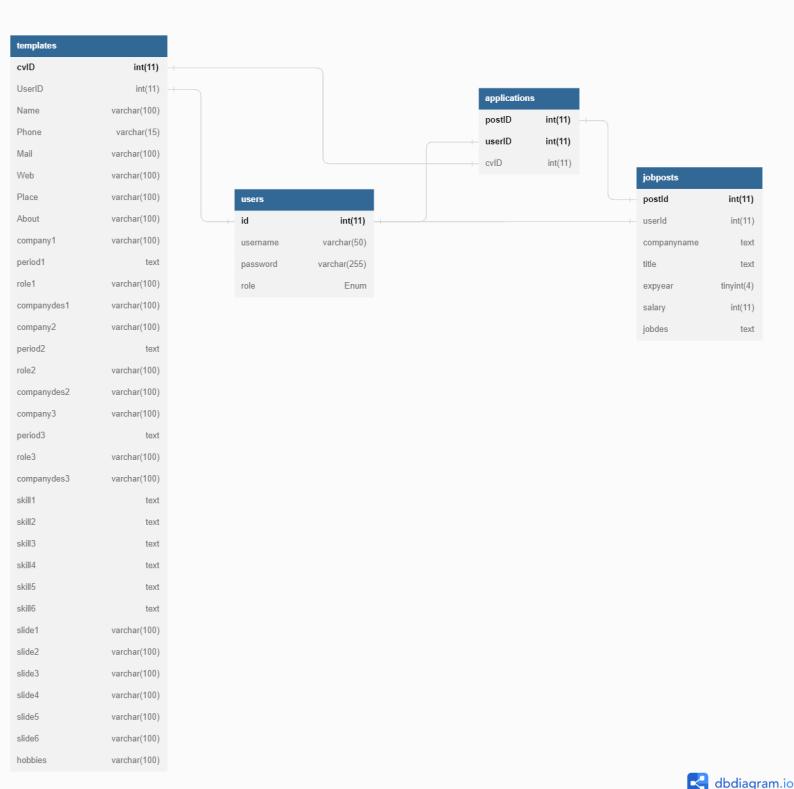


Figure 1: Use-case diagram of the website

4 Database

In order to store all the necessary data, we will need data storage of our own on the server side. This can be achieved with a relational database, particularly to help with showing the relationship between tables.



dbdiagram.io

Figure 2: Relational schema of the database

5 System modelling - Class diagram

Now we establish the class diagram in order to have a closer look at how the website is organized. Below is how the class diagram represents the website. After logging in successfully, users can perform specific features based on their role.

1. Candidates can create, edit and view their CVs, view and apply for a job.
2. Employers can create their job descriptions and view candidates' CVs.
3. Admins can add new users, search for users by username and view job descriptions and all candidates' CVs.

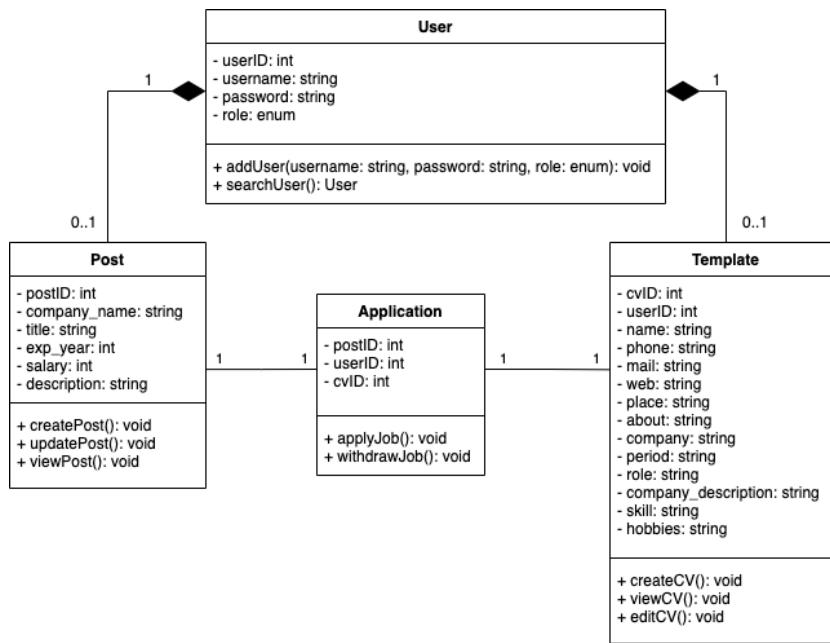


Figure 3: Class diagram of the website

6 Architecture design

6.1 Introduction to MVC

Before applying the MVC to our website, we would like to introduce some key features of the MVC along with its pros and cons.

What is MVC?

MVC stands for Model View Controller. It is a design pattern which separates presentation and interaction from the system data. The system is structured into three logical components that interact with each other.

- The Model component manages the system data and associated operations on that data.
- The View component defines and manages how the data is presented to the user.
- The Controller component manages user interaction and passes these interactions to the View and the Model.

When to use MVC?

MVC is used when there are multiple ways to view and interact with data. It is more of an architectural pattern, but not for a complete application. It is also used when the future requirements for interaction and presentation of data are unknown. MVC mostly relates to the UI/interaction layer of an application.

Advantages of MVC

In here we introduce some advantages of MVC that we think it is suitable for our website:



- Easily modifiable: Using the MVC methodology allows easy modification of the entire application. It allows the data to change independently of its representation and vice versa. So, any changes in a certain section of the application will never affect the entire architecture. This, in turn, will help to increase the flexibility and scalability of the application.
- Multiple views: In the MVC architecture, developing different view components for your model component is easily achievable. It supports the presentation of the same data in different ways, with changes made in one representation shown in all of them.
- Easy planning and maintenance: The MVC paradigm is helpful during the initial planning phase of the application because it gives the developer an outline of how to arrange their ideas into actual code. It also helps limit code duplication and allows easy maintenance of the application.

Disadvantages of MVC

Besides the advantages, MVC pattern also has some disadvantages that we want to illustrate:

- High complexity: The MVC pattern introduces new levels of indirection and this increases the complexity of the solution. In some cases, it is not right suitable for small applications which has an adverse effect on the application's performance and design.
- Cost of frequent updates: developers cannot completely ignore the view of the model even if they are coupled. If the model undergoes frequent changes, the views could be flooded with update requests.



6.2 Applying MVC into our model

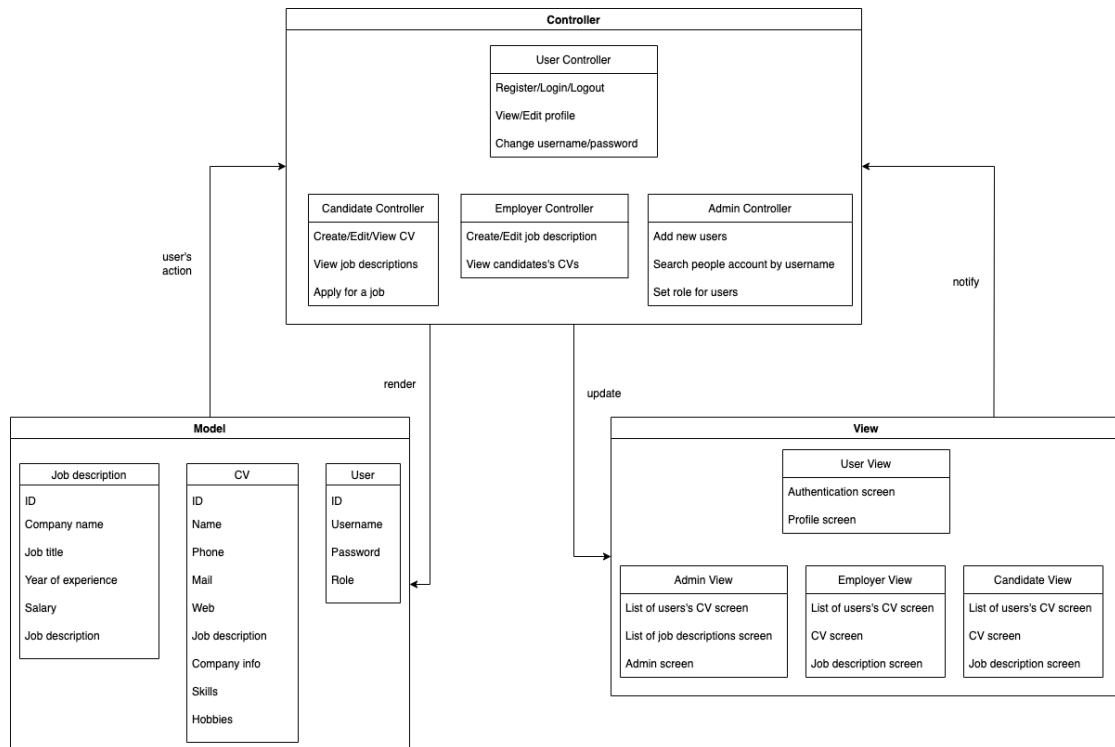


Figure 4: MVC diagram of the website

7 Mock-up design

We use Figma to design our mock-up because it allows us to quickly iterate on our designs.

The figure displays two Figma mock-up designs:

- (a) Mock-up design for home screen:** Shows a landing page with a hero image of a person looking at a computer screen displaying a resume. Text on the page includes "Find the right job for you" and a brief company description. Navigation links at the bottom include "About", "Jobs", "CV", "Logout", "Team's members", "Main Pages", and "CVs and JDs".
- (b) Mock-up design for login screen:** Shows a login form with fields for "Username" and "Password", a "Login" button, and a "Don't have an account? Register" link. Navigation links at the bottom are identical to the home screen.



The image shows two side-by-side mock-up designs for a CV management system. The left mock-up, labeled (a), shows a list of available CVs for review. The right mock-up, labeled (b), shows a detailed view of a specific CV for Tam Nguyen.

(a) Mock-up design for viewing all CVs screen

(b) Mock-up design for viewing specific CV screen

The image shows two side-by-side mock-up designs for job descriptions. The left mock-up, labeled (a), lists several job descriptions with details like title, company, salary, and experience requirements. The right mock-up, labeled (b), provides a detailed view of a specific job post for a Tester position.

(a) Mock-up design for viewing all JDs screen

(b) Mock-up design for viewing specific JD screen

8 Final product

Here is a few notable features of our website to be included in this report.

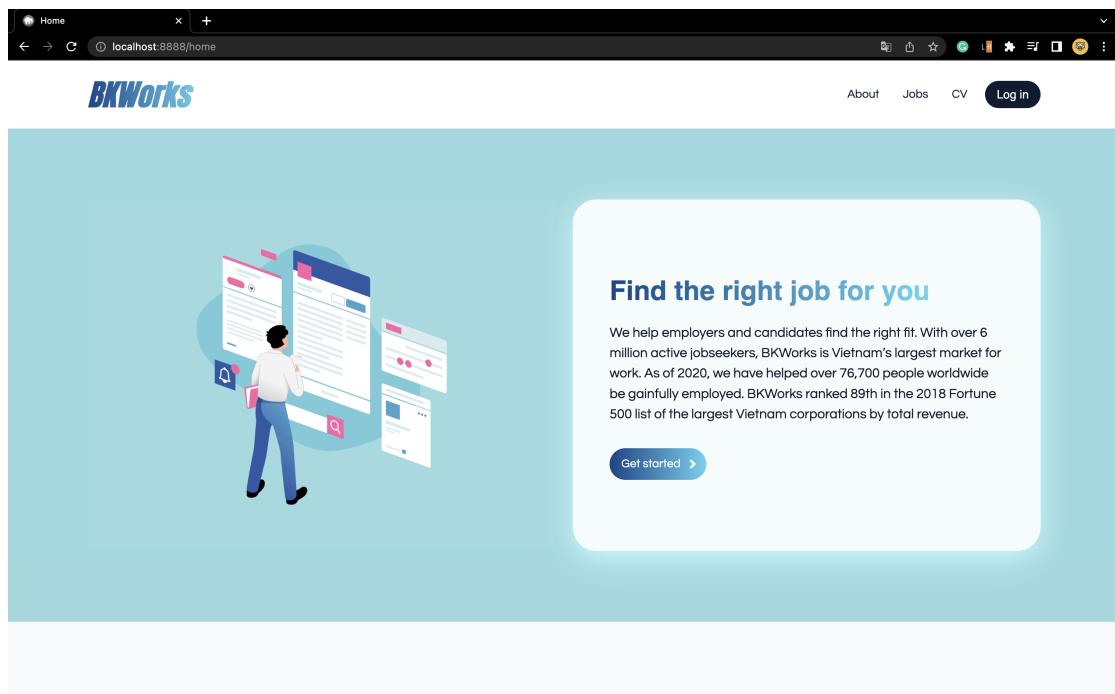


Figure 8: Home screen

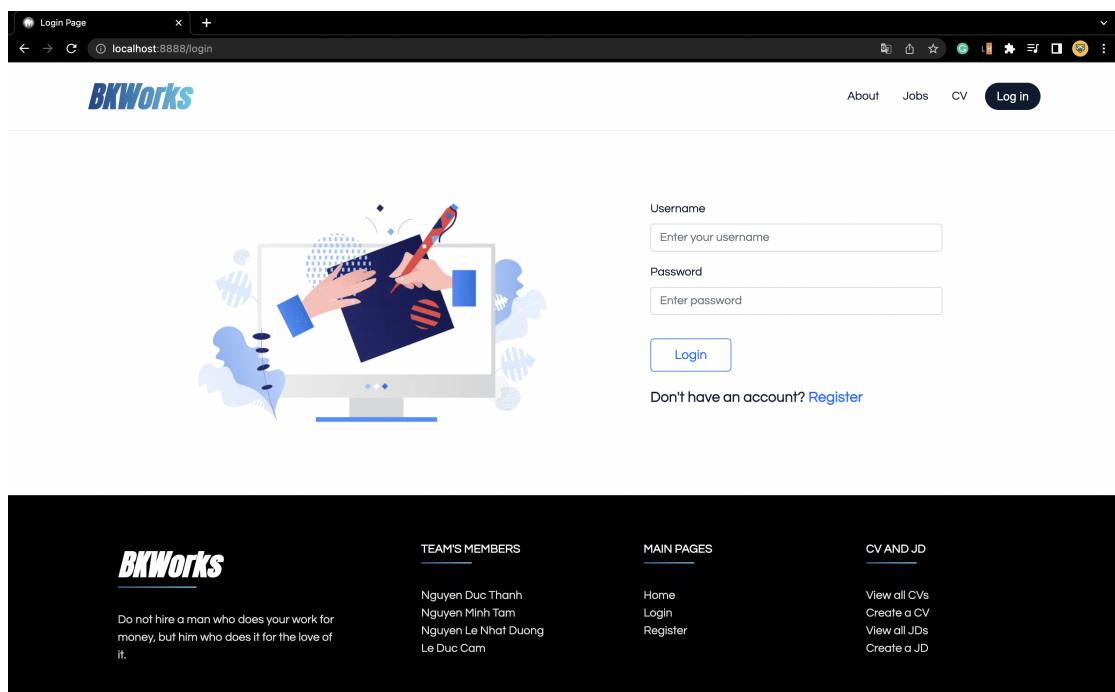


Figure 9: Login screen



Registration Page

localhost:8888/register

BKWorks

About Jobs CV Log in

Username

Password

Confirmed password

Role

Register

Already have an account? [Login](#)



Figure 10: Register screen

View all JDs

localhost:8888/recruit/all

BKWorks

About Jobs CV Log in

All job descriptions

5 jobs available

Tester	\$1000
ABC	
Minimum experience: 2 year(s)	
View	

Full-stack developer	\$5000
VNG	
Minimum experience: 3 year(s)	
View	

Designer	\$3000
Aureole	
Minimum experience: 3 year(s)	
View	

Gamer	\$1500
SGB	
Minimum experience: 2 year(s)	
View	

President	\$8000
PNJ	
Minimum experience: 3 year(s)	
View	

TEAM'S MEMBERS

MAIN PAGES

CV AND JD



Figure 11: View all JDs



The screenshot shows a web browser window for 'View a JD' at 'localhost:8888/recruit/view/1'. The page title is 'BKWorks'. It features a 'Job post' form with the following fields:

- Company's name: ABC
- Job title/position: Tester
- Years of experience: 2
- Salary offered: 1000
- Job description:
File Edit View Format
free

At the bottom right of the form area, it says 'POWERED BY TINY'.

Figure 12: View specific JD

The screenshot shows a web browser window for 'View all CVs' at 'localhost:8888/cv/all'. The page title is 'BKWorks'. It displays a table of available CVs:

CV	Owner	View	Edit
Tam Nguyen	messi	View	Edit

A blue button labeled 'Create your CV' is located below the table.

The footer navigation bar includes links for 'TEAM'S MEMBERS', 'MAIN PAGES', and 'CV AND JD'.

Figure 13: View all CVs



The screenshot shows a web browser window for 'localhost:8888/cv/view/1'. The page title is 'View a CV'. The header includes the 'BKWorks' logo, a 'Welcome back, messi' message, and navigation links for 'About', 'Jobs', 'CV', and 'Log out'. The main content area displays a CV for 'Tam Nguyen'. The CV sections include:

- ABOUT ME**: Placeholder text 'abcd'.
- EXPERIENCES**:
 - First Company**: Role 'Tester' at 'University of Technology' from 'Dec 2022 - Present'.
 - Second Company**: Placeholder text 'example 1'.
- EDUCATION**: MSc in Computer Science from University of London (2016 - 2018).
- LANGUAGES**: First Language, Second Language, Third Language.

Figure 14: View specific CV

The screenshot shows a web browser window for 'localhost:8888/admin/viewAllUsers?page=1&searchUsername=&filterRole='. The page title is 'Admin'. The header includes the 'BKWorks' logo, a 'Welcome back, kylian' message, and navigation links for 'About', 'Jobs', 'CV', 'Admin', and 'Log out'. The main content area displays the 'Administration - Users management' section:

- Add new user**: Form fields for 'Username', 'Password', 'Retype your password', 'Role' (Candidate), and a 'Submit' button.
- Search options**: Form fields for 'Search username', 'Filter role' (All), and a 'Submit' button.
- Users**: A table with columns 'No.', 'Username', and 'Role'. One row is visible with the value 'Candidate' under 'Role'.

Figure 15: Admin screen



9 Conclusion

From this project, we have learned about the way real websites handle requests and serve web pages based on the popular MVC model. We have also expanded our knowledge of PHP as a server-side scripting language and how to use PHP to query data from a database like MySQL. Furthermore, we have learnt how MAMP can help as the web development environment, acting as the local web server and doubling down as a database server, allowing ease of development for web projects. When putting them all together, we were able to build a job portal that can store CVs and JDs for enterprises or job-seeking candidates. This project has taught us many things about building a website and we hope to be able to develop the job portal more in future.