**CHAPTER THREE**

**PROJECT METHODOLOGY**

**3.1 Introduction**

Court case filing system implementation phase involved decomposing the whole system into modules and defining the relationship between the constituent modules. Top down design approach was employed in this work. This involved dividing the system into sub-systems or modules and each subsystem being further divided into even smaller subs. This process of division is repeated until each module is sufficiently small enough to be conveniently coded (implemented) from scratch as an independent entity that performs a clearly defined operation.

The analysis and comparison of existing online filing systems of institutions was done. The critical analysis led to the adoption of the scheme of requirement specifications that highlighted the nature of the online court case filing system implemented in this project work. Court case filing processing system development is all about making court case filing processing very easy for the administrators of courts; plaintiffs and their lawyers would be able to file cases online at the comfort of their homes/offices. The performance evaluation made provided a deep insight into creating a virile web application that serves as a means to enhance speedy processing of the court cases, to eliminate delays related to manual court case filing and processing, and to provide security measure to check unauthorized access to case files. The process followed in the design and implementation of the online court case filing processing system is succinctly written in the following sections of this chapter.

**3.2 Requirement Analysis of the Case filing System**

This section reports the analysis of the programming language, tools and the relational database management system used in the implementation of the case filing system. The design phase of this project was carried out by the use of the following programming language and tools:

- XHTML 2.0

- CSS 2 and 3

- PHP 5.4

- MySQL 5.5

**3.2.1 Extensible Hyper Text Markup Language (XHTML 1.0)**

Hyper Text Markup Language is the language used in the web application to render the view/design for the client/end user’s web browser. It is the only language that the web browser normally understands. The latest version was used for this court case filing processing project, the latest version of HTML is 5 which is equivalent to XHTML 2.0 (Extensible Hyper Text Markup Language) was used in this project work. It is the markup language used in this project in rendering the PHP scripts written so the web browser can understand what is being written in the backend scripts (PHP codes/commands).

The web browser reads the HTML documents and composes them into visible web pages. HTML was used to get the images and objects embedded and to create interactive forms such as the ‘Admin registration form” for instance. HTML was used to create structured documents by denoting structural semantics for text such as the headings, paragraphs, lists, links, quotes and other items during the implementation of the website.

**3.2.2 Cascading Style Sheet (CSS)**

Cascading Style Sheet is the markup language used in the web application that makes up the styling and beauty of HTML. Cascading Style Sheets (CSS) is a style sheet language and it was used in this project for describing the look and formatting of the pages already written in XHTML, it was used to style the pages and interfaces. The version of Cascading Style Sheet used in this project work is 3.0; CSS 2 was used in the project for all layouts while CSS 3 was used in the responsiveness, forms and buttons of the project.

CSS was used to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. This was done to improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content. CSS was also used to allow the XHTML pages to be presented in different styles for different rendering methods, for instance, it was used to for mobile responsiveness of the web app developed, that is; allowing the online court case filing website page to display differently depending on the screen size or device on which it is being viewed, be it mobile phone, mini laptop, laptop or whatever.

**3.2.3 Hyper Text Pre Processor (PHP)**

Hyper Text Pre Processor is the backend programming language used for this web application. PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP 5.4 was used in the implementation of the project. It was used to accept users’ input, process it and perform some actions on it. e.g.: saving registered users and admin in the database and authenticating their passwords for them to login. PHP was used also used to interact with database tables using MySQL.

**3.2.4 MySQL (Structural Query Language)**

This is a widely used open-source relational database management system (RDBMS). MySQL is a popular choice of database programming for use in web applications. MySQL 5.5 was used in the implementation of this project work. It is the backend where all the data generated in the process of design and subsequent use of the court case filing system web application are kept.

MySQL workbench is a tool in MySQL 5.5 database programming language and it was used in drawing and linking different database tables. This tool was used to know the relationships between two or more tables/models in the design phase, even before the development phase of the application is being started. Once the number and details of tables needed are known, the workbench was used to generate the application database tables and the relationship between them; which gave a clear understanding of what is meant to be implemented.

**3.3 System Design**

In this section, the design of the court case filing processing system is presented. The design of the system are expressed here using the activity diagram, the data flow diagram, the use cases and the entity-relationship diagram.

At the beginning of the project implementation, collections and downloading of the applications (software) needed was done. The wireframe and planning of the application was done on paper. Collection of the objects needed for the database and models was then planned and drawn on paper. Then, the Entity relationship between the database tables was done on paper and then finally done on MySQL workbench to know the relationships between tables and table objects as shown in the diagram on the next page (Figure 3.1). The architecture of the court case filing system consists of the frontend developed (coded) with php and designed (styled) using HTML and the backend done with MySQL server. The front end presents an interface where the users can interact.

**3.3.1 Database Design**

The database is a very important part in developing a web based court case filing processing system. The database used in this system was developed using MySQL database, this stores the case file details, the user log in details, admin log in details, etc. MYSQL is an open source database management system developed by Microsoft. It stores data in its own format based on access jet database engine.

Database design includes the tables used in the system database. These following entities can be identified specifically for the system being developed. These tables are used to store the information about each entity. The data structures of the entities on the database are as shown in Table 3.1.

There are two types of users on this web app; the admin with some exclusive admin privileges and the users. The backend is where all users’ data (and every other kind of data of course) are saved for processing and layer retrieval. At the back-end, MySQL database was employed to implement the database management. MySQL work bench was used to draw the database tables; most of the tables have a one-to-one relationship.

There are five main tables created as shown in figure 3.2. The first one is the table for ‘cases, the details of each case being created by the admin is saved on to the table. Every course is given an id automatically once it is being created by the admin, course details include, case\_title, case\_status, course\_code, session and level as shown in figure 3.3. The second table is the ‘case\_category’ table, each category is given an auto id, and the category name is also on the table as shown in figure 3.4. The next table is the ‘type’ table; the table stores the details of every case type created by the admin user. Every case type created is given an auto id, every type is meant to belong to a category, so the category\_id is also included with the type on the class table as shown in figure 3.5.

The user’s table holds all information of the registered users; there two categories of users; the admin user, who approves/decline the cases and create the other entities of the web app as shown in figure 3.7. The tables designed and created can be viewed on phpmyadmin via the localhost.

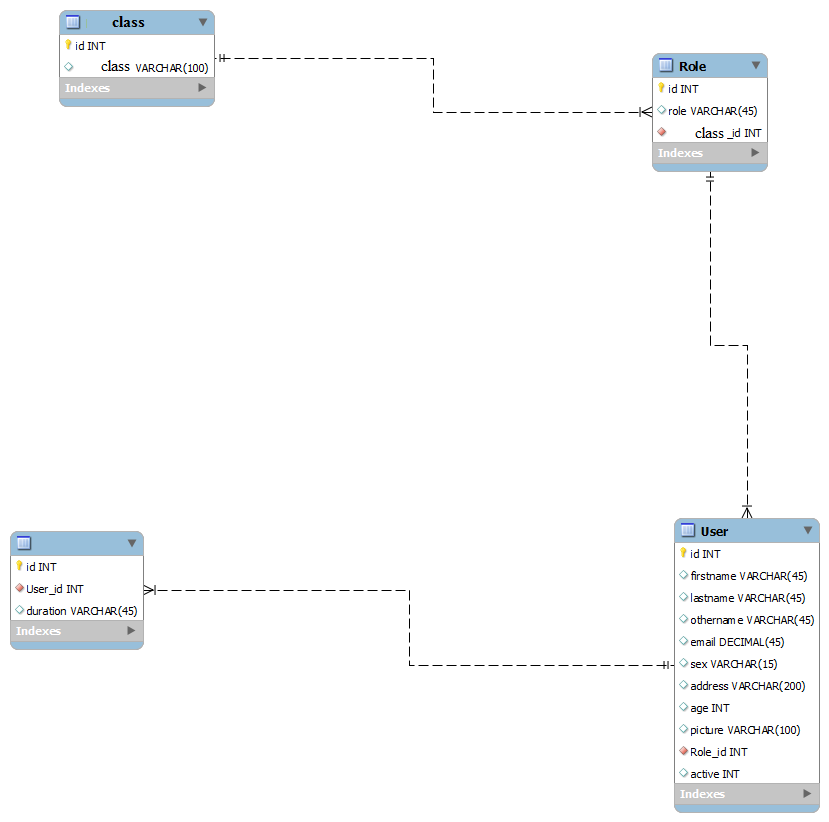


Figure 3.1 Entity Relational Diagram

Table 3.1 Database Table elements and datatypes

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Data type** | **Description** |
| User\_id | int(11) | user identification |
| user\_firstname | varchar(18) | user first name |
| user\_lastname | varchar(20) | user last name |
| user\_password | varchar(13) | user login password |
| user\_gender | varchar(7) | user gender |
| user \_phoneno | int(11) | user phone number |
| user \_emailadd | varchar(30) | user email address |
| Case\_id | Int(11) | Case identification |
| user \_d\_o\_b | Date | user date of birth |
| admin\_id | int(11) | admin identification |
| admin \_firstname | varchar(18) | admin first name |
| admin \_lastname | varchar(20) | admin last name |
| admin \_password | varchar(13) | admin login password |
| admin \_gender | varchar(7) | admin gender |
| admin \_phoneno | int(11) | admin phone number |

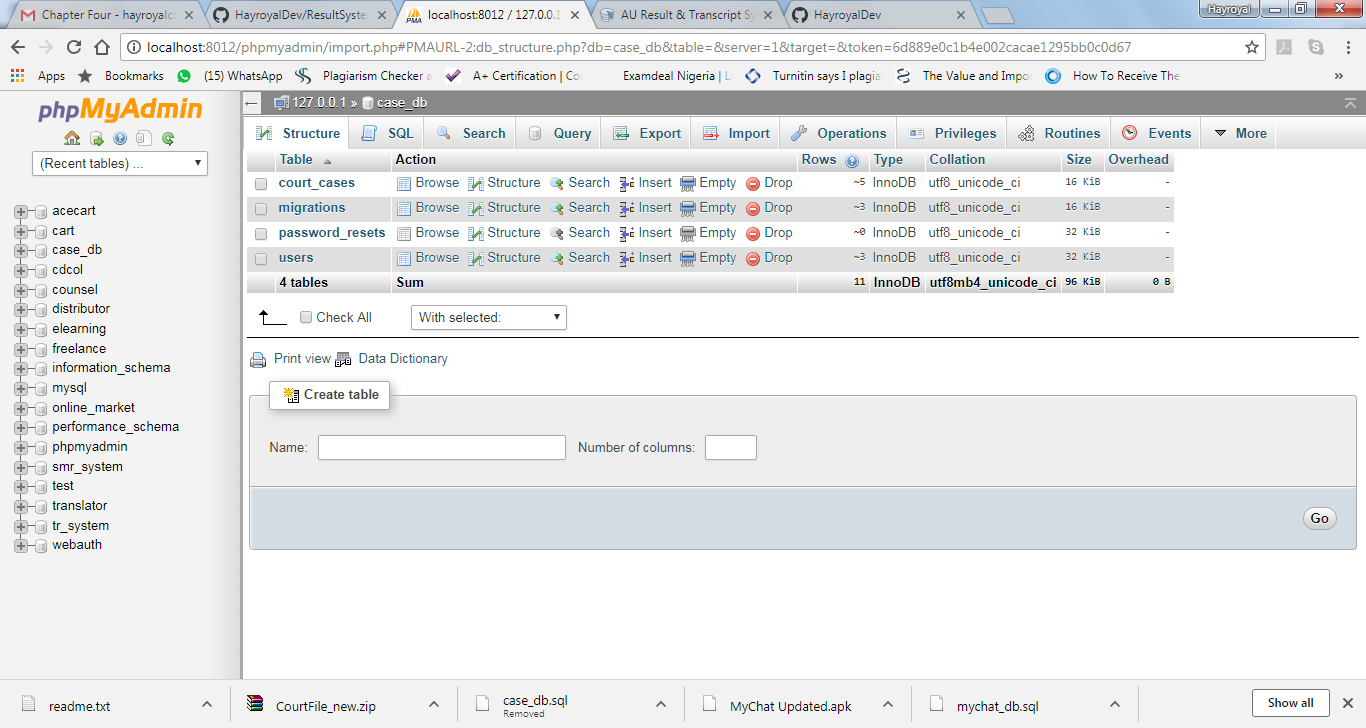


Figure 3.2 All the tables created for the project

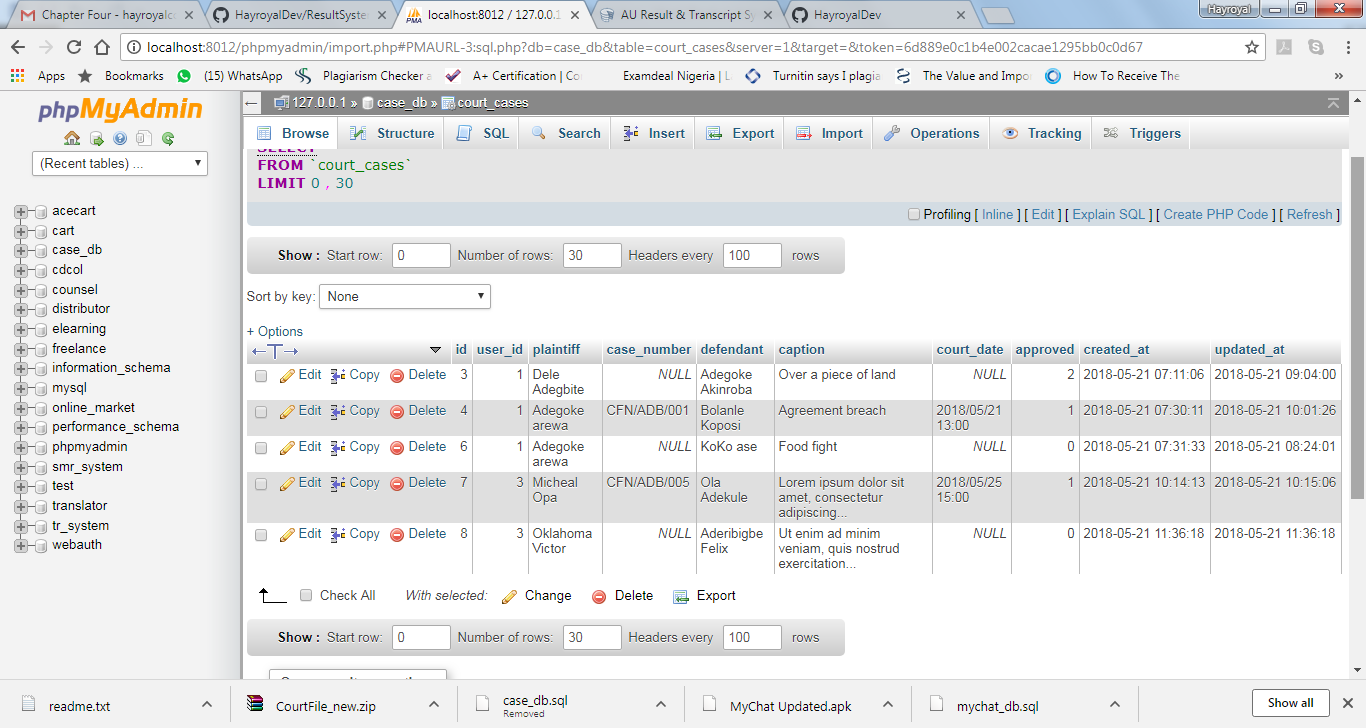


Figure 3.3 Cases Table

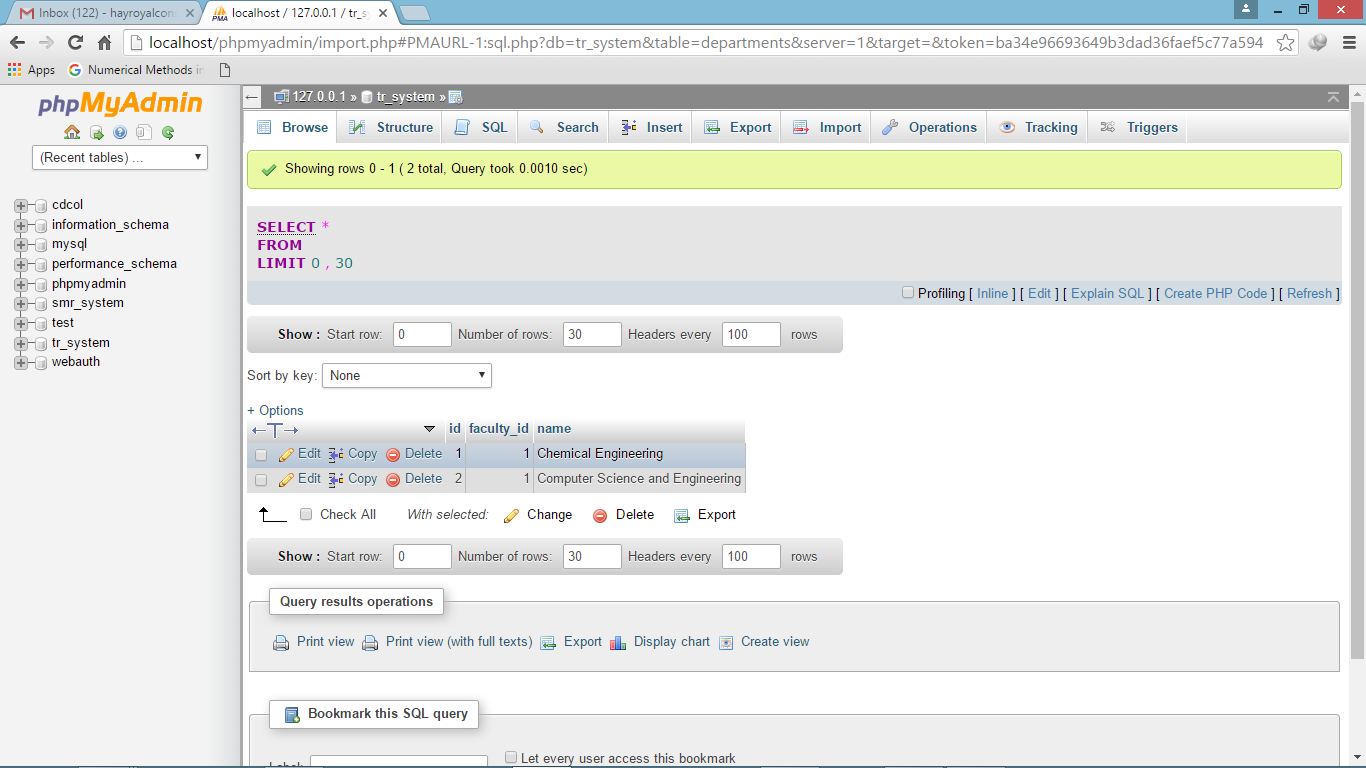


Table 3.4 Category Table

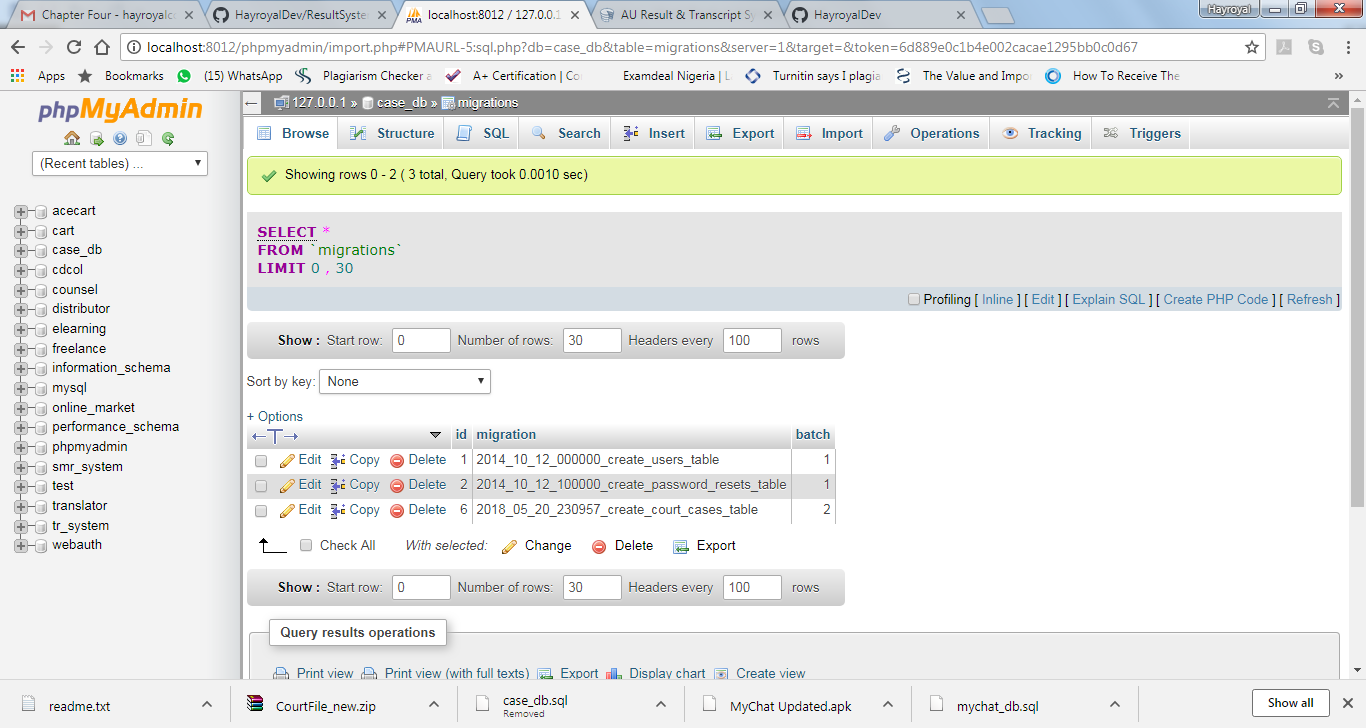


Figure 3.5 Type table

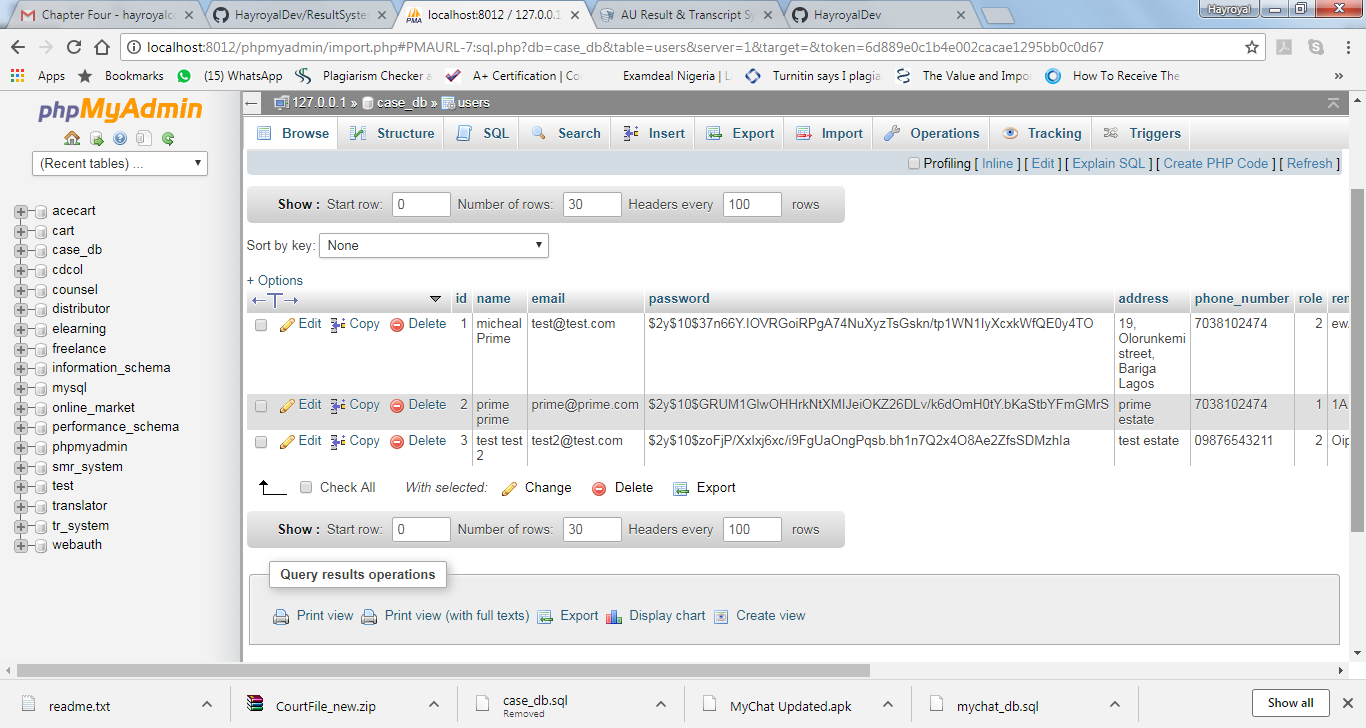


Figure 3.7 Users table

**3.3.2 Activity diagram**

Activity diagrams are graphical representations of workflows depicting stepwise activities and actions with support for choice, iteration and concurrency. In the unified modeling language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

The activity diagram shows the visual representation of the process flow. In this case filing system, it shows how a user will navigate from one point to another on the website in order to getting the task of filing a court case done. The web app was developed with skilfully linked pages that followed a very orderly sequence; the landing page is the home page (index page), on the home page, the user aim is to approve/decline a court case and create other entities of the application software as admin. The user creates the cases and its full description. For a user to be successfully logged in, the provided credentials must match the one stored in the database at the point of registration by the admin. If the credentials are correctly supplied, the user would be taken to the home page, else he/she would be taken back to the ‘login’ page to supply the correct and matching username and password. This and more are more elaborated discussed in chapter four of this final year project report. Figure 3.8 on the next page clearly shows the data and process flow in the online case filing system.

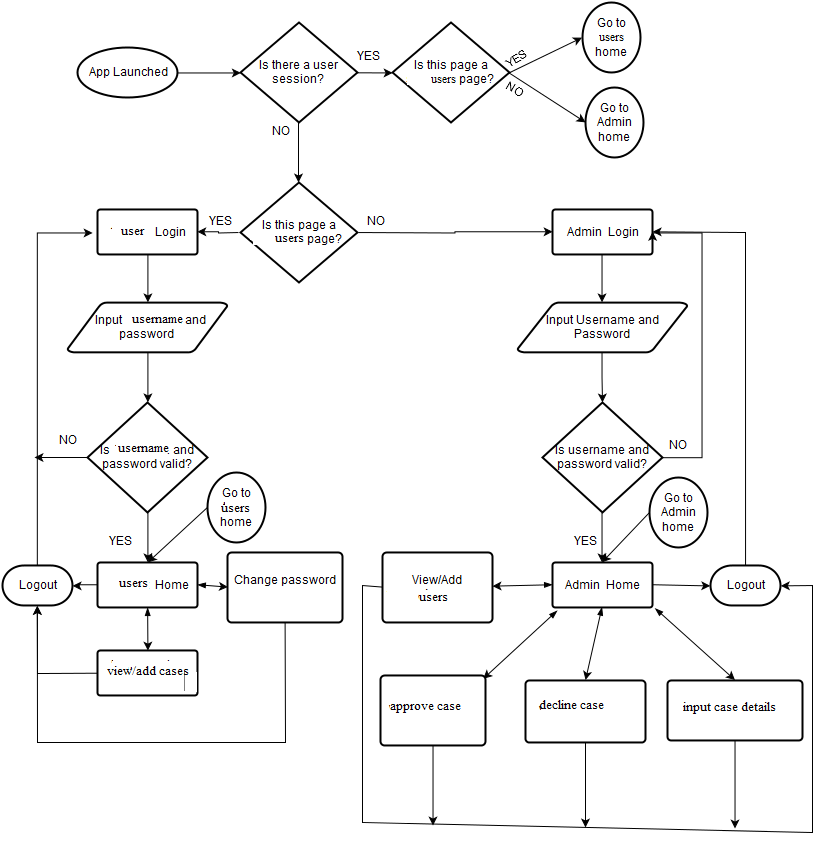


Figure 3.8 Program activity diagram

**3.3.3 Data Flow Diagram (DFD) for the System**

A data flow diagram (DFD) uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among the functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchy represents various sub-functions. The data flow diagram depicted in figure 3.9 below shows the relationship among the entities in the court case filing system. The entity “USER” can file and access his cases after he or she gains access to the system. The unregistered user can only view the login/signup page with the system prompting the user to sign up. The entities “ADMIN” can approve or decline cases filed by users. The entity “SYSTEM” is responsible for authenticating the users of the system and also provides the timing functionality for the users’ session and setting the default password for the users of the system when it is needed during password reset. The system logs off a user upon expiration of the login session.

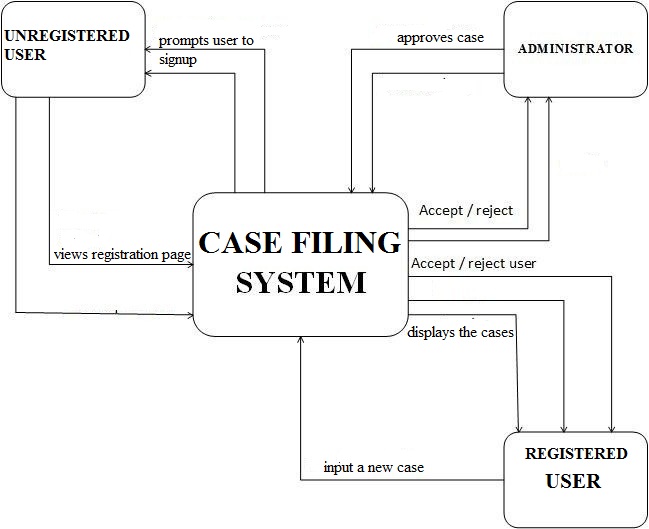


Figure 3.9 Data flow diagram

* 1. **Procedure Phases and Time Frame**

The implementation of the project was divided into the following phases:Requirement phase, Analysis Phase, Design phase, Implementation and testing phase

* Requirement phase: here, a complete description of the behavior of the system was done. The interaction users would have with the software was clearly mapped out for clear coding.
* Analysis Phase: Formal enquiry was carried out in order to identify a better course of action to develop the system, thoughts were shared, and online resources were perused for clear information.
* Design phase: The Graphical User Interface for the application was developed. This was done using the programming languages (HTML, CSS and MySQL) fully discussed in section 3.2 of this project report.
* Implementation Phase: This is the part of the process where the software engineering would be actually done. The coding from scratch using PHP (fully discussed in section 3.2 also). Scripts for different modules were written and fully tested okay. This phase is in progress.

The project will be completed in six months (first and second semesters). It is divided into two segments completed in each semester. The first segment is the gathering of relevant information and other resources for the successful completion of the project, this entailed the preparation of the chapter one (introduction) and two (review of literatures on past related works) of the project report. The second segment involved the actual implementation and discussion of the results attained.