|  |  |
| --- | --- |
| **MIDTERM report**  Viết Trung Personal Website | Abstract  Viết Trung Website is the midterm project of the course Web Application Development. This is the website having my personal information according requirements of the midterm project.  Web Application Development – Dr. Sinh  Hồ Viết Trung – ITITIU17098 |

Contents

[**1.** **Description Personal website application:** 2](#_Toc39013418)

[1.1 **Assumption**: 2](#_Toc39013419)

[1.2 **Tools and Techniques**: 2](#_Toc39013420)

[1.3 **Content**: 2](#_Toc39013421)

[1.4 **Database:** 2](#_Toc39013422)

[**2.** **Design and demo:** 3](#_Toc39013423)

[2.1 **Cover page:** 3](#_Toc39013424)

[2.2 **Home** **page**: 5](#_Toc39013425)

[2.3 **About me Page**: 9](#_Toc39013426)

[2.4 **Contact Page**: 14](#_Toc39013427)

[2.5 **Resume Page**: 18](#_Toc39013428)

[2.6 **Get Message Page**: 21](#_Toc39013429)

[2.7 **Message Java Servlet**: 25](#_Toc39013430)

[2.8 **Sitemap**: 27](#_Toc39013431)

[2.9 **Cascading Style Sheets:** 28](#_Toc39013432)

[**3.** **Use case diagram** 31](#_Toc39013433)

[3.1 **Use Case Diagram**: 31](#_Toc39013434)

[3.2 **Explanation**: 31](#_Toc39013435)

[**4.** **Design a login page with the requirements:** 32](#_Toc39013436)

[4.1 **Login Page:** 32](#_Toc39013437)

[4.2 **Login Page Servlet – Registration Java Server Page**: 34](#_Toc39013438)

[4.3 **Store Java Servlet**: 38](#_Toc39013439)

[4.4 **JavaBean**: 40](#_Toc39013440)

[**5.** **References** 43](#_Toc39013441)

# **Description Personal website application:**

## **Assumption**:

* Viết Trung Personal website is the personal website with my real information. In this website I do not show my Student ID because of privacy problem.
* I just apply all of knowledge in the course Web Application Development to do it, so there are several problems in this website.
* All the images in this website are the real image of me.
* I am not good in design so I use simple ways to implement the front-end of the website.

## **Tools and Techniques**:

* *Editor*: Eclipse IDE
* *Server*: Tomcat 8.5
* Language:   
  + Front-end: HTML5 – CSS3 – Bootstrap 4  
  + Back-end: Java Servlet and Java Server Page
* I use Navbar to manage all of the pages in the website. For example, Navbar will in the top of all the pages. When, you click on the name of the page then it will lead to these pages.
* I use Bootstrap template to make the front-end look better and save time to make them easy to use. I combine between CSS and Bootstrap to make the web more flexible.

## **Content**:

- All of the information in the website is my real information.  
- I do not have many information in order to add to the Website. So, the content is not too good in performance.  
- I will give you a general information in Home page with contact details in the Contact page. If you want to send message directly, It would be recorded in the backend side with Message servlet.

- In the footer of these pages, I embedded the Social Media such as: Facebook, Instagram to provide a connecting way.

## **Database:**

* I prefer to use MySQL to manage the database. I learned DBMS MySQL in the course Introduction to Database, so I have a little experience with it.
* However, in the project there is not too many data in order to use MySQL.
* I use Form to collect the data from the user like Sending message directly via Website, then I will use servlet to show the Details of the message. But I think, most of the people will use email to contact instead of via Website.

# **Design and demo:**

## **Cover page:**

* The cover page is the HTML file that will show the basic information of the Viết Trung Website.
* This is the beginning of the project.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains the Name of the Website and some basic information about the website. There is a button Learn More, which leads to About me page.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:   
  This is the beginning of the project, so it can access to another pages in the project by using navbar.   
  When you click on the Logo in the navbar it always returns to this page.

|  |
| --- |
| **Figure 1. Cover Page – Index.html** |
| **Figure 2. Code of the Cover Page – Index.html** |

## **Home** **page**:

* The Home Page is a HTML Page with the General Personal Information such as: Full name, Date of Birth, Address, Education and Email.
* This is the introduction about myself with the general information and my image.
* In this page, there are 2 buttons: 1 lead to My Instagram and other lead to My Facebook.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains my image and a table with the Personal Information such as: Full name, DOB, Gender, Address, Education and Email. There are 2 buttons leading to My Instagram and My Facebook.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:

|  |
| --- |
| **Figure 3. Home Page – Home.html** |
| **Figure 4-1. Code of the Home.html   Figure 4-2. Code of the Home.html  Figure 4-3. Code of the Home.html** |

## **About me Page**:

* The About Page is a HTML Page with the facts and hobbies of me.
* This is the introduction about myself with the general information and my image.
* In this page, there are 2 buttons: 1 lead to My Instagram and other lead to My Facebook and there is a View My Resume button leading to Resume Page, which is located in Education part.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains 3 parts such as: facts, Hobby, Education.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:

|  |
| --- |
| **Figure 5-1. Demo of the About.html**  **Figure 5-2. Demo of the About.html**  **Figure 5-3. Demo of the About.html** |

|  |
| --- |
| **Figure 6-1. Code of the About.html  Figure 6-2. Code of the About.html   Figrure 6-3. Code of the About.html** |

## **Contact Page**:

* The Contact Page is a HTML Page with the Contact Details.
* This is the table of the Contact details such as: Phone. Email, Facebook, Instagram. Or you can send message directly to me via Button Send Message.
* In this page, there are 2 buttons: 1 lead to My Instagram and other lead to My Facebook and there is a View My Resume button leading to Resume Page, which is located in Education part.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains a table of the contacts details with the Send Message Button which is could send directly message to me.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:

|  |
| --- |
| **Figure 7. Demo of the Contact.html** |
| **Figure 8-1. Code of the Contact.html**  **Figure 8-2. Code of the Contact.html** |

## **Resume Page**:

* The Resume Page is a HTML Page with the Working and Education Information.
* There are 2 tables for contains Working information like Technologies with the programming language with my experience technologies.
* In this page, there are 2 buttons: 1 lead to My Instagram and other lead to My Facebook and there is a Contact Me button leading to Contac Page.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains 2 table of the Working and Education details with the Contact Me Button which is leading to Contact.html page.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:

|  |
| --- |
| **Figure 9. Demo of the Resume.html** |

|  |
| --- |
| **Figure 10-1. Code of the Resume.html**  **Figure 10-2. Code of the Resume.html**  **Figure 10-3. Code of the Resume.html** |

## **Get Message Page**:

* The Get Message Page is a Java Server Page with the form that message from user to me.
* In this page, there is a basic form that there are 4 labels such as: Full name, E-mail address, Subject and Message text, a submit button to submit the form to server.
* In the Subject, this is the select tag. I assume that 4 types of message: Feedback, Job, Report a bug and Other.
* About the Design: I separate it into 3 parts:  
  + Header: with the navbar to link all of the basic pages of the website such as: Home, About, Contact, Resume with 2 buttons likes Login vs Register.  
  + Main: it contains a decorated form, which is recorded the information of users who want to send message directly to me via website.  
  + Footer: this is the signature about my working. When you click on my name, it will lead to my Facebook.
* Demo:

|  |
| --- |
| **Figure 11. Demo of the getMessage.jsp** |



|  |
| --- |
| **Figure 12-1. Code of the getMessage.jsp**  **Figure 12-2. Code of the getMessage.jsp**    **Figure 12-3. Code of the getMessage.jsp** |

## **Message Java Servlet**:

* The Message Page is a Java Servlet with the page that recorded information from the form that message from user to me.
* About the Design: this page apply basic tag from html to record the information then display to the browser.
* The information will store in the table.
* Demo:

|  |
| --- |
| **Figure 13-1. Message Form with input** |

|  |
| --- |
| **When you press Submit your Message Form:  Figure 13-2. Message Record in the Message.java** |
| **Figure 14-1. Code of the Message.java**  **Figure 14-2. Code of the Message.java** |

## **Sitemap**:

* In Viết Trung Website, I use navbar to manage the connection between the pages.
* From the Cover Page, user can visit all of the Pages via navbar tool: Home Page, About Page, Contact Page and Resume Page. And reverse they also visit each other and return the Cover Page by clicking on the Logo in the Navbar.
* Here is my hierarchical diagram website:

|  |
| --- |
| **Figure 13. Sitemap of the Viết Trung Website** |

## **Cascading Style Sheets:**

* I combine between CSS and Bootstrap framework to decorate the front-end of the website.
* I design the website that the content will display in the center of the page, that can easy to read. I also use some inline CSS to make the text could read easily.
* CODE:

|  |
| --- |
| **Figure 15-1. Code of the Cover.css   Figure 15-2. Code of the Cover.css** |



# **Use case diagram**

## **Use Case Diagram**:

|  |
| --- |
| **Figure 16. Use Case Diagram of Viết Trung Website** |

## **Explanation**:

* For each pages in the Website are functions. Because Personal Website is the website that showing all the information so I don’t have too much function.
* My new function in the project is that user can send directly the message to me via Browser.
* In detail, when the User view my Contact Details, they could use the form in the website to send a message to me. The message will display in the table with the heading is the type of Message, which can easily to collect information.
* In the website, all of the pages can visit each other by using navbar. But, in some cases, I use the link to reference to the pages. For example, View Personal Website you can want know more about me so you can click on the button leading to the About me page instead of scroll up to click on the navbar.
* In the View Contact Details, you can choose to Send message directly to me by fill in the form. So, in this case Send Message via Browser use case is the extension of the View Contact Details use case then I use Include relation between Send Message via Browser use case and Record and Display Message use case because if user don’t choose to send message, there will have any of message to record and display.
* Social Media is an actor of the View Contact Details because they are external system that the contact details references to.

# **Design a login page with the requirements:**

## **Login Page:**

* Following the description of the login Page, Login Page is the Java Server Page.
* The conditional is:
  + If user login successful with username(midterm) and password (22042020), then they will go to the registration page (Registration.jsp).
  + If user login unsuccessful with username and password, the browser will print out the message: “Sorry username or password error!”.
  + The user must input Username vs password if they want to login. If they forgot to fill in, Form will make warning.
* Demo:

|  |
| --- |
| **Figure 17-1. Demo of the LoginPage.jsp**  **Figure 17-2. Demo of the LoginPage.jsp** |

* **Code**:

|  |
| --- |
| **Figure 18-1. Code of the LoginPage.jsp**  **Figure 18-2. Code of the LoginPage.jsp** |

## **Login Page Servlet – Registration Java Server Page**:

* Login Servlet is the Java class. The LoginPage after verify the username and password will call to LoginServlet.java to access the Register Form in the session of the user.
* After the user login to the website, them they could access to Register Page to make Member Registration.
* If the user want to register member, they have to finish all the text box in the data form. In the case that the user miss some fields, the browser will announce to the user.
* DEMO:

|  |
| --- |
| **Figure 19-1. Demo of the LoginServlet.java**  **Figure 19-2. Demo of the LoginServlet.java in the case that missing input** |

* CODE:

|  |
| --- |
| **Figure 20-1. Code of the LoginServlet.java Figure 20-2. Code of the LoginServlet.java**  **Figure 20-3. Code of the Registration.jsp**  **Figure 20-4. Code of the Registration.jsp** |

## **Store Java Servlet**:

* After the user finish the Data Form from the Registration member in the Registration page, the user clicks on the register button, which will lead them to the store servlet.
* In the Store, it will display a simple html page with the table of the data recorded in the data form that user filling before.
* DEMO:

|  |
| --- |
| **Figure 21. Demo of the Store.java** |

* CODE:

|  |
| --- |
| **Figure 22-1. Code of the Store.java  Figure 22-2. Code of the Store.java** |

## **JavaBean**:

* To make the JavaBean can handle all of the properties in the Register form, they need:
  + JavaBean: Beans.java
  + Java Server Page: Forms.jsp
* In the Beans.java, it will declare all of the properties in the data form to attribute in the Beans class. It will generate all of the getter/setter of the properties.
* In the Forms.jsp, this is the page to show the change of the properties in the process using JavaBeans.
* DEMO:

|  |
| --- |
| **Figure 23-1. Demo of the JavaBean**    **Figure 23-2. Demo of the JavaBean** |

* CODE:

|  |
| --- |
| **Figure 24. Code of the Form.jsp  Figure 25. Code of the Beans.java** |

# **References**

1. Materials: Web Application Development – Dr. Nguyen Van Sinh
2. <https://stackoverflow.com/questions/15781902/javax-servlet-servletexception-cannot-be-resolved-to-a-type-in-spring-web-app>
3. <https://way2java.com/jsp/title-2-javabean-in-jsp-client-jsp-javabean-example/>
4. Bootstrap4: <https://getbootstrap.com/>
5. Drawing Use case: draw.io