**CHAPTER 5**

**IMPLEMENTATION AND TESTNG**

**5.1 Introduction**

The chapter describes the system implementation process, and system testing for the xxxxxxxxxxxxxx System.

**5.2 System Implementation**

There are two different sides of implementation for this project, they are the website side and the facial recognition side. For website, HTML, CSS, Javascript, and PHP programming languages will be used for coding. Then, software used are XAMPP and Adobe Dreamweaver. XAMPP is a distributor that provides server and database modules. Adobe Dreamweaver is an integrated development environment for website building. Next, as for facial recognition, Python programming language is used for coding. Then, integrated development environment software used is Microsoft Visual Studio.

**5.2.1 Software for System Development**

This section describes some of the software used for system development and its functions.

1. **Adobe Dreamweaver**

The purpose of using Adobe Dreamweaver is to build a website. Figure 5.1 shows the user interface of Adobe Dreamweaver. The user interface is showing the HTML code for account registration part. A list of folders and files are shown on the right-hand side user interface. The folder consists of multiple files such as HTML, CSS, Javascript, and PHP files. The output of these files will be displayed through a web browser and server-side files will be output by software provided by XAMPP distributor.

Text

Description automatically generated

Figure 5.1: Front-End Development in Microsoft Visual Studio

1. **XAMPP**

XAMPP is a software distributor that contains all the essential modules required to set up a server and to execute server-side coding. In accordance with XAMPP distributor, Apache server and MySQL database will be used in this project. Figure 5.2 and Figure 5.3 illustrates the XAMPP user interface and user interface of MySQL database respectively.

Table

Description automatically generated

Figure 5.2: XAMPP User Interface

Graphical user interface, application, table, Excel

Description automatically generated

Figure 5.3: MySQL Database User Interface

1. **Microsoft Visual Studio**

Microsoft Visual Studio was used to develop the facial recognition system. Whereas, the Python programming language is used to create code for facial recognition modules. The CV2 Python Library is used to provide the functionality needed to build facial recognition functions. Figure 5.4 shows the Microsoft Visual Studio user interface. The main screen shows the encoding of the FaceRecognizer.py file. Next, Figure 5.5 shows the file list for the facial recognition module of the project.

Text

Description automatically generated

Figure 5.4: Microsoft Visual Studio User Interface

**5.2.2 Functional Module Development**

This section describes the development of functional modules in a system. Program code is provided to aid clarification.

1. **Account Registration and User Login Module**

Figure 5.6 and Figure 5.7 show the server-side coding and the user interface of the account registration.

Text

Description automatically generated

Figure 5.6: Account Registration Source Code

Based on Figure 5.6, PHP programming language is used to code the account registration process. First of all, the code is written to receive a username and password input which will be passed from HTML and store it into $username and $password variable. Then, a SQL INSERT string is constructed by passing the value of $username and $password. Finally, the SQL INSERT query will be executed. If the result returned is true, then the data has been successfully stored in MySQL database.

Graphical user interface, application

Description automatically generated

Figure 5.7: Account Registration User Interface

Next, Figure 5.7 shows the user interface of account registration page. Generally, the user interface only consists of input box which takes the username and password input, and two buttons. Register button will execute the code as illustrated in Figure 5.6.

1. **Registration and Recognition Module**

*Program code screenshot*

*Explanation*

*User interface screenshot*

*Explanation*

c. xxx Module

*Program code screenshot*

*Explanation*

*User interface screenshot*

*Explanation*

**5.3 Testing**

In this section, a test will be carried out to assess the functionality of each module. A User Acceptance Test (UAT) method is utilized to perform testing.

**5.3.1 Account Registration and Login**

Table 5.1 show the test case for Account Registration and Login module. There are total of 3 test case for this module. The purpose of this test is to verify whether the administrator is allowed to register for an account, login into the system, and whether the system will restrict login if an incorrect credentials is entered. Table 5.1 shows that all the three (3) tests had passed the test.

Table 5.1: Test Case for Account Registration and Login Module

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Module: Account Registration and Login** | | | | |
| Test Case ID | Description | Expected Result | Actual | Pass/Fail |
| M1-1 | To check whether administrator can register for an account | The user should be able to create for an account | The user has successfully created for an account | Pass |
| M1-2 | To check whether a administrator can login into the system | The user should be able to login into the system | The user has successfully logged into the system | Pass |
| M1-3 | To check whether the system will restrict login whenever a wrong credential is entered | The system should restrict login when an incorrect credentials has been entered | The system restricted the login when an incorrect or no credentials has been entered | Pass |

**5.3.2 xxxxxxxxx module**