

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



## Web programming

---

## Assignment report

## CV web app

---

Advisor(s): Nguyen Duc Thai

Student(s): Lê Quốc Tuấn ID 2153944

Trần Xuân Trúc ID 1953061

Lê Việt Tùng ID 2153083

Đào Quang Vinh ID 2053586

HO CHI MINH CITY, DECEMBER 2023



## Member list & Workload

No.	Full name	Student ID	Contribution
1	Lê Quốc Tuấn	2153944	100%
2	Trần Xuân Trúc	1953061	100%
3	Lê Việt Tùng	2153083	100%
4	Đào Quang Vinh	2053586	100%



## Contents

<b>1</b>	<b>Introduction.</b>	<b>5</b>
1.1	MVC design pattern. . . . .	5
1.2	Library and extensions used. . . . .	6
1.2.1	Bootstrap 5. . . . .	6
1.2.2	Ajax. . . . .	6
1.2.3	jQuery. . . . .	7
1.2.4	PhpMyAdmin. . . . .	7
<b>2</b>	<b>Requirement analysis</b>	<b>8</b>
2.1	System requirement . . . . .	8
2.1.1	Relevant stakeholder . . . . .	8
2.1.2	functional requirement. . . . .	8
2.1.3	non functional requirement . . . . .	8
2.2	Use case diagram . . . . .	9
2.2.1	System diagram . . . . .	9
2.2.2	Description . . . . .	10
<b>3</b>	<b>Design</b>	<b>16</b>
3.1	Interface design . . . . .	16
3.2	Database design . . . . .	17
<b>4</b>	<b>Implementation</b>	<b>19</b>
<b>5</b>	<b>Testing</b>	<b>19</b>
5.1	Log in/ Log out . . . . .	20
5.2	Search page . . . . .	21
5.3	CV page . . . . .	21
<b>6</b>	<b>Conclusion</b>	<b>22</b>
6.1	System Evaluation . . . . .	22
6.2	Future Software Development . . . . .	22



## Preface

This assignment aims to follow instructions on Bkel and create an interactive CV-showing app. To complete the assignment it will require us to apply both theoretical and practical knowledge. The assignment not only help us to enhance our coding skills and application of practical knowledge, it also help us improve our communication and team work skill.

Our team would like to extend our heartfelt appreciation to Dr. Nguyen Duc Thai, who is our main instructor for this assignment. Thanks to his wholehearted guidance, our team completed the report on schedule and solved all the questions. It has been the guideline for all team activities and maximized the supportive relationship between lecturer and students in the educational environment.



# 1 Introduction.

## 1.1 MVC design pattern.

The Model-View-Controller (MVC) design pattern outlines that an application comprises a data model, presentation information, and control information, emphasizing the separation of these components into distinct objects. While MVC serves as an architectural pattern, it primarily pertains to the UI/interaction layer of an application. Additional layers like business logic, a service layer, and a data access layer are typically necessary for a complete application.

1. The Model encapsulates the core application data and logic responsible for presenting the data to users. It acts as a conduit for data exchange within the application, such as transmitting data between the backend server, view, and database. In Java programming, the Model can be instantiated using a Plain-Old-Java-Object (POJO), which is a straightforward Java class.
2. The View showcases the data from the model to the user, understanding how to access the model's data without interpreting its meaning or the user's potential manipulations. Views are typically represented in formats like .html or .jsp in Java programming, offering flexibility in presentation.
3. The Controller functions as an intermediary between the view and the model. It listens to events triggered by the view or external sources and responds accordingly, often by invoking a method on the model. Due to the notification mechanism connecting the view and the model, any changes resulting from these actions are automatically reflected in the view.

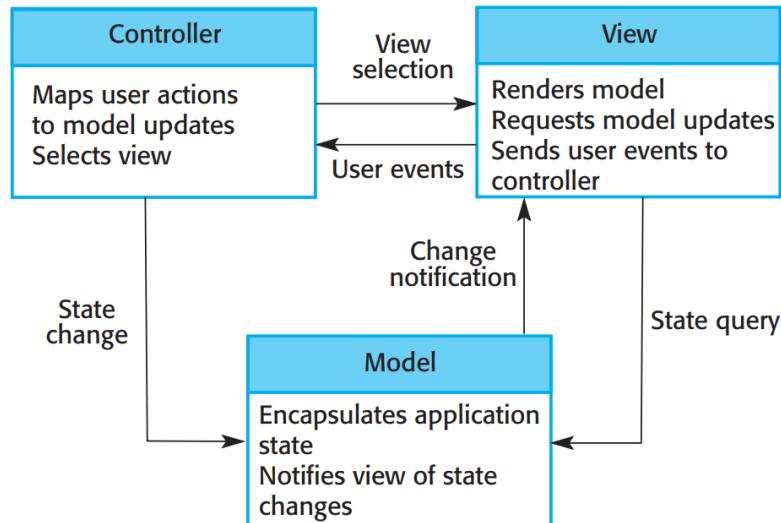


Figure 1.1: MVC design pattern

## 1.2 Library and extensions used.

In the development of our application, incorporated various libraries and extensions to enhance both the aesthetics and functionality. These components work cohesively to create a seamless user experience and adding crucial functionality to our project.

### 1.2.1 Bootstrap 5.

Bootstrap 5, a popular front-end framework, is employed in the development of the user interface. Its responsive design components and utility classes enhance the overall aesthetic and functionality of the application. The grid system, components like navigation bars and modals, and the extensive collection of predefined styles contribute to a cohesive and visually appealing user interface.

### 1.2.2 Ajax.

Ajax (Asynchronous JavaScript and XML) is utilized to implement asynchronous communication between the client and server. This technology allows dynamic updates to the user interface without requiring a page reload. By sending and receiving data asynchronously in the background, Ajax enhances the user experience by providing seamless and responsive interactions.



### 1.2.3 jQuery.

jQuery, a fast and lightweight JavaScript library, is integrated into the application to simplify DOM manipulation and event handling. It streamlines the implementation of interactive features, animation, and AJAX requests. jQuery's cross-browser compatibility ensures consistent behavior across different web browsers, contributing to a more robust and user-friendly application.

### 1.2.4 PhpMyAdmin.

PhpMyAdmin serves as the database management tool, offering a web-based interface for efficient MySQL database administration. It simplifies tasks such as database creation, modification, and maintenance. PhpMyAdmin's intuitive user interface allows developers to interact with the database seamlessly, making it an essential component in managing and organizing application data.



## 2 Requirement analysis

From the requirement from BK we could infer some functionality of the app. The app shall provide the users with the ability to log in to the app, while logged in, The user shall be able to see and print the previous report that they have created.

### 2.1 System requirement

#### 2.1.1 Relevant stakeholder

- User: view and print their CV on demand.
- System administrator: configure form setting, add in further form customization and maintain the web application.

#### 2.1.2 functional requirement.

- **User:**
  - The user shall be able to log in/ log out of the system.
  - The user shall be able to search for their CV that they have created via the search bar on top of the page.
  - The user shall be able to open the CV that they would like to open/view.
  - The user shall be able to print the CV that they have chosen in the form of pdf and specify the printing properties before printing.
- **System administrator:**
  - The system administrator shall be able to log in/log out of the system and will have access to special privileges.
  - The system administrator shall be able to add some further customization for the report form.
  - The admin should be able to delete User and their CV.

#### 2.1.3 non functional requirement

- The website shall be able to have a secure authentication system.

- The interface should be understandable and comprehensive so that users could navigate without trouble even with out any guidance.
- The time between user request for a specific CV and that CV appear on the page should be no more than 100ms.

## 2.2 Use case diagram

### 2.2.1 System diagram

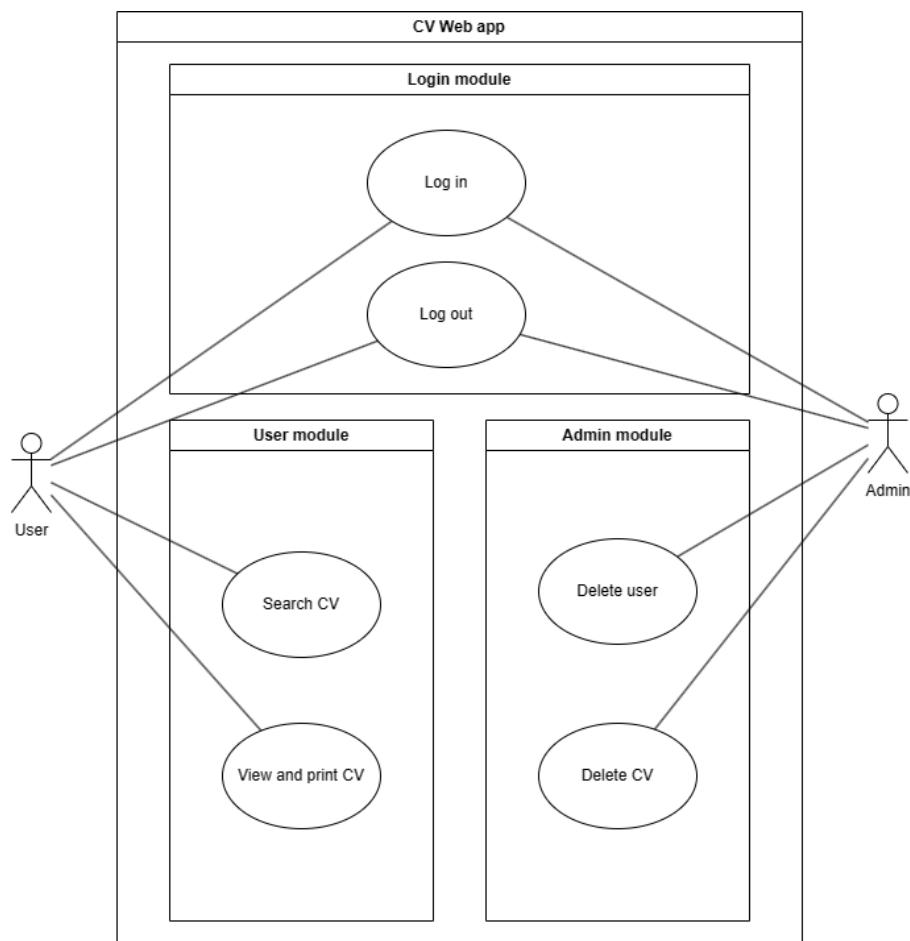


Figure 2.1: Caption



### 2.2.2 Description

Table 2.1: Use Case 1.1: Log In

<b>Use-case ID</b>	U1.1
<b>Use-case Name</b>	Log in
<b>Use-case overview</b>	User want to sign in to the service
<b>Actors</b>	User, Admin
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Users are connected to the Internet.</li></ul>
<b>Trigger</b>	The User clicks on the button “submit”.
<b>Basic Flow</b>	
	<ol style="list-style-type: none"><li>1. User input the username.</li><li>2. User input the password.</li><li>3. User submit the login form.</li><li>4. Application verify the login request.</li><li>5. User is redirected back to the home page</li></ol>
<b>Alternative Flow</b>	
<b>Post condition</b>	<ul style="list-style-type: none"><li>• The User is logged in and at the home page</li></ul>
<b>Exception flow</b>	4b. If the username and password failed to verify, send the user back to step one with error message



Table 2.2: Use Case 1.2: Log Out

<b>Use-case ID</b>	U1.2
<b>Use-case Name</b>	Log Out
<b>Use-case Overview</b>	User wants to log out of the service.
<b>Actors</b>	User, Admin
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Users are connected to the Internet.</li><li>• User is currently logged in.</li></ul>
<b>Trigger</b>	The user clicks on the "Log Out" button.
<b>Basic Flow</b>	
	<ol style="list-style-type: none"><li>1. User clicks on the "Log Out" button.</li><li>2. Application logs out the user.</li><li>3. User is redirected to the login page.</li></ol>
<b>Alternative Flow</b>	
<b>Post Condition</b>	<ul style="list-style-type: none"><li>• The user is logged out and redirected to the login page.</li></ul>
<b>Exception Flow</b>	2b. If there is an error during the logout process, display an error message.



Table 2.3: Use Case 1.3: Search CV

<b>Use-case ID</b>	U1.3
<b>Use-case Name</b>	Search CV
<b>Use-case Overview</b>	Users search for a CV using the CV title.
<b>Actors</b>	User
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Users are connected to the Internet.</li><li>• Users are authenticated.</li></ul>
<b>Trigger</b>	The user presses enter after inputting CV title.
<b>Basic Flow</b>	<ol style="list-style-type: none"><li>1. User inputs the CV title.</li><li>2. User submits the search form.</li><li>3. Application processes the search request.</li><li>4. System displays the matching CVs.</li></ol>
<b>Alternative Flow</b>	<p>1b. No matching CVs found. <i>System displays a message indicating no results.</i></p>
<b>Post Condition</b>	<ul style="list-style-type: none"><li>• The user is shown the matching CVs.</li></ul>
<b>Exception Flow</b>	<p>3b. If there is an error processing the search request, display an error message.</p>



Table 2.4: Use Case 1.4: View and Print CV

<b>Use-case ID</b>	U1.4
<b>Use-case Name</b>	View and Print CV
<b>Use-case Overview</b>	Users open a CV for viewing and printing.
<b>Actors</b>	User
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Users are connected to the Internet.</li><li>• Users are authenticated.</li><li>• The users have available CVs for viewing.</li></ul>
<b>Trigger</b>	The user clicks on the row containing the desired CV.
<b>Basic Flow</b>	<ol style="list-style-type: none"><li>1. User selects a CV from the displayed list.</li><li>2. User chooses the option to view the selected CV.</li><li>3. Application loads and displays the selected CV.</li><li>4. User has the option to print the displayed CV.</li></ol>
<b>Alternative Flow</b>	
<b>Post Condition</b>	<ul style="list-style-type: none"><li>• The user views and may print the selected CV.</li></ul>
<b>Exception Flow</b>	3b. If there is an error loading the CV, display an error message.



Table 2.5: Use Case 1.5: Delete User

<b>Use-case ID</b>	U1.5
<b>Use-case Name</b>	Delete User
<b>Use-case Overview</b>	Admin deletes a user account.
<b>Actors</b>	Admin
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Admin is connected to the Internet.</li><li>• Admin is authenticated.</li><li>• There is at least one user account to delete.</li></ul>
<b>Trigger</b>	The Admin selects a user account for deletion.
<b>Basic Flow</b>	<ol style="list-style-type: none"><li>1. Admin selects a user account to delete.</li><li>2. Admin confirms the deletion action.</li><li>3. Application deletes the selected user account.</li><li>4. System notifies Admin that the user account has been successfully deleted.</li></ol>
<b>Alternative Flow</b>	2b. Admin cancels the deletion action. System notifies Admin that the deletion action has been canceled.
<b>Post Condition</b>	<ul style="list-style-type: none"><li>• The selected user account is deleted from the system.</li></ul>
<b>Exception Flow</b>	<ol style="list-style-type: none"><li>1. If there is an error during the deletion process, display an error message.</li></ol>



Table 2.6: Use Case 1.6: Delete CV

<b>Use-case ID</b>	U1.6
<b>Use-case Name</b>	Delete CV
<b>Use-case Overview</b>	Admin deletes a CV from the system.
<b>Actors</b>	Admin
<b>Preconditions</b>	<ul style="list-style-type: none"><li>• The system is running.</li><li>• Admin is connected to the Internet.</li><li>• Admin is authenticated.</li><li>• There is at least one CV to delete.</li></ul>
<b>Trigger</b>	The Admin selects a CV for deletion.
<b>Basic Flow</b>	<ol style="list-style-type: none"><li>1. Admin selects a CV to delete.</li><li>2. Admin confirms the deletion action.</li><li>3. Application deletes the selected CV.</li><li>4. System notifies Admin that the CV has been successfully deleted.</li></ol>
<b>Alternative Flow</b>	<p>2b. Admin cancels the deletion action. <i>System notifies Admin that the deletion action has been canceled.</i></p>
<b>Post Condition</b>	<ul style="list-style-type: none"><li>• The selected CV is deleted from the system.</li></ul>
<b>Exception Flow</b>	<ol style="list-style-type: none"><li>1. If there is an error during the deletion process, display an error message.</li></ol>



## 3 Design

### 3.1 Interface design

Based on the requirements of the project our team has determined that the project will have 3 pages: that being login, search page and a CV page.

**Login page:** On this page, users are required to log in and authenticate their accounts before gaining access to the CV search functionality. Authentication is a mandatory step to ensure a secure and personalized experience when searching for CVs.

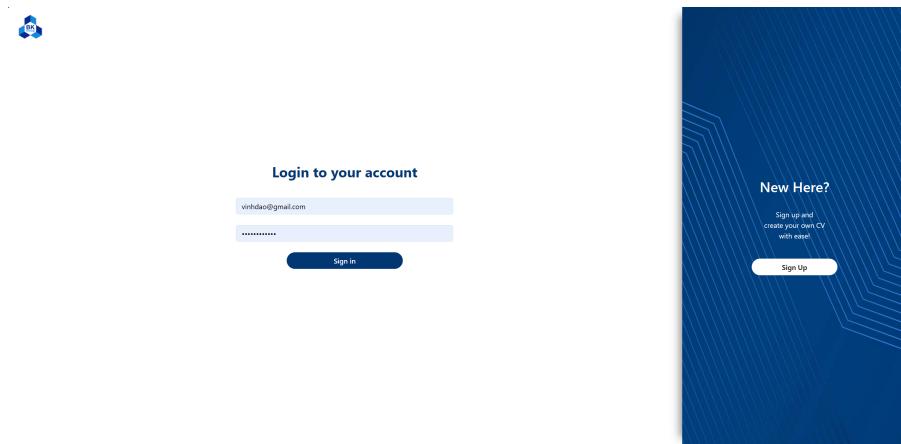


Figure 3.1: Login page

**Search page:** On this page, users will have the capability to search for their CV, and additionally, they will find an option to log out of the platform. This allows users to efficiently navigate through the CV search feature while providing the flexibility to securely log out when needed.

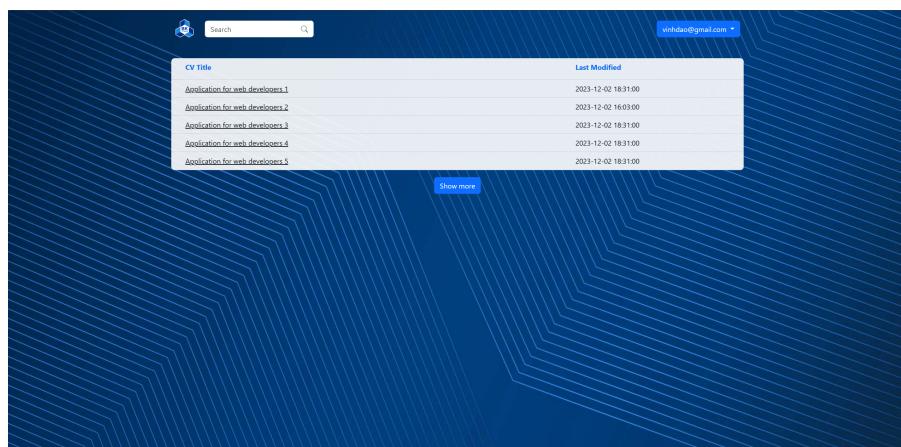


Figure 3.2: Search page

**CV page:** Upon selecting their CV, users will be seamlessly redirected to the dedicated CV page. Here, they will find options to both view and print their CV, providing them with the convenience to access a digital version for online viewing and the flexibility to generate a physical copy if desired.



Figure 3.3: CV page

### 3.2 Database design

Although was not required in the assignment, as we couldn't get the database of the group that was doing the assignment 1. We have decided to develop our own database.

This is the design of our ER diagram which will include a total of 5 entity and their subsequence attribute.

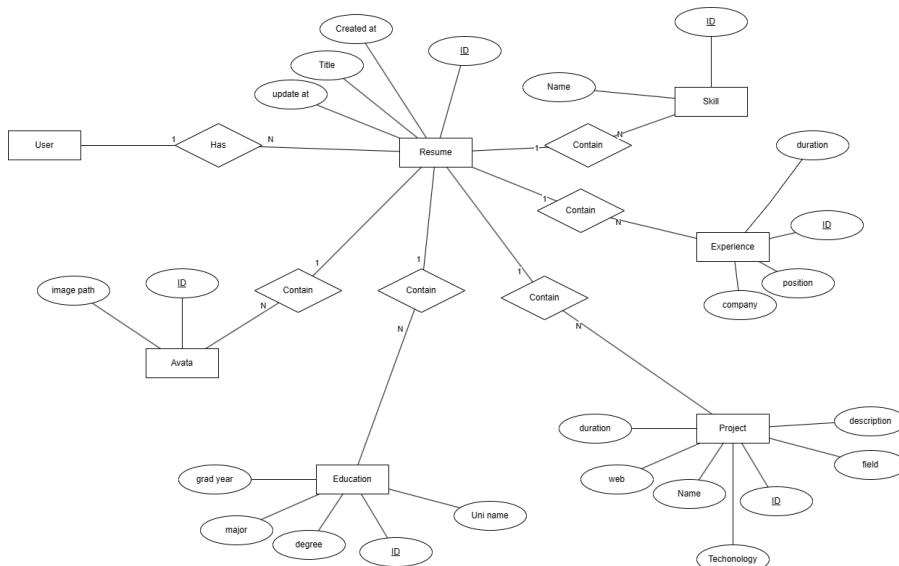


Figure 3.4: ER diagram



The following ER diagram will be mapped to the Schema which create a total of 5 table.

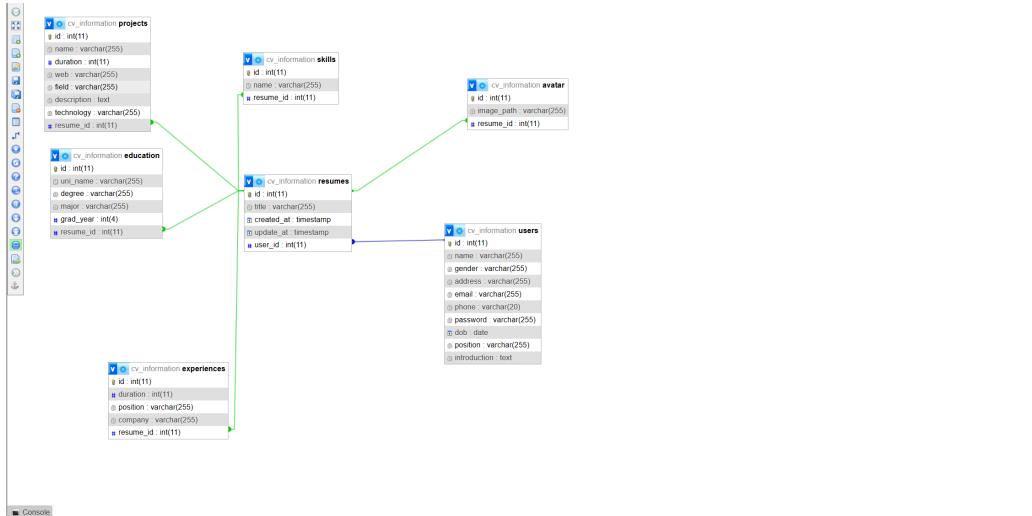


Figure 3.5: Schema



## 4 Implementation

Our team has developed The Code which can be found at this github [Link](#)

## 5 Testing

As the time were limited, we could only do and set up the test for a couple of function. All test will be done through a browser.

### Testing plan:

- Precondition: the database has been created using db\_init.php in db\_init folder.
- Functionality that will be tested:
  - Log in/log out.
  - Search page.
  - CV page.
- Testing strategy: Testers will directly execute test cases on their browsers.
- Acceptance Criteria:
  - The application runs smoothly on the tester's browser, with no website conflicts.
  - Successfully passes the defined test cases.



## 5.1 Log in/ Log out

Test Case	Test Steps	Expected result	Actual result	evaluation
Log in				
Enter the correct username and password	<ul style="list-style-type: none"><li>User enter a username: vinhdao@gmail.com.</li><li>User enter password: truetruetruetrue.</li><li>User press submit.</li></ul>	User are redirected to home page	User are redirected to home page	Pass
Enter the wrong username but correct password	<ul style="list-style-type: none"><li>User enter a username: hacker.</li><li>User enter password: somerealpassword.</li><li>User press submit.</li></ul>	User are required to reinput the username and password	User are required to reinput the username and password	Pass
Enter the correct username but wrong password	<ul style="list-style-type: none"><li>User enter a username: admin.</li><li>User enter password: wrongpassword.</li><li>User press submit.</li></ul>	User are required to reinput the username and password	User are required to reinput the username and password	Pass
Enter the Wrong username and password	<ul style="list-style-type: none"><li>User enter a username: hacker.</li><li>User enter password: wrongpassword.</li><li>User press submit.</li></ul>	User are required to reinput the username and password	User are required to reinput the username and password	Pass



## 5.2 Search page

Test Case	Test Steps	Expected result	Actual result	evaluation
Search page				
User search for an existing CV	<ul style="list-style-type: none"><li>User enter an existing CV title into the search bar.</li></ul>	User are show the CV that the user searched for.	User are show the CV that the user searched for.	Pass
User search for a non-existing CV	<ul style="list-style-type: none"><li>User enter a non existing cv title into the search bar.</li></ul>	Search results show found no CV.	Search result show found no CV.	Pass
User selects a CV to view	<ul style="list-style-type: none"><li>User select a CV from the available CV list</li></ul>	User are directed to CV page.	User are directed to CV page.	Pass
Log out				
Log out of the system	<ul style="list-style-type: none"><li>User Click on log out button in the navigation bar</li></ul>	User is log out and their session is deleted	User log out and their session is deleted	Pass

## 5.3 CV page

Test Case	Test Steps	Expected result	Actual result	evaluation
Search page				
Printing the CV	<ul style="list-style-type: none"><li>User click on the print button</li><li>User specify the printing properties.</li><li>User press submit.</li></ul>	A downloadable pdf file is created	A downloadable pdf file is created	Pass



## 6 Conclusion

### 6.1 System Evaluation

In conclusion, our team has successfully developed a comprehensive web application that facilitates user login, enabling the viewing and printing of CV pages.

### 6.2 Future Software Development

Looking ahead, our team envisions several enhancements to further elevate the project's quality:

- Streamline the transition between the Search page and CV viewing page for an improved user experience.
- Empower users in assignment 1 by integrating the capability to create a fully developed web app directly from the Create CV feature.
- Strengthen security measures through the incorporation of additional authentication methods.
- Expand form customization options, allowing users to effortlessly transfer data between forms with distinct styles.

These future developments aim to enhance functionality, usability, and security, ensuring a robust and user-friendly web application.



## References

- [1] Geek for geek. MVC Framework Introduction. <https://www.geeksforgeeks.org/mvc-framework-introduction/>, 2023.
- [2] ian Sommerville. *Software Engineering 10th edition*. Pearson, 2016.
- [3] World Wide Web Consortium (W3C). W3Schools Online Web Tutorials. <https://www.w3schools.com>, 2023.