**CHAPTER THREE**

**ANALYSIS AND DESIGN**

**3.0 INTRODUCTION**

In this chapter we will takes an overview on the system design and the entire research work; we will present a skeletal approach to the design and analysis of the whole system.

System analysis has to do with the process of studying a procedure or a system in order to identify its goals and purposes and create systems and procedures that will achieve them in an efficient way.

**3.1    Analysis of the Existing System**

In this part of the world (Nigeria) the existing system for managing depression is still the use of human experts, this requires interactions between therapist and patience either directly in the same physical location or through mediums like video call, chat rooms, telephone calls and so on. The therapist ask the patience questions related to their symptoms gets feedback from the patience then make a diagnosis of the type of depression and make appropriate prescriptions, this system is effective in regions with adequate exposure to health education about depression and modern health care facilities, but less effective in region where people are less informed about depressive symptoms.

**3.1.1    Advantages of the Existing System**

The current systems of managing depression do have some advantages and they are:

1. The system does the job of diagnosing depressive cases and managing them properly.
2. The system encourages extensive interaction between therapist and patients.
3. The system keeps track of patients’ progress until fully recovered.
4. The system keeps adequate record of patients past diagnosis and prescriptions.

**3.1.2    Disadvantages of the Existing System**

The weakness of the existing system includes:

1. The existing system is not fully automated, that is, it still requires booking of appointments with the therapist, that makes it a first come first serve system and the implication of that is that patients will wait in line for the limited therapist available to get to them.
2. The existing system is not very efficient as it take more time and resource to get in a session with a therapist.
3. A lot of people in this part of the world (Nigeria) still see depression and other related cases of mental illness as something to be ashamed of and so may prefer to keep their situation private.
4. The issue of affordability is another key issue as many persons (especially those in the rural communities and low income level citizens) may not be able to afford the services of a human expert.

**3.2    Analysis of the Proposed System**

Most cases of depression are usually mild and mostly do not require a visit to a therapist so long as the patient knows a few things to do to manage the situation, in some cases just taking a walk for a few minutes can bring some improvement, even a change in diet can turn a situation around. The proposed system will use expert knowledge to diagnose the type of depression and the severity and give advice on how to manage the situation. The system will be web based and will not require any appointment with a human expert as the system will be fully automated and will utilize the knowledge of experts in the knowledge base to make diagnoses and suggest possible ways to manage the situation. The patient will not be required to sign up or login to use the system to maintain the user’s privacy as most users will prefer to remain anonymous.

**3.2.1 Advantages of the proposed system**

Some of the advantages of the proposed system are as follows:

1. Availability: the proposed system is always available, does not go on holidays or breaks all you need is an internet enabled device and access to the internet.
2. Accessibility: the system can be accessed by anyone from any location, at the same time, no queue or booking of appointment is required
3. Privacy: access to the system will be anonymous, anyone can take the test and get help without need to feel ashamed or have a feeling that someone is mining their data.
4. It will save the user a lot of time and money as the assessment time will be very fast and the platform will be totally free of charge.

**3.2.2 Disadvantages of the Proposed System**

The proposed system has two major setbacks and they are:

1. The system will be web based and so cannot be access without internet connection and internet enabled device.
2. The second drawback is that severe cases of depression like clinical depression that requires prescription of medication will not be managed by the system as such cases requires close monitoring by a human expert.

**3.3    Methodology**

The methodology adopted for the development of this system is the rule based expert system methodology.

A rule-based expert system is a system that contains set of rules that are used to describe certain patterns. Observed data are collected and evaluated using these rules. If the rules are logically satisfied, the pattern is identified, and a problem associated with that pattern is suggested. Each particular problem (symptom) might imply a specific treatment. These rules do not take into consideration the uncertainty and the impreciseness of human observed data and reasoning and real world knowledge that characterized by incompleteness, inaccuracy, and inconsistency. The rule-based approach uses IF-THEN type rules. IF-THEN rules take the following form: IF there is a flame THEN there is a fire.

**3.4    System Design**

The proposed system will have three basic components:

1. . A list of rules base, which is a specific type of knowledge base as it relates to depression.
2. . An inference engine, which infers information or takes action based on the interaction of input (user symptoms) and the rule base.
3. . A user interface through which input and output signals are received and sent. The user can be either the knowledge engineer inputting new symptoms into the knowledge base or the patient that is taking a test on the platform.

The proposed system design is as shown in fig 3.1 below

USER INTERFACE

DATA BASE/

RULE BASE

(Symptoms)

INFERENCE

ENGINE

RESULT AND

EXPLANATION

FACILITIES

**Fig 3.1 Design of proposed system**

**3.5    Use Case Diagram**

The use case diagram is shown below in figure 3.2. The knowledge engineer enters the symptoms into the knowledge base by login into the system, the patient visits the page takes the test and views his or her result.

Knowledge

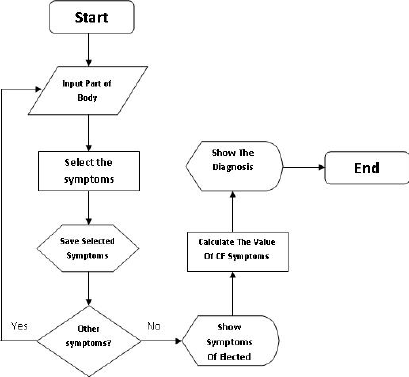
Engineer

Patient

Fig 3.2 Use case Diagram for Depression Management Expert System

3.6 Proposed System Flow Chart

Start



Login

Start Test

Yes

No

Admin

Select Symptom

View Symptoms List

Add Symptoms To Database

Save Selected Symptom

No

Yes

Other Symptoms?

Calculate Value OF Symptoms

Stop

Show Result

Fig 3.3 Proposed System Flow Chart.

**3.6 Software Requirements**

1. Operating system- Windows and mobile operating system is used as the operating system as it is stable and supports more features and is more users friendly.
2. Database MYSQL-MYSQL is used as database as it easy to maintain and retrieve records by simple queries which are in English language which are easy to understand and easy to write.
3. Development tools and Programming language- HTML and is used to write the whole code and develop webpages with cascading style sheet, java script for manipulating the document object model(DOM) and hypertext pre-processor (PHP) for sever side scripting.

**3.6.1 Software tools used**

The whole Project is divided in two parts the front end and the back end.

**FRONT END:** The front end is designed using of HTML, CSS, Java script

1. **HTML**- HTML or Hyper Text Mark-up Language is the main mark-up language for creating web pages and other information that can be displayed in a web browser.HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behaviour of HTML web pages.
2. **CSS**- Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation.CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed.
3. **JAVA SCRIPT**- JavaScript (JS) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side programming, game development and the creation of desktop and mobile applications. JavaScript is a prototype-based scripting language with dynamic typing and has first- class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the self and Scheme programming languages. It is a metaparadigm language, supporting object-oriented, imperative, and functional programming styles.

**BACK END-** The back end is designed using MySQL which is used to design the databases and PHP which is a scripting language for server side.

1. **MYSQL**- MySQL ("My S-Q-L", officially, but also called "My Sequel") is (as of July 2013) the world's second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius daughter, My. The SQL phrase stands for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for- profit firm, the Swedish company MySQLAB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL. For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr, and YouTube.
2. **PHP**- PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by RasmusLerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Pre-processor, a recursive backronym.PHP code is interpreted by a web server with a PHP processor module, which generates the resulting web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications. PHP is free software released under the PHP License. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

**3.7 Hardware Requirements**

A laptop, desktop, tablet or mobile device with at least 1gigabyte RAM and a functioning web browser e.g. Firefox and Chrome.

**CHAPTER FOUR**

**SYSTEM DESIGN AND IMPLEMENTATION**

**4.1 Introduction**

This chapter deals with the system implementation which is the actual development of the program and its documentation. The Hardware and Operating system requirement is also discussed here.

**4.2 Objective of the new system**

The objectives of the new system are:

1. To easily take a test (diagnosis of depression) and get the result and recommendation instantly.
2. Convenient user interface.
3. Easier end user interpretation.

**4.3 Landing Page (Home Page)**

The landing page or the home page is the first page a user sees when they visit the site. The landing page contains just two menu items: a link to admin(knowledge engineer) login and another link for the user(to take the test)

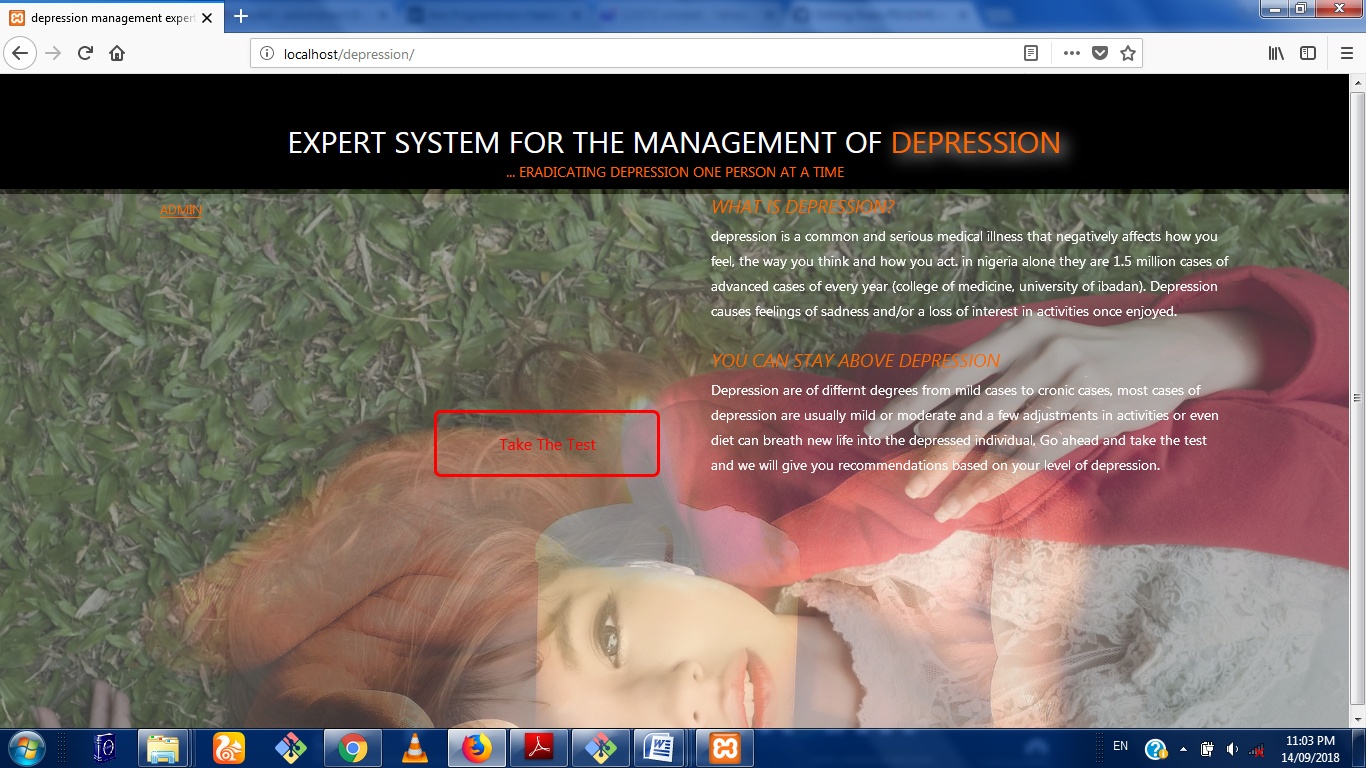


Fig 4.1: Main Menu

**4.4 Database Specification**

The database used in the design of this work is MySQL. A database with name of depression was created. It has two tables namely: admin and symptoms as shown in figure 4.2 below

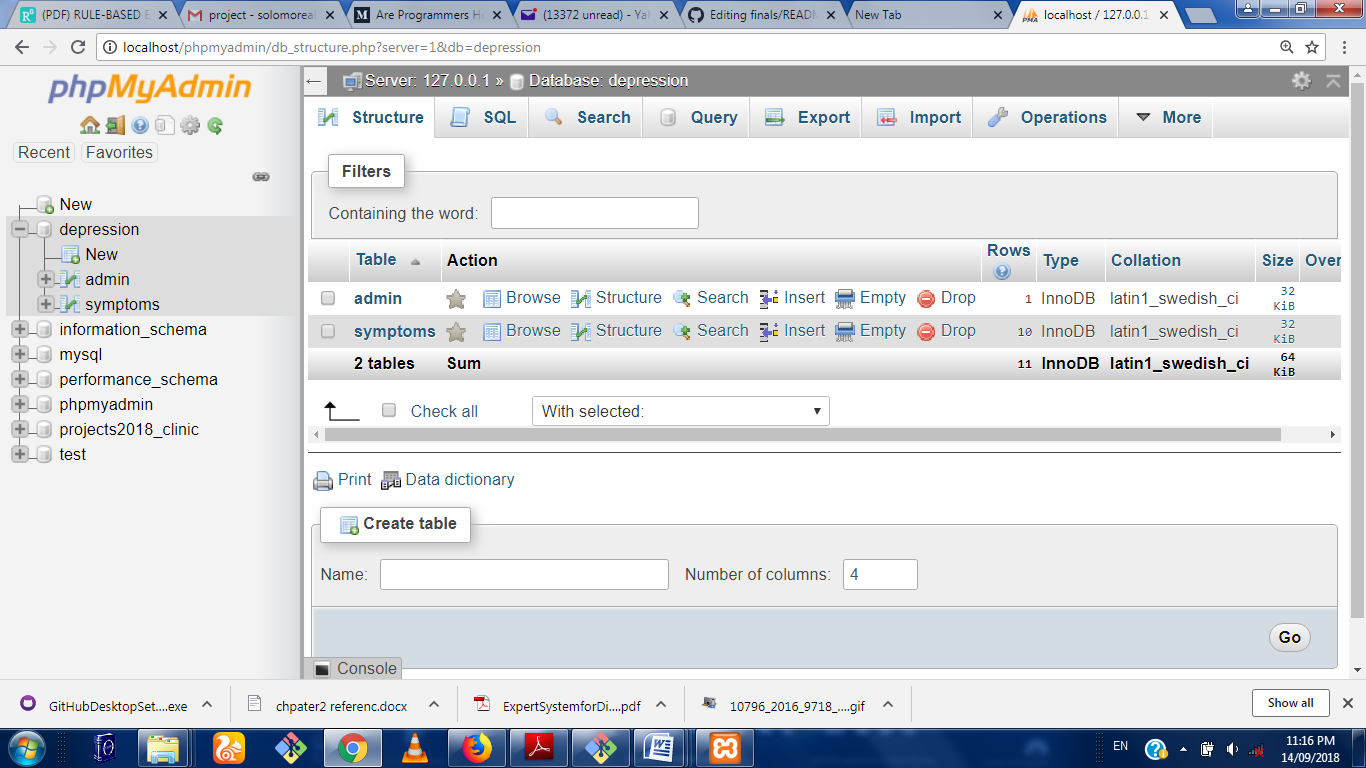
.

Fig 4.2 Database

The specification for the admin table is shown in fig 4.3. In this table, the attributes used are id, username, and password, with data types of integer and variable character for storing alpha numeric character.

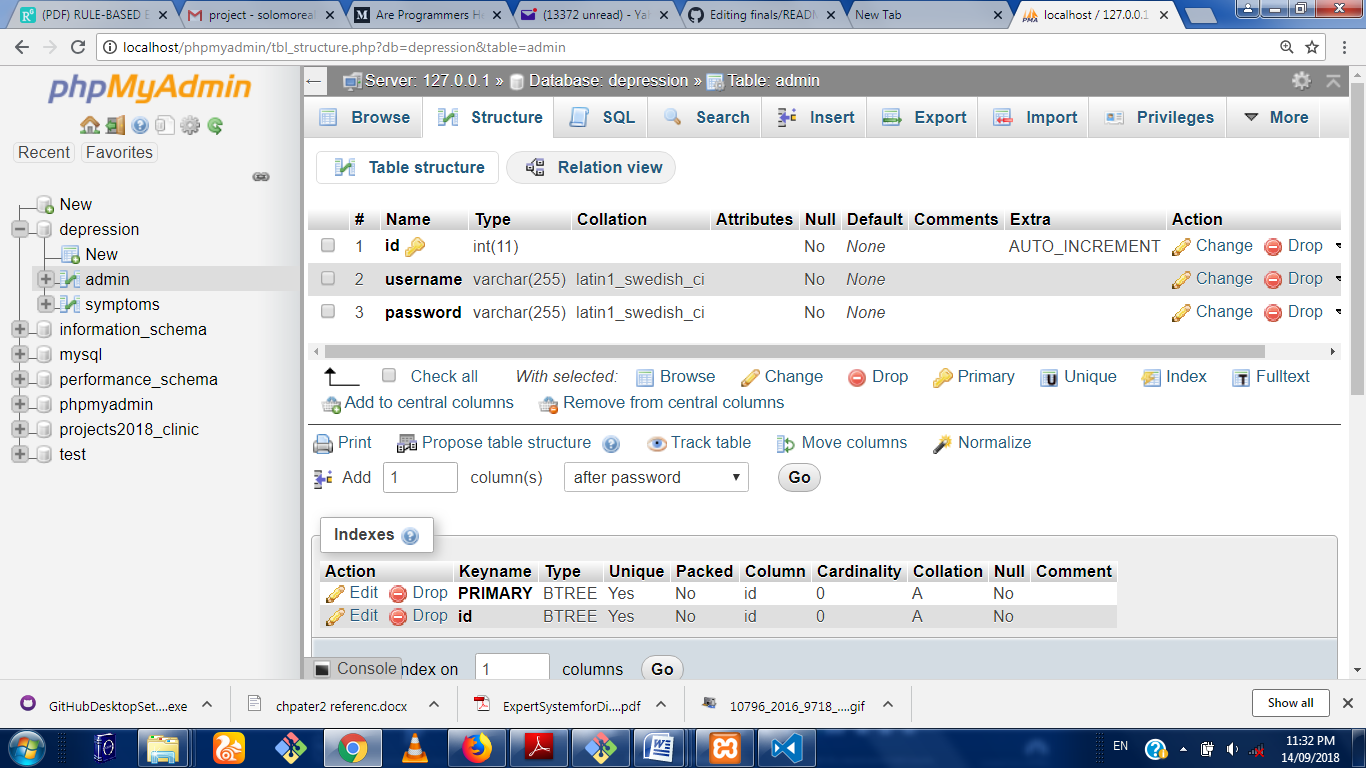


Fig 4.3

The specification for the Symptoms table is shown in Fig 4.4. In this table, the attributes used are id, and symptoms with data types of integer and variable character for storing alpha numeric characters.

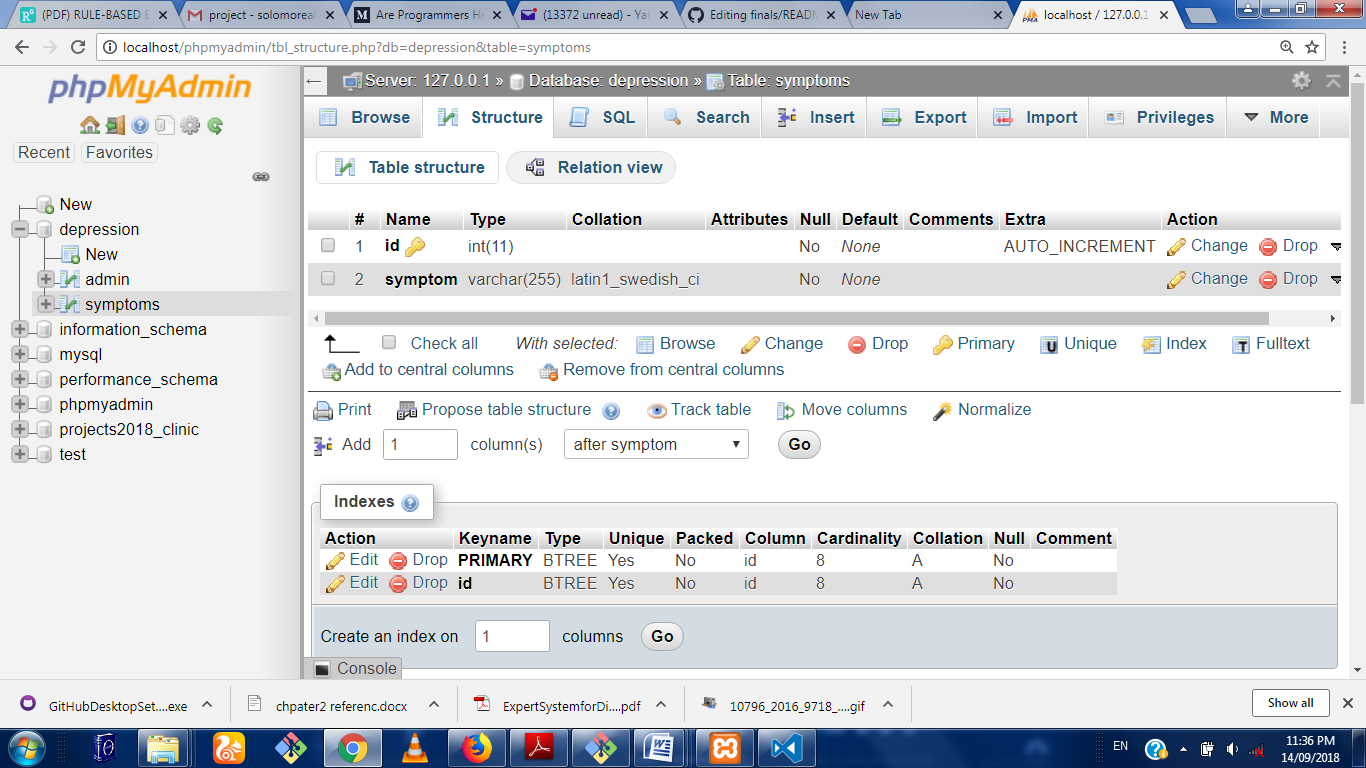


Fig 4.4

**4.5 Input/Output Screen/Format**

The input screen format shows the screen shot of all the input format in the program. The first input screen is the login page where the admin login to his dashboard.

This is shown in figure 4.5.

**4.5.1 Login Module**

This is the first point where the admin interacts with the program. The admin is required to enter his/her username and password and from here if authentication is successful, the admin dashboard.

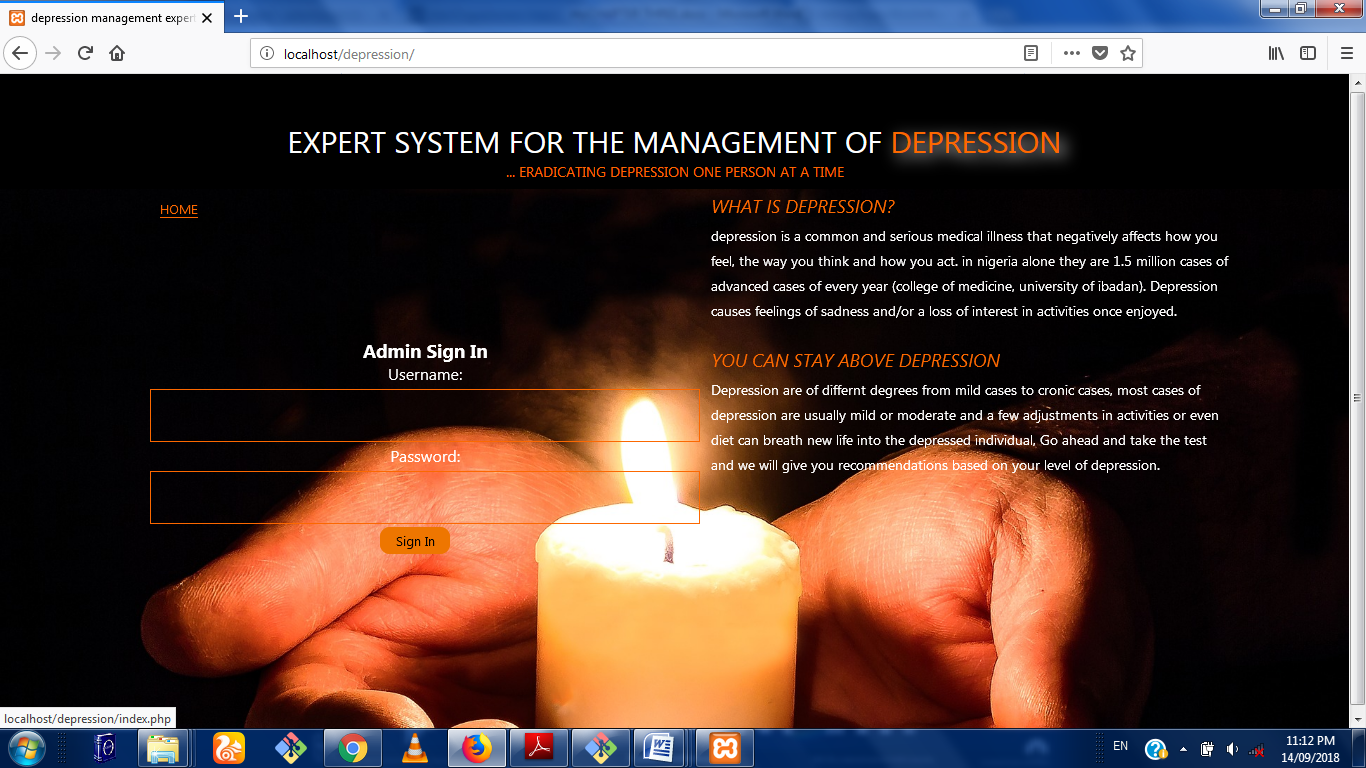
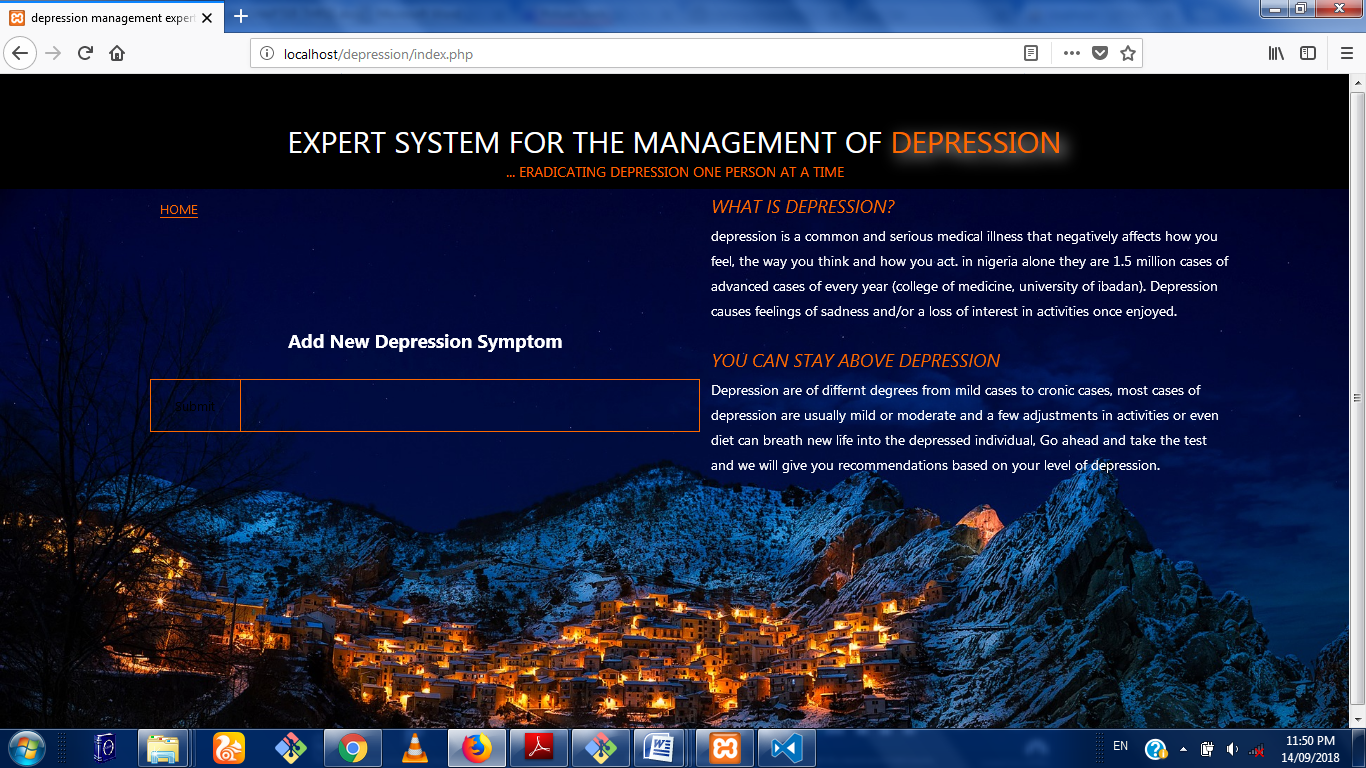


Fig 4.2: Login Module

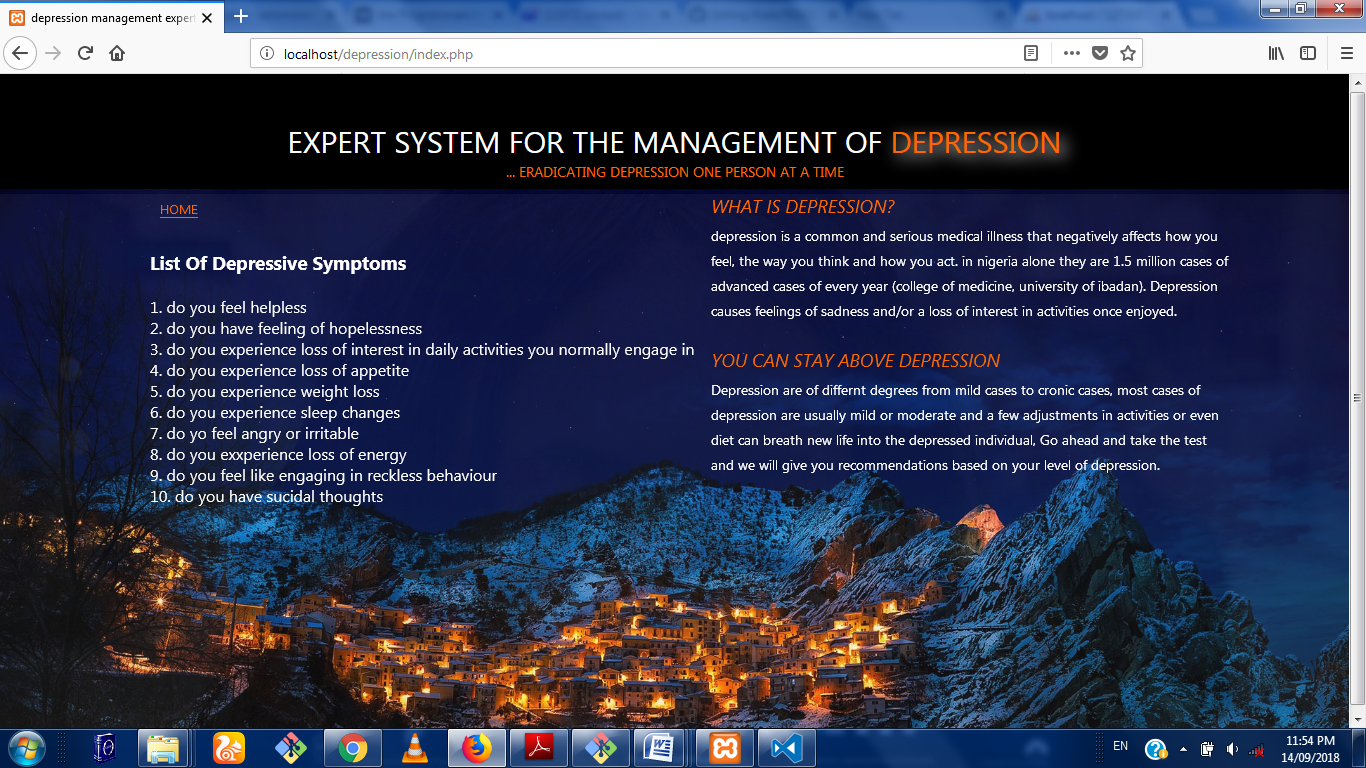
**4.5.3 Admin Symptom Input Module**

This is a page linked to the admin dashboard where the admin can add or create a new symptom by inputting on the required text field. Once the admin clicks on the submit button, a new symptom will be added to the knowledge base.

Fig 4.4 Admin Input Module

**4.5.4 Admin dashboard Symptom Output Module**

This is the region where the Admin get to view all the symptoms he/she has added to the data base

Fig 4.5 Admin symptom Output Area

4.5.5 Patient Diagnosis Module

This is where the patient takes the test by answering a few questions relating to depressive symptoms

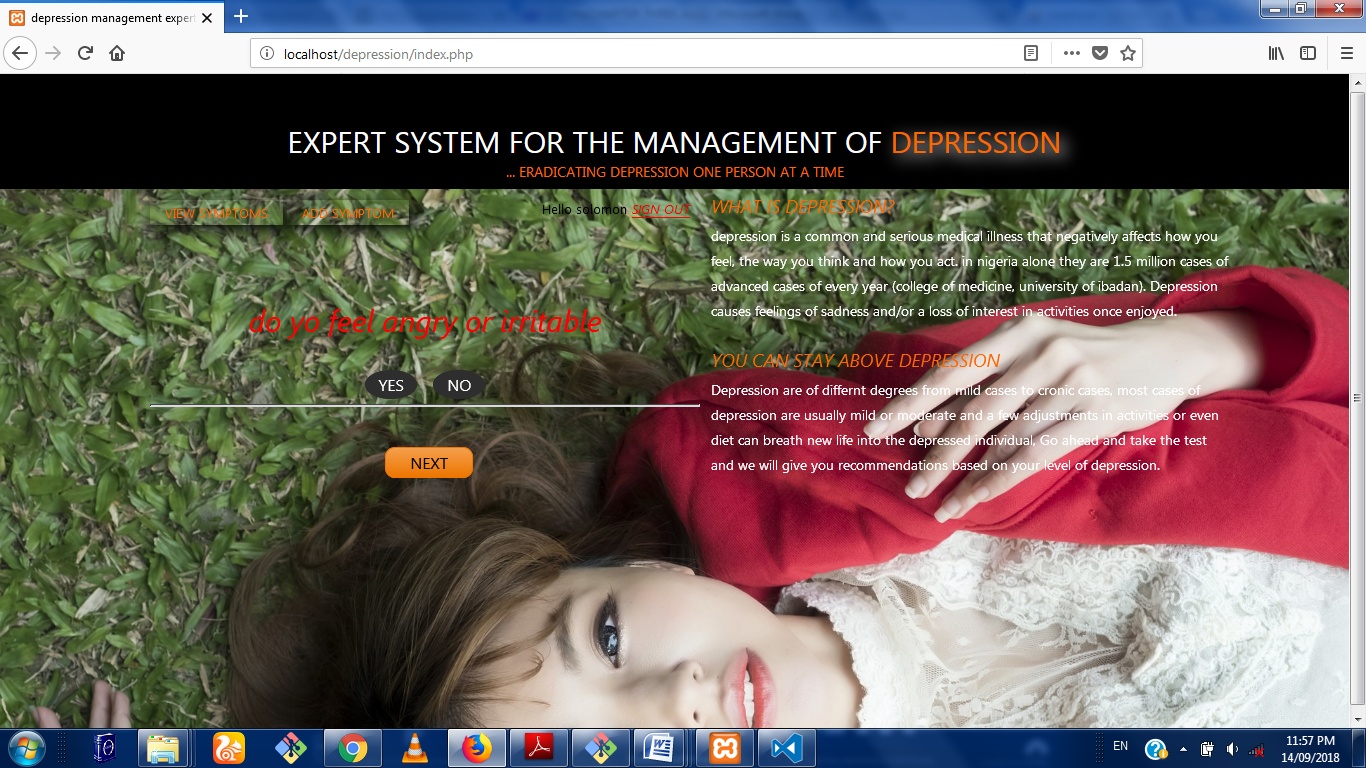


Fig 4.5 Testing Area.

**4.5.6 Result Display Module**

This is where the result of the patients diagnosis is displayed along side some expert advice



Fig 4.7 Result Area

**The 4.6 Hardware and Operating System requirement**

The hardware and Software system needed to run the program include

**4.6.1 Hardware Specifications**

1. Processor Name: Intel Dual Core / AMD
2. Processor Speed: 1.66 GHz
3. RAM: 1 GB
4. Hard Disk Capacity: 50 GB
5. Display Device: 14’ to 19’ Inch Monitor
6. Keyboard Type: PS2 or USB
7. Mouse Type: PS2 or USB

**4.6.2**  **Software Specifications:**

1. Language Used: HTML, PHP,CSS,JAVASCRIPT, SQL
2. Software Used: MySql,xampp Server ,Php
3. Operating System: Windows XP/ Windows 7/ Windows8 / windows10/Linux

**4.7 Software testing**

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements. The main purpose of testing is to detect errors and error-prone areas in a system. Testing must be thorough and well-planned. A partially tested system is as bad as an untested system. And the price of an untested and under-tested system is high.

The implementation is the final and important phase. It involves user-training, system testing in order to ensure successful running of the proposed system. The user tests the system and changes are made according to their needs. The testing involves the testing of the developed system using various kinds of data. While testing, errors are noted and correctness is the made.

The objectives of testing are:

1. Testing is a process of executing a program with the intent of finding errors.
2. A successful test case is one that uncovers an as yet undiscovered error.

System testing is a stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as per the user need, before the live operation commences. As stated before, testing is vital to the success of a system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. A series of tests are performed before the system is ready for the user acceptance test.

**4.7.1 Testing Methods**

System testing is the stage of implementation. This is to check whether the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. The candidate system is subject to a variety of tests: on line response, volume, stress, recovery, security and usability tests. A series of tests are performed for the proposed system is ready for user acceptance testing. The testing Steps are:

1. **Unit Testing;**Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.
2. **Integration Testing**

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. The objective is to take unit tested modules and builds program structure. All the modules are combined and tested as a whole.

1. **Validation**

At the culmination of the integration testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

a. The function or performance characteristics confirm to specification and are accepted.

b. A deviation from specification is uncovered and a deficiency lists is created.

c. Proposed system under consideration has been tested by using validation test and found to be working satisfactory.

1. **Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in a specific format. The output format on the screen is found to be correct. The format was designed in the system design time according to the user needs. For the hard copy also; the output comes as per the specified requirements by the user. Hence output testing did not result in any correction for the system.

1. **User Acceptance Testing**

User acceptance of a system is the key factor for the success of any system. The system under consideration is tested for the user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes whenever required.

**CHAPTER FIVE**

**SUMMARY, CONCLUSION AND RECOMMENDATIONS**

**5.1 Summary**

In this work, the web based depression management expert system was developed to ensure easy access to expert diagnosis of depressive cases and to get appropriate advice on how to manage the situation. It was proposed to work side by side the existing system that involves the direct interaction with human experts. The pros and cons of both systems were discussed and a suitable module was developed to aid in the depression management process. An application was developed to show the proof of concept and from the result; the system is noted to have performed well.

**5.2 Conclusion**

The depression management expert system was developed and has proving from a series of test to be an effective tool in the management of depressive cases.

**5.3 Recommendation**

In this work the expert system methodology was used but a more efficient system could be build that uses machine learning algorithms to learn from every patient that uses the system thereby improving itself and showing advanced intelligence.