**CHAPTER FOUR**

**IMPLEMENTATION AND TESTING**

**Preamble**

This chapter presents the programming language that is being used to implement the system, the system’s hardware and software requirements that would make the system to function effectively. In this chapter, we also present the user documentation that will be used for training users. The system modules as well as the different interfaces of the system are also presented.

**4.1 Implementation**

The implementation technique of use is majorly determined by the developer and the framework that was established in previous sections. Thus, the overall mode of implementation is discussed below:

**4.1.1 Implementation Model**

**Bootstrap Framework**

Bootstrap is a sleek, intuitive, and powerful, mobile first front-end framework for faster and easier web development. It uses HTML, CSS, and JavaScript. Bootstrap was developed by Mark Orto and Jacob Thornton at twitter. It was released as open source product in August 2011 on GitHub. Bootstrap was used because of the following features:

* **Mobile first approach:** Bootstrap 3 framework consists of mobile first styles throughout the entire library instead of in separate files.
* **Easy to get started:** With just the knowledge of HTML and CSS, anyone can get started with Bootstrap. Also, the Bootstrap official site has a good documentation.
* **Responsive design:** Bootstrap’s responsive CSS design adjusts to Desktops, Tablets and mobiles
* Provides a clean and uniform solution for building an interface for developers.
* It contains beautiful and functional built-in-components which are easy to customize.
* It also provides web based customization.
* It is an open source.

**JavaScript**

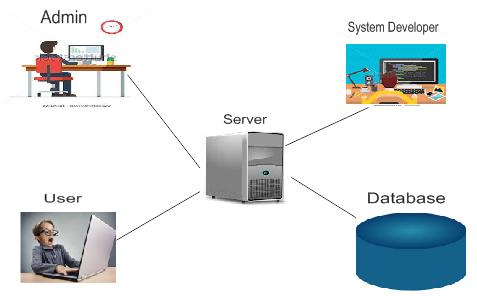
JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementation allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

**Apache Wampserver:** This is a local server on my PC which is used to simulate the client server architecture implemented by the system by utilizing my personal computers home port number

**Sublime Text Editor:** This is used for writing HTML and CSS.

**4.1.2 Deployment Diagram**

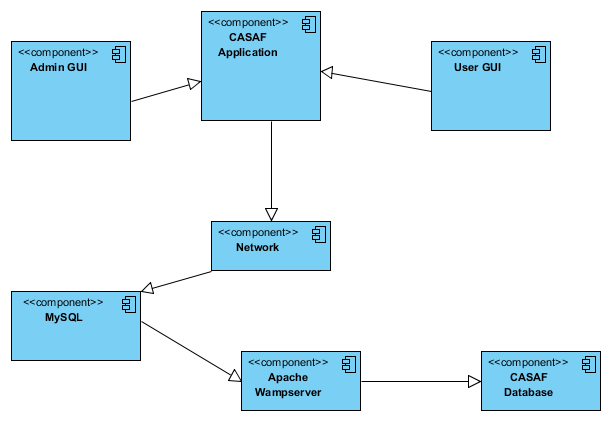
The deployment diagram describes the physical deployment of information generated by the software program on hardware components. The deployment diagram describes the physical deployment of information generated by the software program on hardware components. The figure below shows the deployment diagram of the Customer Advisory System on Automobiles Faults

****

**4.1 Deployment Diagram of the CASAF system**

This diagram shows the different software applications and the software that are associated with a system.

The figure below shows the component diagram of the CASAF which indicates that the system has two (2) main interfaces which are: the admin interface and the users(members) interface. These interfaces are dependent on the logic in the CASAF system for their functionalities. The CASAF system pulls data that is to be displayed on these interfaces from the (Wampserver) through over a network. Also, the data mining module analyzes the data from the Wampserver.



**Fig 4.2 Component diagram of the CASAF system**

**Description of the CASAF system Component Diagram**

|  |  |
| --- | --- |
| **ELEMENT** | **DESCRIPTION** |
| User interface | This component is part of the CASAF system. It is only meant for the visitors of the system |
| Admin Interface | This interface is meant for the admin of the system only. Admin uses this interface to regulate threads and comments |
| CASAF Application | This is the system in its entirety. It houses all the interfaces in the system that are used to perform various functions in the system |
| MySQL | This is the database management system that is used for the proper management of the CASAF system database |
| CASAF Database | This is the collection of records stored in tables used to ensure the efficient running of the system |
| Apache Wampserver | This is a local server on my PC which is used to simulate the client server architecture implemented by the system by utilizing my personal computers home port number |
| Network | The Network connection is the means by which the CASAF application connects with the server to access and store data located on its database. |

**4.2 System Requirements**

System requirements describe the necessary hardware and software needed for a successful deployment of the system. The system could be implemented on a local area network (LAN) and the system requirements for the effective performance of the system are in 2 parts which are: Hardware and Software requirements. However, there is an important need to distinguish between the minimum and recommended requirements.

**4.2.1 Hardware and Software Requirement for the server**

Below are the hardware and software configuration required by the system in order to function properly.

**4.2.1.1 Hardware Requirements**

**Minimum Hardware Requirements**

* A desktop or Laptop computer
* A Pentium 4 or equivalent microprocessor
* 1024MB RAM and above
* 2048MB available disk space
* Keyboard and Mouse

**Recommended Hardware Requirements**

* A desktop or Laptop computer
* A dual core processor and above
* 2048MB RAM or more
* 4GB available disk space
* Internet connection
* Keyboard and Mouse

**4.2.1.2 Software Requirements**

**Minimum Software Requirements**

* Operating system: Windows XP and above
* PHP version 5.1
* Web browser, 2011 and later
* Bootstrap version 3.0 and later
* Wampserver version 1.7

**Recommended Software Requirements**

* Operating system: Windows 7 and above
* PHP version 5.1 and above
* Web browser: Mozilla
* Bootstrap version 3.0.3 and later
* Wampserver version 2.2 or later

**4.3 Testing**

Testing is done in order to ensure that the developed system operates in accordance to the design and also performs its functionalities as expected while also ensuring that it meets the customer specifications. It allows us to locate, investigate, and correct errors and bugs. Three levels of testing will be explained. They are:

* + 1. **Unit Testing**

This is the testing of program individual components, such as objects and methods, to see if they can work independently of other components. The components that are tested in this system are explained below with their respective screenshots:

**User Login –** A user inputs valid email and password he/she registers with to get access into the system

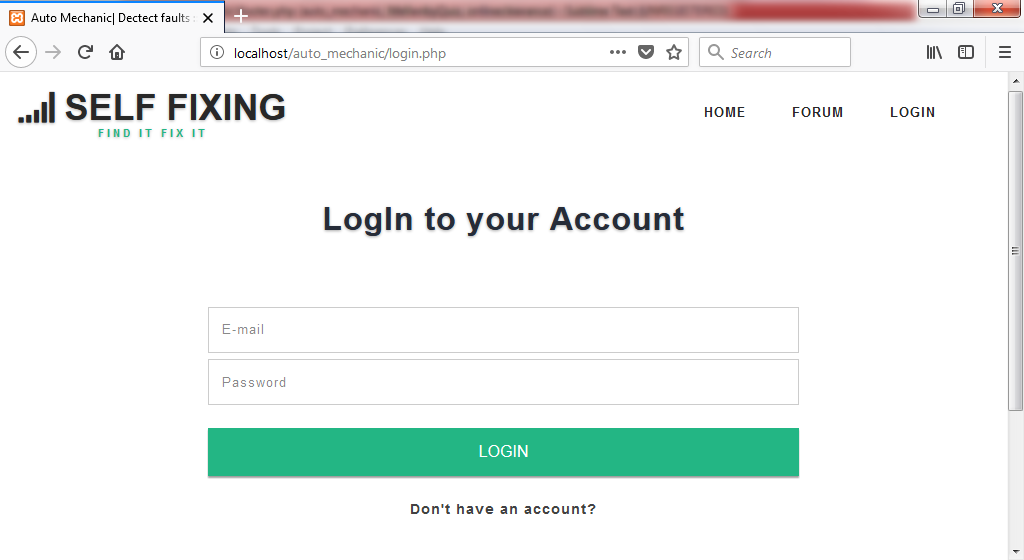


Fig 4.3 User login page

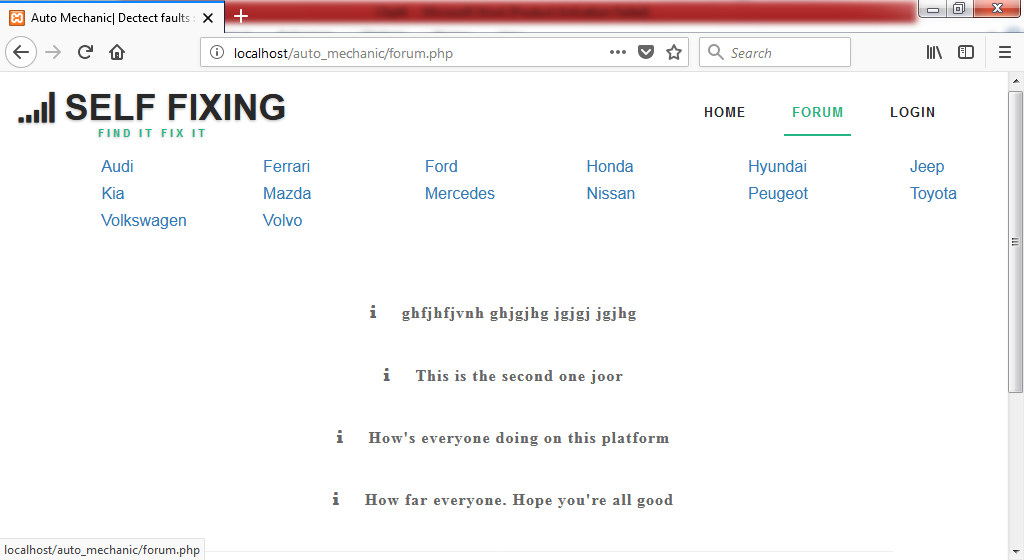
**Thread page –** This page shows all the created thread based on selected category

Fig 4.4 Thread page with categories

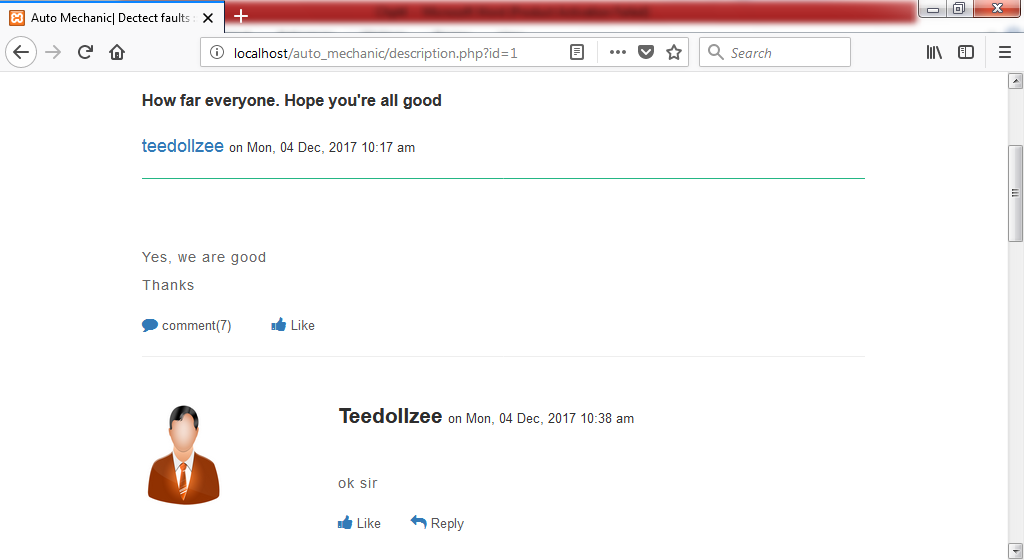
**Thread body page –** This show everything about a thread including the thread’s comments

Fig 4.5 Thread description page

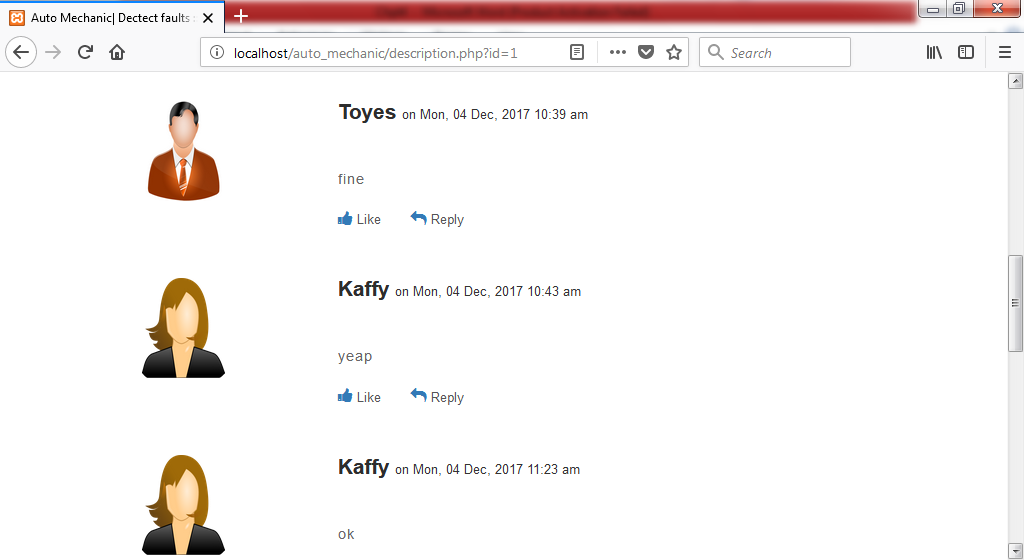
**Thread comment page – This shows all comments associated with a selected thread**

Fig 4.6 Thread’s comment page

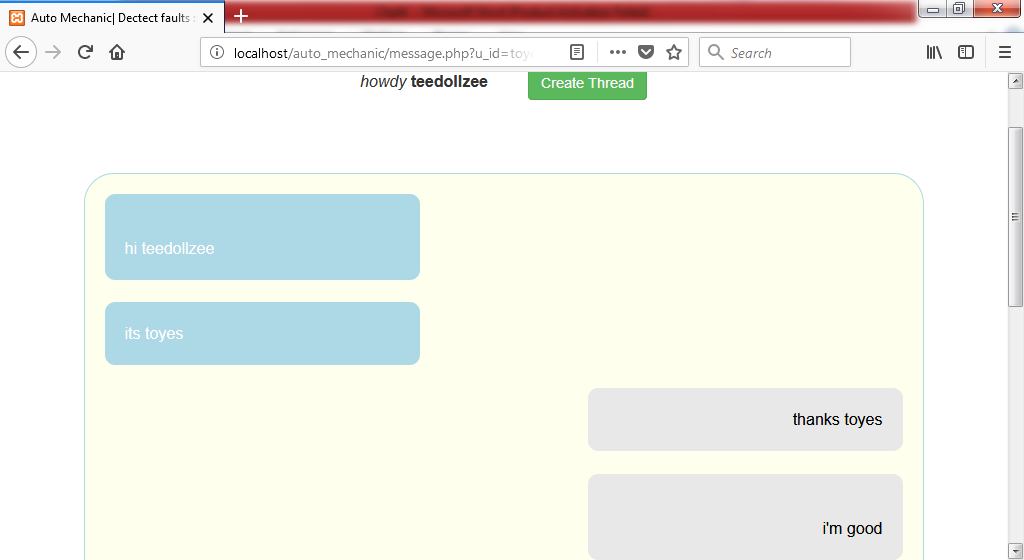
**Message page – This allows sending private messages between two users**

Fig 4.7 Message page

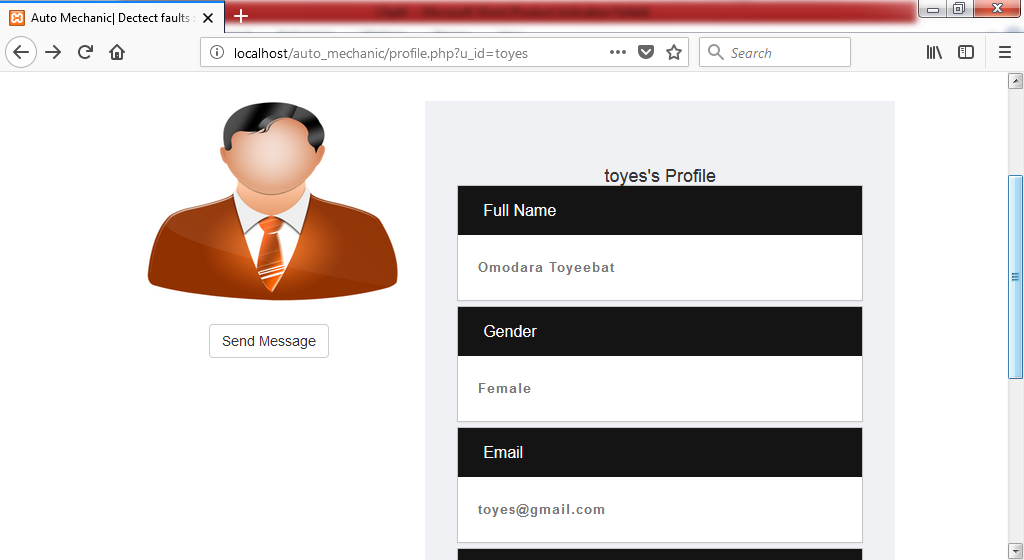
**Profile page –** A page where one user can check other users’ profiles

Fig 4.8 User’s profile page

**4.3.2 Integration Testing**

Integration testing involves the activity of exercising the interactions among different units by pulling together the different modules composing a system. It is characterized by involving different interacting units which have been generally developed by different programmers. Interface problems, missing functionalities, and unforeseen side-effects of procedure invocation are the major faults that can be detected by integration testing (Sommerville, 2011).

* + 1. **System Testing**

This is carried out during development ad it involves the integration of components to create a version of the system and then testing the integrated system. System testing checks if components are compatible, interact correctly and transfer the right data at the right time across their interfaces

* 1. **User Documentation**

**User Interface**

* Visit the URL on the computer.
* The latest created threads will be displayed, select any of the threads to view it details.
* Comments can be seen, including users’ profiles.
* Comments cannot be made and messages cannot be sent.
* Login to have access to more features in the system
* Comment on a thread by sending comments through the comment box
* View other user’s profiles by clicking on their user names
* Start conversation with a user by clicking on “message” button on the profile page
* Check unread messages by clicking on “inbox” o the menu bar

**Admin Interface**

* Visit the URL on the computer.
* View created thread by clicking on “view threads” on the menu bar
* Delete a thread by clicking on the “delete” button in front of a thread.
* View thread’s comments by clicking on the thread’ subject.
* Delete a thread’s comments by clicking on the “delete” button in front of a thread’s comments.