**DESIGN AND IMPLEMENTATION OF STUDENTS’ INTERNET VOTING SYSTEM FOR THE DEPARTMENT OF DATA AND INFORMATION SCIENCE, FACULTY OF MULTI-DISCIPLINARY STUDIES, UNIVERSITY OF IBADAN**

**BY**

**ADEDIGBA SYLVESTER SEUN**

**224148**

**CHAPTER FOUR**

**SYSTEM DESIGN**

**4.1 Introduction**

System design is the process of defining, developing, and designing systems that satisfy the specific needs and requirements of an organization. It defines elements of a system such as modules, architecture, components, interfaces, and the data for a system based on the specified requirements. The previous chapter discussed the detailed analysis of the existing system, constraints, and a feasible alternative chosen. This chapter deals with the design of the proposed system, objectives, description, interfaces, database design, data dictionary, and system security.

**4.2 Objective of the New System**

The main objective of the new system is to design and develop an easy-to-use and interactive internet-based student voting system for the Department of Data and Information Science, Faculty of Multi-Disciplinary Studies, University of Ibadan. This system provides a remote mechanism that enables students to vote at their convenience and saves voters time and energy from long queues and complex voting processes and procedures.

**4.3 General Description of the New System**

The new system to be developed is a web-based internet voting system that would be hosted on a web server and be accessible from various web browsers. The new system will provide a platform for simplifying the electoral process for DDISSTA that employ voting in selecting representatives. It will offer a voter registration form for students where students will register with their matriculation number and be permitted to log in as either students or aspirants. A password will be generated and sent to every registered student’s email to log in, read manifestos and cast a vote. The new system will also provide a platform where voters and aspirants can interact and thus aspirants perform their campaigns. The system will compute and provide the election results in easy-to-read summaries and graphs for all of the posts, with the winner determined by the highest number of votes in each post. The design of the new system is divided into four (4) interfaces such as admin, aspirant, voter, and election observer.

**4.3.1 Use Case Diagram of the New System**

Add Post

Message

Pre-election Phase

Election Results

Post-election Phase

Create User

Change Password

Forum

Edit Profile

Upload Passport

Send Report

Vote

Logout

Aspirant

Admin

Voter

Observer

**Figure 4.1: Use Case Diagram of the New System**

**(Source: Author’s Concept)**

**4.3.2 Detailed Function of the New System**

The new system procedures involve the use of web applications in undertaking voting activities and will run on any browser from a device that is internet enables. The database used by the new system is MySQL which is the most popular open-source relational database management system in the world. It stores data in tables that are made of columns and rows. Using SQL statements, users can manipulate, define, control, and query data. The four (4) interfaces made up of the new system are:

1. **Admin Interface:** This is the user interface for the DDISSTA electoral council. It enables the user to perform pre-election and post-election activities such as registering aspirants, uploading eligible voters, creating users, setting election duration, attending to messages, releasing election results, and auditing.
2. **Aspirant Interface:** This is the aspirant interface to interact with the system. It enables the user to add a manifesto, cast a vote, add posts to the forum, and view election results.
3. **Voter Interface:** This interface enables the voter to cast a vote, send a message to the electoral council, read the aspirant’s manifesto, interact with the aspirant through the forum, and view election results. All voting choices are confidential and cannot be linked to the voter and results will be hidden until the election ends.
4. **Election Observer Interface:** This interface enables election observers to monitor the conduct of the election on election day and send a comprehensive report to the electoral council to deliver credible fraud-free elections.

**4.4 Design Technique**

The top-down approach was adopted in the design of the new system. It is an approach where the complete system is divided into smaller sub-systems with more details. Each part again goes the top-down approach till the complete system is designed with all minute details.

**4.4.1 Data Flow Diagram (DFD)**

DFD is a graphical technique that maps out the flow of information through a system and gives insight into the inputs and outputs of each entity and the process itself.

fill registration form upload eligible voters

Voter/Aspirant

Admin

Voting Process

login election results

**Figure 4.2: Level 0 Data flow Diagram of the New System**

**(Source: Author’s Concept)**

Voter

Observer

Aspirant

Admin

Login

Voting Process

Logout

Voting Database

**Figure 4.3: Level 1 Data Flow Diagram of the New System**

**(Source: Author’s Concept)**

**4.4.2 System Flowchart**

A flowchart is a graphical representation widely used to depict the flow of algorithms, workflows, or processes. It is used to document, study, plan, detect problems, improve, and communicate complex processes in clear, easy-to-understand diagrams.

Figure 4.0 shows the system flowchart of the proposed system

Start

Register Voter

Is voter registered?

Display Error Message

Login

Correct Username & Password

Invalid Username or Password

Admin Page

Voter Page

Aspirant Page

Observer Page

Logout

Stop

No

Yes

No

Yes

**Figure 4.4: System Flowchart**

**(Source: Author’s Concept)**

**4.5 Database Design**

MySQL is an open-source Relational Database Management System (RDBM) with a client-server model which is supported by Oracle Company. It is fast, scalable, and easy-to-use database management system compared to Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based applications. In designing the database for the new system, the following steps were carried out:

**(a) Requirement Analysis**

The purpose of requirement analysis is to gather information needed to design a database that meets the informational needs of the organization. An interview was conducted to understand how the association gets and stores data about its electoral processes.

**(b) Conceptual Modelling**

Using the Entity Relationship (ER) diagram, the conceptual modelling involves identifying the relevant entities and the relationship between them. The ER model helps to systematically analyse data requirements to produce a well-designed database. The entities involved are shown in Table 4.1 below:

|  |  |
| --- | --- |
| **ENTITIES** | **ATTRIBUTES** |
| Student | stdID, matric\_no, name, email |
| Aspirant | aspirant\_id, name, matric\_no, post |
| Vote | vote\_id, vote\_code, aspirant\_name, session |
| Admin | admin\_id, name, security\_question |

**Table 4.1: ER Diagram Entities**

Figure 4.5 shows the entity relation diagram of the proposed system displaying the entities, attributes, and the corresponding relationships

Admin ID

Name

Security Question

Vote

Electoral Observer

Student

Apply

Aspirant

Cast

Admin

Campaign

Upload

Register

Name

Name

Count

Matric No

Post

Matric No

Email

Std ID

Vote ID

Aspirant Name

Session

Vote code

Aspirant ID

Monitor

**Figure 4.5: Entity Relation Diagram of the New System**

**(Source: Author’s Concept)**

**(c) Logical Modelling**

Logical database modelling enables the designer to convert the conceptual model (ER diagram) to a schema in a chosen data model of the DBMS. For a relational database, this means converting the conceptual schema to a relational schema as shown in Figure 4.6 below:

**STUDENT**

|  |  |  |  |
| --- | --- | --- | --- |
| stdID | matric\_no | name | email |

**APPLY**

|  |  |
| --- | --- |
| stdID | aspirantID |

**ASPIRANT**

|  |  |  |  |
| --- | --- | --- | --- |
| aspirantID | matric\_no | name | post |

**CAMPAIGN**

|  |  |
| --- | --- |
| aspirantID | stdID |

**VOTE**

|  |  |  |  |
| --- | --- | --- | --- |
| voteID | vote\_code | aspirant\_name | session |

**CAST**

|  |  |
| --- | --- |
| stdID | voteID |

**ADMIN**

|  |  |  |
| --- | --- | --- |
| adminID | name | Security\_question |

**UPLOAD**

|  |  |
| --- | --- |
| stdID | adminID |

**COUNT**

|  |  |
| --- | --- |
| voteID | adminID |

**REGISTER**

|  |  |
| --- | --- |
| aspirantID | adminID |

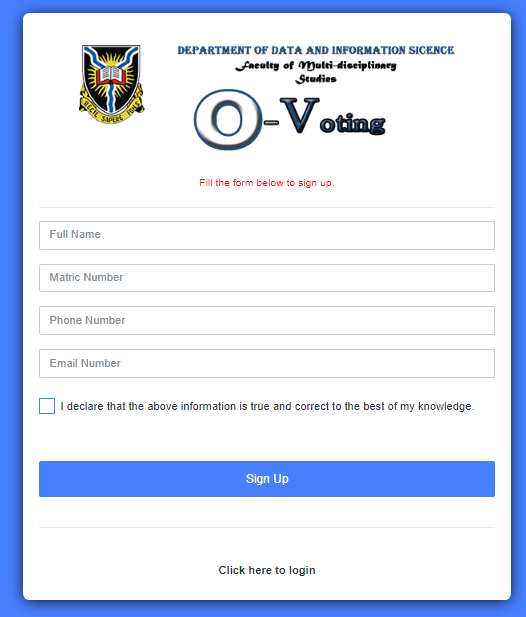
**Figure 4.6: Logical Modelling**

**(Source: Author’s Concept)**

**4.6 Input Design**

The input design is the layout of the interface between the information system and the user through HTML forms. It focuses on controlling the amount of input required, errors, avoiding delay, extra steps, and keeping the process simple.

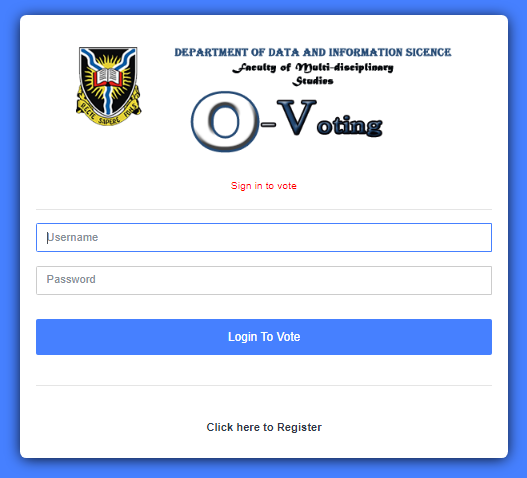
Figure 4.7 is an input form for voters to sign up by entering their required details



**Figure 4.7: Sign-up Page**

**(Source: Author’s Concept)**

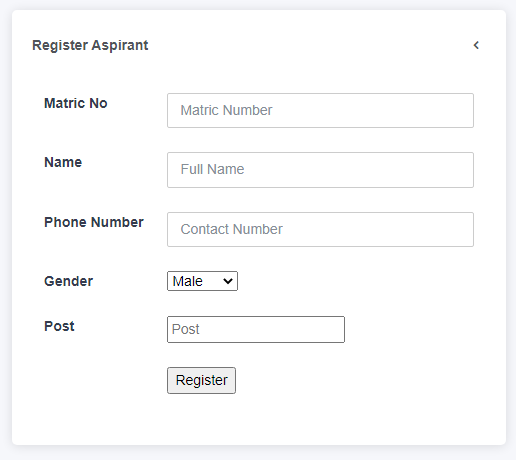
Figure 4.8 is an input form for voters, aspirants, admin, and election observers to log in by entering login details



**Figure 4.8: Login Page**

**(Source: Author’s Concept)**

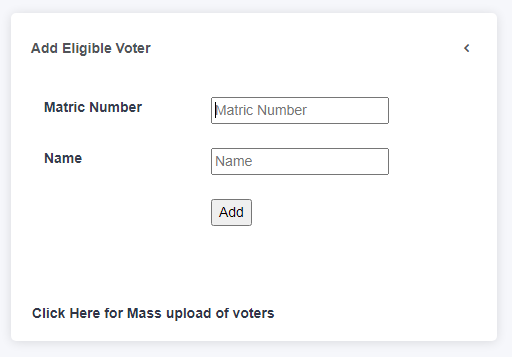
Figure 4.9 is an input form to register aspiring student



**Figure 4.9: Aspirant Registration Page**

**(Source: Author’s Concept)**

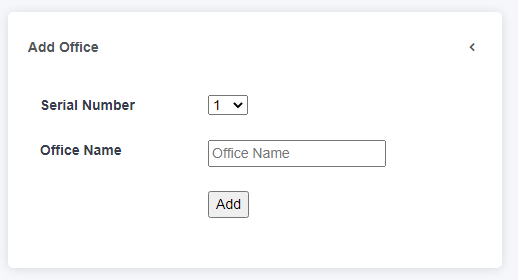
Figure 4.10 is an input form to add eligible voters



**Figure 4.10: Add Eligible Voter Page**

**(Source: Author’s Concept)**

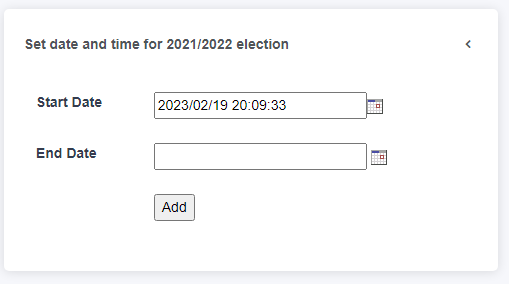
Figure 4.11 is an input form to add various aspiring offices



**Figure 4.11: Add Office Page**

**(Source: Author’s Concept)**

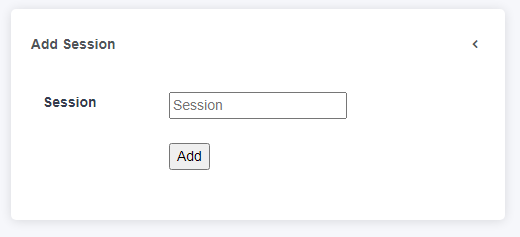
Figure 4.12 is an input form to set election date and time



**Figure 4.12: Set Election Duration Page**

**(Source: Author’s Concept)**

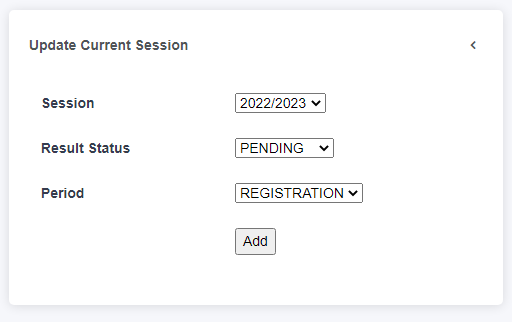
Figure 4.13 is an input form to add a new election session



**Figure 4.13: Add Session Page**

**(Source: Author’s Concept)**

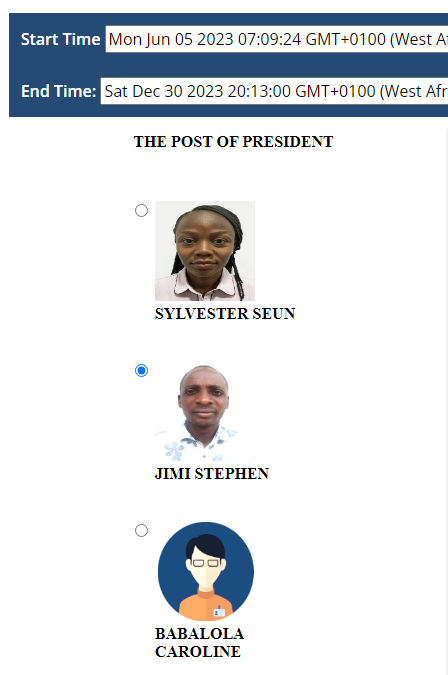
Figure 4.14 is an input form to update the current session, result status, and period of election activities



**Figure 4.14: Update Current Session Page**

**(Source: Author’s Concept)**

Figure 4.15 is an input form to cast a vote by a voter. All voting choices are confidential and cannot be linked to the voter and results will be hidden until the end of the election.



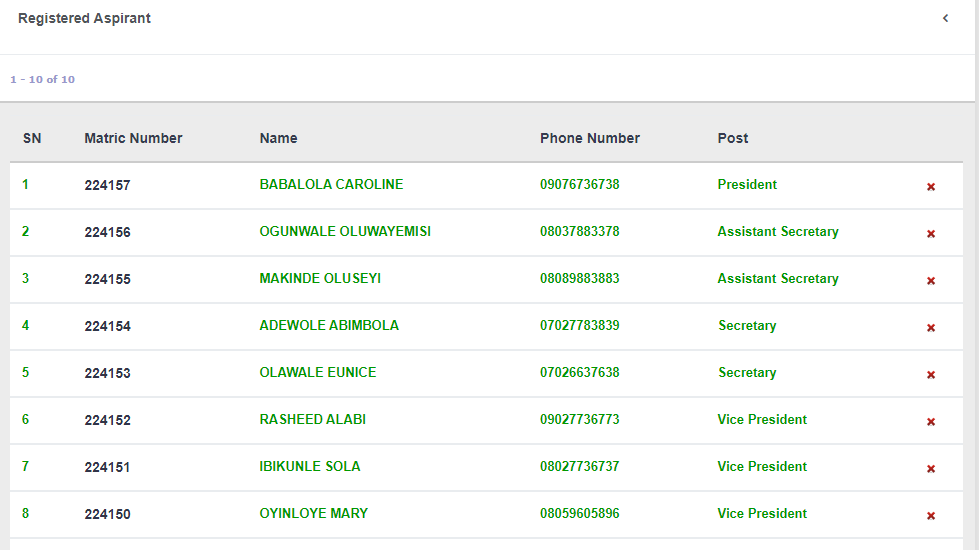
**Figure 4.15: Voting Page**

**(Source: Author’s Concept)**

**4.7 Output Design**

The output design determines how the information is to be displaced for immediate needs. It is the most important and direct source information to the user and efficient output design improves the system’s relationship to help user make decisions. These interfaces are shown below

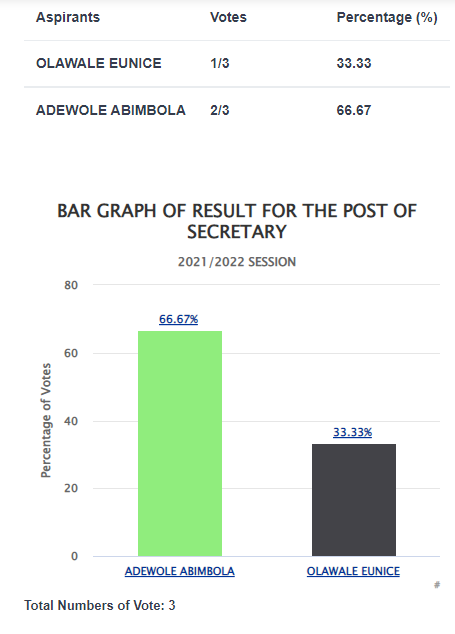
Figure 4.16 is an output view to display all registered aspirants



**Figure 4.16: Registered Aspirants Page**

**(Source: Author’s Concept)**

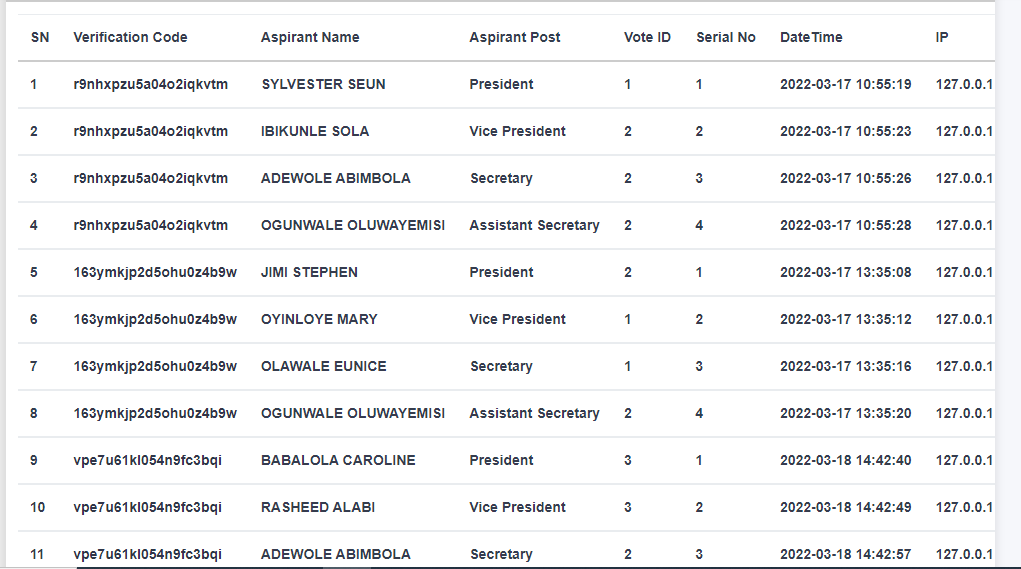
Figure 4.17 is an output view of election result



**Figure 4.17: Election Result Page**

**(Source: Author’s Concept)**

Figure 4.18 is an output view of an election result



**Figure 4.18: Election Audit Page**

**(Source: Author’s Concept)**

**4.8 Data Dictionary**

The data dictionary is the collection of all data elements or contents of a database such as names, types, sizes, and descriptions of the system. A data dictionary makes it easier for users and analysts to use data as well as understand and have common knowledge about the inputs, outputs, and components of a database. The data dictionary of the new system is shown in the tables below:

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | cid | INT(100) | The identity number of the user |
| 2 | name | VARCHAR(50) | Name of the user |
| 3 | username | VARCHAR(30) | Username of the user |
| 4 | password | VARCHAR(30) | The password of the user |
| 5 | t\_section | VARCHAR(50) | Electoral section |
| 6 | s\_question | VARCHAR(200) | Security question |
| 7 | s\_answer | VARCHAR(200) | Security Answer |

**Table 4.2: Create Login Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(11) | The identity number of the aspirant |
| 2 | sch\_session | VARCHAR(20) | Current school session of the election period |
| 3 | matric\_no | VARCHAR(200) | Matric number of the aspirant |
| 4 | password | VARCHAR(20) | Password of the aspirant |
| 5 | name | VARCHAR(200) | Full name of the aspirant |
| 6 | a\_profile | TEXT | Aspirant profile |
| 7 | gender | VARCHAR(20) | Gender of the aspirant |
| 8 | phone | VARCHAR(11) | Contact phone number of the aspirant |
| 9 | a\_post | VARCHAR(20) | Post aspiring to contest |
| 10 | s\_no | VARCHAR(5) | Post serial number |
| 11 | vid | VARCHAR(10) | Serial number of aspirants aspiring for the post |
| 12 | status | VARCHAR(20) | Status of aspirant registration |
| 13 | file\_name | VARCHAR(200) | Aspirant passport file name |
| 14 | file\_type | VARCHAR(200) | Aspirant passport file type |
| 15 | file\_size | VARCHAR(200) | Aspirant passport file size |
| 16 | date | VARCHAR(50) | Date of registration |
| 17 | ip | VARCHAR(20) | The IP address of the device used for the registration |

**Table 4.3: Aspirant Registration Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(100) | The identity number of the comment |
| 2 | fid | VARCHAR(20) | Forum post identity number |
| 3 | name | VARCHAR(50) | Name of the commenter |
| 4 | comment | TEXT | Comment of the commenter |
| 5 | m\_comment | VARCHAR(50) | Matric number of the commenter |
| 6 | ip | VARCHAR(20) | IP address of the device used by the commenter |
| 7 | date | VARCHAR(20) | Date commented |
| 8 | time | VARCHAR(20) | Time commented |

**Table 4.4: Comment Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | The identity number of the current session |
| 2 | sch\_session | VARCHAR(20) | Current school session |
| 3 | status | VARCHAR(20) | Status of the electoral process |
| 4 | period | VARCHAR(20) | Period of the electoral process |

**Table 4.5: Current Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | The identity number of the access code |
| 2 | a\_code | VARCHAR(50) | Access code number |
| 3 | created\_by | VARCHAR(20) | User created the access code |
| 4 | ip | VARCHAR(20) | The IP address of the device used to create the access code |
| 5 | cdate | VARCHAR(20) | Date generated the access code |
| 6 | ctime | VARCHAR(20) | Time generated the access code |

**Table 4.6: Election Audit Access Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | Identity number of the eligible list |
| 2 | matric\_no | VARCHAR(20) | Matric number of voter |
| 3 | s\_name | VARCHAR(200) | Name of the voters |
| 4 | sch\_session | VARCHAR(20) | Current school session |

**Table 4.7: Eligible List Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | fid | INT(100) | The identity number of the forum post |
| 2 | content | TEXT | Content sent to the forum |
| 3 | created\_by | VARCHAR(100) | User created the post |
| 4 | matric\_no | VARCHAR(50) | Matric number of the user |
| 5 | sch\_session | VARCHAR(20) | Current school session |
| 6 | ip | VARCHAR(20) | The IP address of the device used to create the post |
| 7 | pdate | VARCHAR(20) | Date created the post |
| 8 | ptime | VARCHAR(20) | Time created the post |

**Table 4.8: Forum Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | The identity number of the table |
| 2 | starttime | VARCHAR(50) | Voting start time |
| 3 | endtime | VARCHAR(50) | Voting end time |
| 4 | sch\_session | VARCHAR(10) | Current school session |

**Table 4.9: Instruction Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | The identity number of the table |
| 2 | matric\_no | VARCHAR(20) | Matric number of the aspirant |
| 3 | a\_name | VARCHAR(100) | Name of the aspirant |
| 4 | content | TEXT | Manifesto content |

**Table 4.10: Manifesto Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(100) | The identity number of the message |
| 2 | matric\_no | VARCHAR(50) | Matric number of the sender |
| 3 | name | VARCHAR(50) | Name of the sender |
| 4 | message | TEXT | Content of the message |
| 5 | sch\_session | VARCHAR(20) | Current school session |
| 6 | s\_date | VARCHAR(20) | Message sent date |
| 7 | status | VARCHAR(50) | Status of the message |

**Table 4.11: Message Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(4) | The identity number of the table |
| 2 | s\_no | VARCHAR(20) | Serial number of the office |
| 3 | name | VARCHAR(50) | Office Name |

**Table 4.12: Office Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(11) | The identity number of the voter table |
| 2 | matric\_no | VARCHAR(200) | Matric number of the voter |
| 3 | name | VARCHAR(200) | Full name of the voter |
| 4 | password | VARCHAR(200) | The password of the voter |
| 5 | phone\_no | VARCHAR(11) | Contact phone number of the voter |
| 6 | email | VARCHAR(50) | Email of the voter |
| 7 | declaration | VARCHAR(20) | Voter information declaration |
| 8 | status | VARCHAR(20) | Status of voter registration |
| 9 | file\_name | VARCHAR(200) | Voter passport file name |
| 10 | file\_type | VARCHAR(200) | Voter passport file type |
| 11 | file\_size | VARCHAR(200) | Voter passport file size |
| 12 | date | VARCHAR(50) | Date of registration |
| 13 | ip | VARCHAR(20) | The IP address of the device used for the registration |

**Table 4.13: Voter Registration Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | id | INT(11) | The identity number of the result table |
| 2 | matric\_no | VARCHAR(20) | Matric number of the voter |
| 3 | v\_code | VARCHAR(200) | Voter verification code |
| 4 | sch\_session | VARCHAR(50) | Current school session |
| 5 | a\_name | VARCHAR(200) | Name of the aspirant |
| 6 | a\_post | VARCHAR(200) | Aspiring office |
| 7 | vid | VARCHAR(50) | Serial number of aspirants aspiring for the post |
| 8 | s\_no | VARCHAR(20) | Post serial number |
| 9 | ip | VARCHAR(20) | The IP address of the device used to cast a vote |
| 10 | date | VARCHAR(20) | Date cast vote |
| 11 | time | VARCHAR(20) | Time cast vote |
| 12 | status | VARCHAR(20) | Status of the vote cast |

**Table 4.14: Election Result table**

|  |  |  |  |
| --- | --- | --- | --- |
| **S/N** | **Field** | **Type(Size)** | **Description** |
| 1 | session\_id | INT(4) | The identity number of the table |
| 2 | schl\_session | VARCHAR(20) | Current school session |

**Table 4.15: School session table**

**4.9 System Specification**

This consists of both the hardware and software components that the new system required to function. The new system will run only on the web platform and the following system specifications are needed.

|  |  |  |
| --- | --- | --- |
| **CATEGORY** | **REQUIRED** | **DESCRIPTION** |
| Web Server | 1GB Disk Space, 10GB Bandwidth and 2GB RAM | Minimum hosting requirement |
| Operating System | Microsoft Windows, Android OS, Apple, Mac | Web browser host platform operating system |
| Web Browser | Google Chrome, Firefox, Internet Explorer, Safari, Opera | Displaying web pages from the web server |
| JavaScript | Supported and Enabled | Front end functionality |
| Internet Access | 1-10Mbps | WAN access services |

**Table 4.16: Hardware and Software Specification**

**4.10 System Security and Control**

For a system to perform its required functions and give valid output, security measures, and standards must be put in place. The new system was subjected to security and control measures to safeguard the integrity of the data collected to be stored in a database. The following security measure was implemented.

**(a) Operational Control**

This refers to the process of controlling input into the system. It is aimed at validating entries into the new system. The new system used restricted value checks to fields that are known to take only a certain number of values.

**(b) Access Control**

These controls ensure the authentication of the right person to access and use the system.

1. Username and Password: These are unique identifiers used to authenticate each system user thereby restricting unauthorized users from accessing the system.
2. Access Level: This is used to restrict privileges and access among users.