**DESIGN AND IMPLEMENTATION OF ONLINE VOTING SYSTEM**

**(A CASE STUDY OF SRC FEDERAL UNIVERSITY BIRNIN KEBBI)**

**BY**

**OYEFABI BENJAMIN OLAMIDE**

**(ADM NO. 1510204054)**

**A PROJECT PRESENTED TO:**

**DEPARTMENT OF COMPUTER SCIENCE,**

**FEDERAL UNIVERSITY BIRNIN KEBBI,**

**KEBBI STATE OF NIGERIA.**

**IN PARTIAL FULFILMENT OF THE REQUIREMENTS**

**FOR THE AWARD OF DEGREE**

**(BACHELOR OF SCIENCE HONOURS) IN COMPUTER SCIENCE**

**SUPERVISOR:**

**MR. ABDULHAKEEM IBRAHIM**

**SEPTEMBER, 2019.**

**CERTIFICATION**

This is to certify that **Oyefabi Benjamin Olamide** with matric number 1510204054 compiled this report in partial fulfillment of requirements for the award of degree of Bachelor of Science in Computer Science.

…………………………………. ………………………

Abdulhakeem Ibrahim Date

Project Supervisor

………………………………….. ………………………

Aminu Aliyu Date

Departmental Project Coordinator

………………………………… ………………………

Faruku Musa Aliyu Date

Ag. Head of Department

**DEDICATION**

This report is dedicated foremost to God Almighty for his grace, mercy, and favor upon my life. I would also like to dedicate it to my parents for their love and support, and everyone else that contributed towards making my study a successful one.

**ACKNOWLEDGMENT**

With deep sense of gratitude, respect, appreciation, I am deeply indebted to God Almighty, the giver of all wisdom, knowledge and understanding, without whom I would have achieved nothing at all. To my parents' Overseer and Deaconess Oyefabi, my brother Issac Abiodun Oyefabi, my sisters Elizabeth Ayobami Oyefabi and Shefiat Babatunde and friends, thank you all for your moral and financial support from the beginning of my program to this point.

I want to say a very big thank you to my able Ag. H.O.D. Mal. Farouk M. Aliyu, my project supervisor Mr. Abdulhakeem Ibrahim, my departmental project coordinator Mr. Aminu Aliyu and everyone in ICT and Computer Science Department FUBK for your support, thank you all. To my amiable mentor, Mr. Abubakar M. Hashim thank you for the supportive advice and encouragement.

**Table of Contents**

**Title Page** i

**Certification** ii

**Dedication** iii

**Acknowledgment** iv

**Table of Contents** v

**Abstract** vi

**Chapter One: Introduction** 1-5

# ABSTRACT

Online Voting System is an interactive voting system application with which users can vote from any location remotely using their information stored prior to database securely. An online voting system involves transmission of ballots and votes via network. The main objective of this work is to develop an interactive voting system application to replace the appointing method adopted by the school authorities with which users can participate in voting for their various candidate using their information stored prior in database and users verification are done by the system through OTP Message send the user’s (voter’s) phone number stored in the database. In this system students of Federal University Birnin Kebbi can cast their vote through online at any location without going to any physical polling station

# CHAPTER ONE

# Introduction

This chapter introduces the main topic of the research with the following headings; Background of the study, the purpose of the study, research problem, research aims and objectives, scope and limitation of the study.

**1.1 Background of the study**

Voting is a mean of choosing or selecting leaders and is not restricted to only Federal, State and Local Government of Nigeria but also institutions and social organizations. Therefore, the conduct of elections in a democratic society is very important not only because through it a change of leadership is effective but also because voting is the main form of political participation for most people for democracy to be sound and smooth, it should be enhanced on election free from fraud (Idongesit, 2018).

For a voting system to be ideal, four attributes must be satisfied: anonymity, scalability, speed, and accuracy. Online Voting System is a web-based system that facilitates the running of elections and surveys online. This system has been developed to simplify the process of organizing elections and make it convenient for voters to vote remotely from their home computers while taking into consideration security, anonymity and providing auditioning capabilities.

The Students Representative Council (SRC), Federal University Birnin Kebbi, is a student body that seeks the welfare of the students in all of the Faculties (Science, Art, Social and Management Science, Health Science and Environmental) of the University. Appointing of Students Representative is carried out every session where the President and nine (8) other executive members are appointed by the School Authorities and conducted by the SRC Electoral Commission.

Online Voting System provides a platform for simplifying the electoral process for all institutions that employ voting in decision-making. It is geared towards increasing the voting percentage in Universities and Colleges since it has been noted that with the old voting method (Manual System), the voter turnout has been wanting case.

**1.2 Purpose of Study**

The online voting system will be designed to meet the electoral needs of the university and also tackles the inherent problems of the present manual voting system. This current system (manual voting) is characterized by absenteeism, inconvenience, long queues, stress, a lot of paperwork, error-prone human effort involved in vote computation, omissions, delays and other election irregularities that plague the system and defeat the whole aim of voting.

In view of the rapid development of computer technology in virtually all fields of operations and its use in relation to information management, it has become pertinent to look into the development of an Online Voting System that can achieve the following:

1. Each voter will be able to vote only once
2. Nobody will have access to the votes before the official opening of the electronic ballot box.
3. The votes cast cannot be intercepted, modifies or diverted
4. Conduct free and fair elections.
5. Only registered voters will have access to the application
6. Safeguard data and information in the system.
7. Reduce workload in the process of conducting elections.
8. Keep an accurate record of votes.
9. The System will not accept vote outside the voting period
10. Reduce time wasted in announcing election results.
11. Eliminate disenfranchising electorates.

# 1.3 Research Problem

The appointed method that is adopted in the selection of the Student Representatives in the FUBK (Federal University Birnin Kebbi) slows down the decision making process of the Association (SRC) and the fact that they are been appointed not elected limits their power. Some of the appointed Student Representative are mediocre and they up disappointing the school as a whole. But if the students are given the right to elect their respective candidate of their choice it will be of benefit to the school in question.

# 1.4 Research Aims and Objective

1. Design a voting system for the University that uses computer technology or information technology to simplify the electoral process.
2. Review the existing/current appointing process or approach in the university;
3. Design an automated voting system that should be able to handle extremely large volumes of data;
4. Implement an automated/online voting system that should support multi-user environment;

# 1.5 Scope of the Study

The system is designed in order to replace the appointing processes of Student Representative Council (SRC) Federal University Birnin Kebbi with an automated voting process. Online Voting will also include the voter’s personal records, candidate’s personal records, and voting procedures.

# 1.6 Limitation of the Study

This Design project work is limited to the Student Representative Union of Federal University Birnin Kebbi State.

**1.7 Project Organization**

The purpose of this research document is to give an overview of major phases involved throughout the development of the project. Basically, the research document is divided into five (5) Chapters.

***Chapter 1*** Covers the introduction of the research, overview of the research including problem statements, objectives, scope and limitations of the research.

***Chapter 2*** Covers literature review of research. The chapter concentrates on the overall aspect of the internet as a framework for new economy, the emergence of the web, understanding the concept of web mining, web scraping and web crawling. It also includes exploration and observation of the existing systems, and finally presents research gap.

***Chapter 3*** Covers system analysis and design. The chapter contains justification of the proposed methodology used in completing the research. It also includes requirement analysis of system, analysis of the proposed system functionality, system Modeling, tools and technology to be used during the research.

***Chapter 4*** Covers system implementation and testing. This chapter contains some of the necessary factors that affect system implementation such as Human-Computer Interaction factors, Usability factors, Web Page Design Principles and Justification of the Programming

Language used.

***Chapter 5*** Covers the summary, conclusion, and recommendation of the research.

**CHAPTER TWO**

1. **Introduction**

This chapter of this project considers how Science and Technology becomes a framework for a new economy and also attempts to explore and consider various opinions of some scholars and practitioners as regards the development and implementation of Online Voting System. Finally, the organizational pattern of this literature review is thematically.

# 2.1 Literature Review

Grossman (2003), Developed the Simply Voting system that is full-time online voting service provider purposely designs to host simple elections. It was launched in 2003 and the majority of its clients are university students. It is said to have over 1000 organizations from 48 countries which lean on it for electoral services. But one weakness Simply Voting system possesses is that it does not allow the user to create election start date, election end date, the names of the contenders, or any information that could help in identifying the participants. However, it is vividly clear that the system in question does not undergo proper maintenance as conﬁrmed by the hosting vendor.

Event A. (2009), Designed Election Buddy which is another online voting platform. They provide their services to their immediate communities. Based on our observations, Election Buddy is characterized by a single interface which allows the users to vote only one type of category at a given period of time. No multiple categories are allowed for users. Hence, the system at ﬁrst allows the users to create accounts and then reconnect them to a separate page where the user is expected to ﬁll his voting request, in other words, his voting list. However, after creating the voting category, the system then furthers its request to the user to create a ballot design. And under this ballot design, the user is required to put down a design of options, requirements and so many other options that are unnecessary. By making the design complexity to the user, the system violates one of the principles ISO 9126 (McCall, 1977; Boehm, 1978) which states that software should be usable and learnable. As such, it makes it difﬁcult for some users to be able to design the ballot and of course to proceed to the voting area.

K.P Kaliyamurthe, R.Udayakumar, D. Parameswari & S. Nmuguntha (2013). Developed an interactive voting system application with which users can participate using their information stored prior to the database while creating the voter ID and the information needs to be updated at a period of less than six months for perfect user verification. But the System has a poor method of authenticating users in the sense that, another user can use the logging details of another voter to cast vote.

Amna Saad (2014). Developed an Electronic Voting System for campus election and the system is based on a client-server system and the application was developed using Microsoft Visual Basic (VB6) and SQL as the database. The system is been integrated with an RFID system for the voter’s authentication purpose. The System consists of two application programs which include: Management Program which is to be used by the administrator and the other is the voting terminal program for voters to cast their votes. The system wastes a lot of time due to the process involved before the voter will be allowed to cast vote.

Mr. Prashant Pandit, Mr. Sagar Bhawar and Prof. Manisha. Desai (2014),They proposed a system that is implemented to allow each and every student to actively participate in the college election process irrespective of the place. This is done by the android application which will accept the votes of different students using the application and also through web application. The rapid development in operating system of mobile phones gives rise to application development on a large scale. The system allows each and every student to actively participate so that they can get acquainted with the candidates and select the college representatives. The aim is to provide convenient, easy and safe way to capture and count the votes in college elections. The system also denotes a voting process, which enables voters to cast a secure and secret ballot over a network as the manual voting process is time-consuming and prone to security breaches. But the system does not have any means of authenticating voter which means, even students from other schools can take the logging details of another student to perform voting exercise on his or her behalf so therefore, is open to fraud.

Olayemi M. Olaniyi et al, (2016). They developed a system to solve the authentication, integrity and confidentially security issue of e-voting in kiosk and poll site e-voting scenarios using unimodal fingerprint biometric and advanced encryption standard based wavelet-based Crypto-water making Approach. The developed system solves the possibility of blundering voter’s authentication, integrity, and confidentiality of vote stored in the server. One problem this system face is that the voters cannot cast votes from any location using their mobile phones or Computer but rather, they have to come to the polling unit in order to go through the authentication exercise using the Fingerprint Biometric Machine which in turn waste a lot of time and others might not be able to cast their votes due to failure of machine from authenticating their fingerprint.

Mr. Avadhut Ghadashi1 et al, (2016). Developed an online voting system that work on Ethernet and allows online voting. It will also create and manage voting and an election detail as all the users must log in by user name and password as well as upload own picture using webcam, Election Commission officer verifies the user by comparing the user’s submitted picture (using webcam only) and the existing image of user from the database. Only Authenticated users are allowed for further process, user can click on his favorable candidates to cast vote. In case of identical twins, the system verification fails in such situation since is human that is responsible for verifying the user image submitted and the one in the database, not the computer.

Umesh A. Wakpanjar et al (2018), introducing Finger Print Based Voting, where a person can be authorized based on his fingerprint. This will also stop fake voting. So today we are building Fingerprint Based Biometric Voting Machine using Arduino. It is an application where the user is recognized by his finger pattern. Since the finger pattern of each human being is different, the voter can be easily authenticated. The system allows the voter to vote through his fingerprint. Fingerprint is used to uniquely identify the user. The finger print minutiae features are different for each human being. Fingerprint is used as an authentication of the voters. Voters can vote the candidate only once; the system will not allow the candidate to vote for the second time. The weakness of this system is that, if there exist students who engage in farming work or other technical work or even accident and have little scratch on thumbs they will not be able to vote because the Biometric machine will not be able to authenticate the voter and it also Waste a lot of time since the students need to come together in a polling unit to authenticate using biometric machine and cast their votes.

**2.2 Emergence of the Web**

Sir Tim Berners-Lee noticed the difficulty of sharing information. *“In those days, there was different information on different computers, but you had to log on to different computers to get at it. Also, sometimes you had to learn a different program on each computer. Often it was just easier to go and ask people when they were having coffee…”* Tim says.

Tim thought he saw a way to solve this problem – one that he could see could also have much broader applications. Already, millions of computers were being connected together through the fast-developing internet and Berners-Lee realized they could share information by exploiting an emerging technology called hypertext. In March 1989, Tim laid out his vision for what would become the web in a document called “Information Management: A Proposal”. Believe it or not,

Tim’s initial proposal was not immediately accepted. In fact, his boss at the time, Mike Sendall, noted the words “Vague but exciting” on the cover. The web was never an official CERN project, but Mike managed to give Tim time to work on it in September 1990. He began work using a NeXT computer, one of Steve Jobs’ early products. By October of 1990, Tim had written the three fundamental technologies that remain the foundation of today’s web (and which you may have seen appear on parts of your web browser):

* **HTML:** Hypertext Markup Language. The markup (formatting) language for the web.
* **URI:** Uniform Resource Identifier. A kind of “address” that is unique and used to identify each resource on the web. It is also commonly called a URL.
* **HTTP:** Hypertext Transfer Protocol. Allows for the retrieval of linked resources from across the web.

Tim also wrote the first web page editor/browser (“WorldWideWeb.app”) and the first web server (“https“). By the end of 1990, the first web page was served on the open internet, and in 1991, people outside of CERN were invited to join this new web community. As the web began to grow, Tim realized that its true potential would only be unleashed if anyone, anywhere could use it without paying a fee or having to ask for permission. He explains (Berners, S. T., & Web, W. W. (2018): *“Had the technology been proprietary, and in my total control, it would probably not have taken off. You can’t propose that something be a universal space and at the same time keep control of it.”* So, Tim and others advocated ensuring that CERN would agree to make the underlying code available on a royalty-free basis, forever. This decision was announced on April 1993, and sparked a global wave of creativity, collaboration and innovation never seen before. In 2003, the companies developing new web standards committed to a Royalty Free Policy for their work. In 2014, the year we celebrated the web’s 25th birthday (Berners, S. T., & Web, W. W. (2018), almost two in five people around the world were using it. Tim moved from CERN to the Massachusetts Institute of Technology in 1994 to found the World Wide Web Consortium. (Berners, S. T., & Web, W. W. (2018)), an international community devoted to developing open web standards. He remains the Director of W3C to this day. The early web community produced some revolutionary ideas that are now spreading far beyond the technology sector:

* Decentralization: No permission is needed from a central authority to post anything on the web; there is no central controlling node, and so no single point of failure and no “kill switch”! This also implies freedom from indiscriminate censorship and surveillance.
* Nondiscrimination: If I pay to connect to the internet with a certain quality of service, and you pay to connect with that or greater quality of service, then we can both communicate at the same level. This principle of equity is also known as Net Neutrality.
* Bottom-up design: Instead of code being written and controlled by a small group of experts, it was developed in full view of everyone, encouraging maximum participation and experimentation.
* Universality: For anyone to be able to publish anything on the web, all the computers involved have to speak the same languages to each other, no matter what different hardware people are using; Where they live; Or what cultural and political beliefs they have. In this way, the web breaks down silos while still allowing diversity to flourish.
* Consensus: For universal standards to work, everyone had to agree to use them. Tim and others achieved this consensus by giving everyone a say in creating the standards, through a transparent, participatory process at W3C.

New permutations of these ideas are giving rise to exciting new approaches in fields as diverse as information (Open Data), politics (Open Government), scientific research (Open Access), education, and culture (Free Culture). It has also helped in the building of Online voting system which enables many individuals from different location to be able to cast their votes without physically present in the polling station once they are connected to the internet.

**2.3 Electronic Voting:**

Electronic voting (also known as e-voting) is voting that uses electronic means to either aid or take care of casting and counting votes. Depending on the particular implementation, e-voting may use standalone electronic voting machines (also called EVM) or computers connected to the Internet. It may encompass a range of Internet services, from basic transmission of tabulated results to full-function online voting through common connectable household devices. The degree of automation may be limited to marking a paper ballot, or maybe a comprehensive system of vote input, vote recording, data encryption and transmission to servers, and consolidation and tabulation of election results. A worthy e-voting system must perform most of these tasks while complying with a set of standards established by regulatory bodies, and must also be capable to deal successfully with strong requirements associated with security, accuracy, integrity, swiftness, privacy, auditability, accessibility, cost-effectiveness, scalability and ecological sustainability.

Electronic voting technology can include punched cards, optical scan voting systems and specialized voting kiosks (including self-contained direct-recording electronic voting systems, or DRE). It can also involve transmission of ballots and votes via telephones, private computer networks, or the Internet. In general, two main types of e-voting can be identified:

* e-voting which is physically supervised by representatives of governmental or independent electoral authorities (e.g. electronic voting machines located at polling stations);
* remote e-voting via the Internet (also called i-voting) where the voter submits their votes electronically to the election authorities, from any location.

**2.4 Type of E-voting**

Electronic voting systems for electorates have been in use since the 1960s when punched card systems debuted. Their first widespread use was in the USA where 7 counties switched to this method for the 1964 presidential election. The newer optical scan voting systems allow a computer to count a voter's mark on a ballot. DRE voting machines which collect and tabulate votes in a single machine, are used by all voters in all elections in Brazil and India, and also on a large scale in Venezuela and the United States. They have been used on a large scale in the Netherlands but have been decommissioned after public concerns.

Internet voting systems have gained popularity and have been used for government elections and referendums in Estonia, and Switzerland as well as municipal elections in Canada and party primary elections in the United States and France. There are also hybrid systems that include an electronic ballot marking device (usually a touch screen system similar to a DRE) or other assistive technology to print a voter-verified paper audit trail, then use a separate machine for electronic tabulation.

**2.4.1 Paper-based electronic voting system**

Sometimes called a "document ballot voting system", paper-based voting systems originated as a system where votes are cast and counted by hand, using paper ballots. With the advent of electronic tabulation came systems where paper cards or sheets could be marked by hand, but counted electronically. These systems included punched card voting, mark sense and later digital pen voting systems. These systems can include a ballot marking device or electronic ballot marker that allows voters to make their selections using an electronic input device, usually a touch screen system similar to a DRE. Systems including a ballot marking device can incorporate different forms of assistive technology. In 2004, Open Voting Consortium demonstrated the 'Dechert Design', a General Public License open-source paper ballot printing system with open source bar codes on each ballot.

**2.4.2 Direct-recording electronic (DRE) voting system**

A direct-recording electronic (DRE) voting machine records votes by means of a ballot display provided with mechanical or electro-optical components that can be activated by the voter (typically buttons or a touchscreen); that processes data with computer software; and that records voting data and ballot images in memory components. After the election, it produces a tabulation of the voting data stored in a removable memory component and as a printed copy. The system may also provide a means for transmitting individual ballots or vote totals to a central location for consolidating and reporting results from precincts at the central location. These systems use a precinct count method that tabulates ballots at the polling place. They typically tabulate ballots as they are cast and print the results after the close of polling.

**2.4.3 Electronic Voting Machine (EVM) used in India**

In 2004, India adopted Electronic Voting Machines (EVM) for its elections to its parliament with 380 million voters casting their ballots using more than one million voting machines. The Indian EVMs are designed and developed by two government-owned defense equipment manufacturing units, Bharat Electronics Limited (BEL) and Electronics Corporation of India Limited (ECIL). Both systems are identical, and are developed to the specifications of Election Commission of India. The system is a set of two devices running on 7.5volt batteries. One device, the voting Unit is used by the voter, and another device called the control unit is operated by the electoral officer. Both units are connected by a five-meter cable. The voting unit has a blue button for each candidate. The unit can hold 16 candidates, but up to four units can be chained, to accommodate 64 candidates. The control unit has three buttons on the surface – one button to release a single vote, one button to see the total number of votes cast till now, and one button to close the election process. The result button is hidden and sealed. It cannot be pressed unless the close button has already been pressed. A controversy was raised when the voting machine malfunctioned which was shown in Delhi assembly.

**2.4.4 Public network DRE voting system**

A public network DRE voting system is an election system that uses electronic ballots and transmits vote data from the polling place to another location over a public network. Vote data may be transmitted as individual ballots as they are cast, periodically as batches of ballots throughout the election day, or as one batch at the close of voting. This includes Internet voting as well as telephone voting. Public network DRE voting system can utilize either precinct count or central count method. The central count method tabulates ballots from multiple precincts at a central location.

**Internet voting** can use remote locations (voting from any Internet-capable computer) or can use traditional polling locations with voting booths consisting of Internet-connected voting systems.

Corporations and organizations routinely use Internet voting to elect officers and board members and for other proxy elections. Internet voting systems have been used privately in many modern nations and publicly in the United States, the UK, Switzerland and Estonia. In Switzerland, where it is already an established part of local referendums, voters get their passwords to access the ballot through the postal service. Most voters in Estonia can cast their vote in local and parliamentary elections, if they want to, via the Internet, as most of those on the electoral roll have access to an e-voting system, the largest run by any European Union country. It has been made possible because most Estonians carry a national identity card equipped with a computer-readable microchip and it is these cards which they use to get access to the online ballot. All the voters need a computer, an electronic card reader, their ID card and PIN, and they can vote from anywhere in the world. Estonian e-votes can only be cast during the days of advance voting. On election day itself people have to go to polling stations and fill in a paper ballot.

**CHAPTER THREE**

**SYSTEM ANALYSIS AND DESIGN**

**3.0 Introduction:**

Chapter three of this study presents the methodology, analysis, and design of the Online Voting System, the proposed system.

The development of the World Wide Web has had a profound effect on all of our lives. Initially, the Web was primarily a universally accessible information store and it had little effect on software systems. These systems ran on local computers and were only accessible from within an organization. Around 2000, the Web started to evolve and more and more functionality was added to browsers. This meant that web-based systems could be developed where, instead of a special-purpose user interface, these systems could be accessed using a web browser. This led to the development of a vast range of new system products that delivered innovative services, accessed over the Web.

The advent of the web, therefore, has led to a significant change in the way that business software is organized. Before the web, business applications were mostly monolithic, single programs running on single computers or computer clusters. Communications were local, within an organization. Now, software is highly distributed, sometimes across the world. This radical change in software organization has, obviously, led to changes in the ways that web-based systems are engineered. For example: It is now generally recognized that it is impractical to specify all the requirements for such systems in advance. Web-based systems should be developed and delivered incrementally. User interfaces are constrained by the capabilities of web browsers. Although technologies such as AJAX (Holdener, 2008) mean that rich interfaces can be created within a web browser, these technologies are still difficult to use. Web forms with local scripting are more commonly used. Therefore, the methodology used in developing this web-based system is an incremental approach, as it is easier to change the system process to reflect changing customer requirements.

**3.1 Requirement Analysis**

Requirement analysis has been carried out in order to identify system requirements which includes user requirements, software requirements and hardware requirements. The system requirements were identified through reviewing findings of recent studies, observation of similar existing systems and discussion with experienced users.

**3.1.1 User Requirement**

The user requirements include:

1. The User must be a Student of Federal University Birnin Kebbi

2. The user must Have a working Phone Number for verification

**3.1.2 Software Requirement**

Software requirement that needs to be installed to provide optimal functioning of the proposed

system include;

1. Operating System: Such as Windows 7, 8, 8.1 and 10, macOS, Linux.

2. Web browser: Such as Mozilla Firefox, Google Chrome, UC browser, Opera mini, etc

3. JavaScript

4. SQL database

5. CSS

**3.1.3 Hardware Requirement**

Physical computer resources also known as hardware is among the most common set of requirements by any operating system or software application. The physical resources required in the proposed system include;

1. Processing Power: The power of the central processing unit is a fundamental system requirement for any software. AMD Athlon at 1GHz or Intel Pentium at 1GHz or higher is required.

2. Memory: All software, when run, resides in random access memory (RAM) of a computer therefore, 1GB of RAM or higher is required.

3. Secondary Storage: Hard-disk requirement is essential, because the user often needs to store information on secondary storage device for later use. Hence, 65GB free of space is required on hard disk.

**3.2 Analysis of the Proposed System Functions**

The system functions can be divided into three modules, the administrator, Candidates and the Students module. Administrator is a super person that has control over the system, candidate can view results and update profile while students are Voters.

**3.2.1 Functional Requirement**

Functional requirement is a function or feature that must be included in an information system to satisfy the business needs and user acceptance (Whitten, 2002). A clear and detail functional requirement for the proposed system as described below;

a. Administrator should be able to access number of Voters and votes.

b. Administrator should be able to upload the list students in the University with their personal information.

c. The administrator should able to set the time for voting and Upload latest news on the page.

d. Administrator should be able to add/delete Candidate

e. Students should be able to vote and cast votes.

f. Students should be able to view personal profile and update if necessary.

g. Students should be able to view Election result after the voting time expiries

h. Candidate should be able to update his or her profile and be able to check the final result after the voting period expiry.

**3.2.2 Non-Functional Requirement**

Non-functional requirement is a description of the features, characteristics, and attributes of the system as well as any constraints that may limit the boundaries of the proposed solution (Whitten, 2002). Such constraints usually narrow down the selection of programming language,

operating system platform or implementation techniques.

The proposed system, Email Crawler; a web-based email scraper must ensure certain web application qualities such as ease of use, user-friendliness, correctness, functionality, reliability, response time, security, robustness as well as maintainability. The following lists the non-functional requirements of the system.

a. Ease of use

b. Reliability

c. Robustness

d. Response time

e. Correctness

f. User-friendly

g. Security

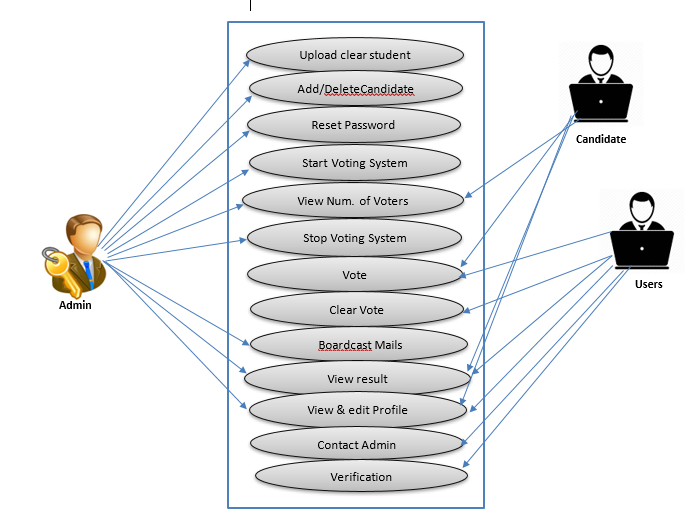
**3.3 System Modelling**

This section of the project presents abstract models of the proposed system (i.e Online Voting System) with each model presenting a different view or perspective of the system. The models are derived from the system requirements.

**3.3.1 Use Case**

A use case diagram at its simplest is a representation of a user’s interaction with a system and depicting the specification of a use case (Gemino & Parker, 2009). Use case analysis is a major technique used to find out the functional requirements of a software system. Use case, an important concept in use case analysis, represents an objective user wants to achieve with a system. It can be in text form, or be visualized in a use case diagram. A use case is an objective user(s) wants to achieve with a system. Use cases are named with the verb or verb + noun phrase. It is usually short yet descriptive enough to describe a user objective. You are encouraged to use concrete and specific verbs and nouns to avoid ambiguity. The diagram represents the use case diagram for the proposed system (Online Voting system).

**3.3.1.1 Use Case Diagram**

**Actors:** Administrator, candidates, Students

**Use case:**

**Fig 3.1 Online Voting Use case Diagram**

**3.3.2 Activity Diagram**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support of choice, iteration, and concurrency (Rumbaugh & Jacobson, 1999). In unified modeling language, activity diagrams are intended to model both computational and

organizational processes (i.e. workflows). Activity diagrams show the overall flow of control.

Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shapes type: rounded rectangles represent actions; diamonds represent decisions; bars represent the start (split) or end (join) of concurrent activities; a black circle represents the start (initial state) of the workflow; an encircled black circle represents the end (final state). Arrows run from the start towards the end and represent the order in which activities happen

Authenticate Admin username and password

Check

Validity

Not valid

Valid

Add Candidate

Load Voters data

Set Election Time

View result

**Fig 3.2 Admin Active Diagram**

Authenticate Candidate username and password

Check

Validity

Not valid

Valid

View No Voters

Update Profile

View result

Contact Admin &

Enquirer

**Fig 3.2 Candidate Active Diagram**

Authenticate Student(voter) username and password

Check

Validity

Not valid

Valid

Vote

Update Profile

View result

Contact Admin &

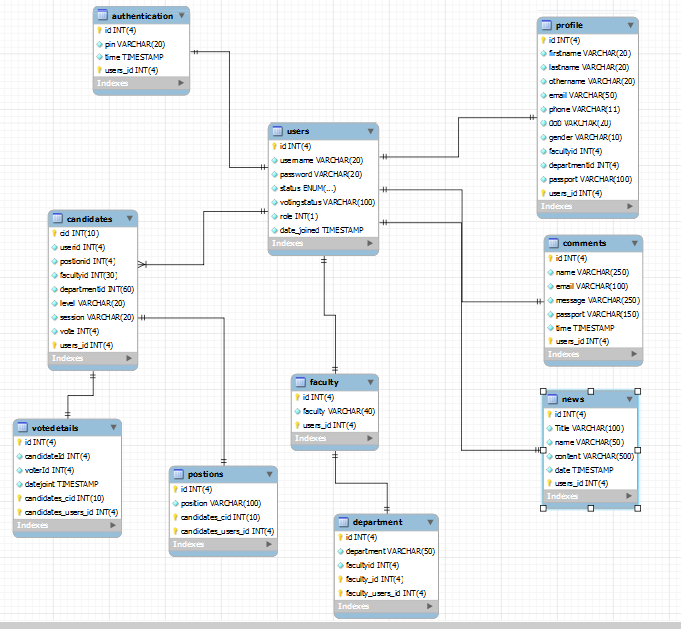
Enquirer

**Fig 3.3 Voter Active Diagram**

**3.3.3 Database Design**

The process of database design of the proposed system is divided into different parts. It consists of a series of steps. They are Conceptual Database Design (ER diagram), Logical Database Design (Tables, Normalization, etc.), Physical Database Design (Table indexing, Clustering, etc.)

**3.3.3.1 Conceptual Database Design**

The requirement analysis is modeled in this conceptual design. The ER Model is used at the conceptual design stage of the database design. The ER diagram is used to represent this conceptual design. ER diagram consists of Entities, Attributes and Relationships. The figure below shows the ER diagram of SRC Online Voting System.

**Figure 3.4**

**3.3.3.2 Logical Database Design**

Once the relationships and dependencies are identified the data can be arranged into logical structures and is mapped into database management system tables. Normalization is performed to make the relations inappropriate normal forms. The database has been normalized to third normal form.

USERS (id, username, password, status, votingstatus, role, date\_joint)

PROFILE (id, userId, firstname, lastname, othername, email, phone, dob, gender, facultyid, departmentid, passport)

CANDIDATES (cid, userId, postionId, facultyId, departmentId, level, session, vote)

AUTHENTICATION (id, userId, pin, time)

COMMENTS (id, userId, message, time)

DEPARTMENT (id, department, facultyId)

FACULTY (id faculty)

POSITIONS (id, position)

VOTEDETAILS (id, candidateId, voterId, datejoint)

**3.3.3.3 Physical Database Design**

Physical database design deals with the physical implementation of the database in a database management system. It includes the specification of data elements, data types, indexing, etc.

The tables below show the physical database design of the SRC Online Voting System.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Users | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to identify each User |
| username | VARCHAR | 20 |  |  |  | A field that indicate the username of the user |
| password | VARCHAR | 20 |  |  |  | A field that indicate the password of the user |
| status | ENUM(‘Active’ ‘Nonactive ) |  |  |  |  | A field that indicates whether the user is active or not |
| Voting status | VARCHAR | 10 |  |  |  | A field that indicates if the user has voted or not |
| role | INT | 1 |  |  |  | A field that indicate the role of the user |
| date\_joined | TIMESTAMP |  |  |  |  | A field that indicate the date and time the user is added to the database |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Profile | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each User |
| userid | INT | 4 |  | Foreign |  | A unique foreign key that relate user table and profile table |
| firstname | VARCHAR | 20 |  |  |  | A field that indicate the firstname of the user |
| Lastname | VARCHAR | 20 |  |  |  | A field that indicate the lastname of the user |
| othername | VARCHAR | 20 |  |  |  | A field that indicate the othername of the user |
| email | VARCHAR | 50 |  |  |  | A field that indicate the email of the user |
| dob | VARCHAR | 20 |  |  |  | A field that indicates the date of birth of the user |
| gender | VARCHAR | 10 |  |  |  | A field that indicate the gender of the user |
| facultyid | INT | 4 |  |  |  | Is a foreign key that relates the faculty and user |
| departmentid | INT | 4 |  |  |  | Is a foreign key that relates the department and user |
| passport | VARCHAR | 100 |  |  |  | A field that indicate the passport of the user |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Candidate | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| cid | INT | 4 |  | Primary |  | A unique id to  identify each Candidate |
| userid | INT | 4 |  | Foreign |  | A unique foreign key that relate user table and Candidate table |
| Positionid | INT | 4 |  |  |  | A foreign key that relates position to the candidate |
| Session | VARCHAR | 20 |  |  |  | A field that indicates the session where the candidate is aspiring for position |
| Vote | INT | 4 |  |  |  | A field that indicate the number vote the candidate has |
| facultyid | INT | 4 |  |  |  | A foreign key that relates faculty table to candidate table |
| departmentid | INT | 4 |  |  |  | A foreign key that relates the department table to candidate table |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Authentication | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each Authentication |
| Userid | INT | 4 |  | Foreign |  | A unique foreign key that relate user table and Authentication table |
| Pin | VARCHAR | 4 |  |  |  | A field that indicate the OTP send to the user verification |
| Time | TIMESTAMP |  |  |  |  | A field that indicates the time the OTP was sent |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Position | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each position |
| Position | VARCHAR | 100 |  |  |  | A field that indicate position |
| Table Name: Faculty | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each Faculty |
| Faculty | INT | 4 |  |  |  | A field that indicate faculties |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Department | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each department |
| Facultyid | INT | 4 |  | Foreign |  | A unique foreign key that relate faculty table and department table |
| Department | VARCHAR | 50 |  |  |  | A field that indicate the department |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: votingdetails | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each voting details |
| Candidateid | INT | 4 |  |  |  | A unique foreign key that relate candidate table and votingdetails table |
| Voterid | INT | 4 |  |  |  | A foreign key that relates user table and votingdetails table |
| Datejoint | TIMESTAMP |  |  |  |  | A field that indicate the time voter cast his/her vote |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: Comment | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each comment |
| Userid | INT | 4 |  | Foreign |  | A unique foreign key that relate user table and comment table |
| Message | VARCHAR | 250 |  |  |  | A field that indicate the comment of the user |
| Time | TIMESTAMP |  |  |  |  | A field that indicates the date and time the user commented. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name: News | | | | | | |
| Field Name | Data Type | Length | Null | Key | Reference | Description |
| id | INT | 4 |  | Primary |  | A unique id to  identify each news |
| Title | VARCHAR | 100 |  |  |  | A field that indicates the title of the news |
| Name | VARCHAR | 50 |  |  |  | A field that indicate the name of the announcer |
| Content | VARCHAR | 20 |  |  |  | A field that indicate the content of the news |
| Date | INT | 4 |  |  |  | A field that indicate the Date the news is announced |

**3.4 Tools and Technologies to be used**

This section introduced the tools and technology used in the development of the proposed SRC Online Voting System.

**3.4.1** **Hypertext markup language (HTML):**

HTML has two essential features hypertext and universality. **Hypertext** means you can create a link in a Web page that leads the visitor to any other Web page or to practically anything else on the Internet. It means that the information on the Web can be accessed from many different directions. Tim Berners-Lee, the creator of the Web, wanted it to work more like a person's brain and less like a static source of data, such as a book. While **Universality** means that because HTML documents are saved as Text Only files, virtually any computer can read a Web page. It doesn't matter if your visitors have Macintosh or Windows machines, or whether they're on a UNIX box or even a handheld device like a Palm.

HTML has the following features:

* Publish online documents with headings, text, tables, lists, photos, etc.
* Retrieve online information via hypertext links, at the click of a button.
* Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.

Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

**3.4.2** Hypertext Processor (PHP)

PHP stands for Hypertext Preprocessor: a scripting language designed specifically for use on the Web. In its early development by a man named Rasmus Lerdorf, it was called Personal Home Page tools. When it developed into a full-blown language, the name was changed to Hypertext Preprocessor, more in line with its expanded functionality. (Janet, 2007). Hypertext preprocessor is a general-purpose scripting language (server-side) that, in addition to HTML and JavaScript, used in developing web applications. PHP script arc computer files containing instructions in the PHP language that tell the computer to do things, such as display Hello on the screen or store some specified data in a database used and are used for web applications which produce HTML documents as output. When PHP statements are executed, only the output is sent by the Web server to the Web browser i.e. codes are not included in the output sent to the browser, so the PUP code is secure and transparent to the user. PHP files have extension ".php". PHP is only server-side, meaning it can't interact directly with the user's computer.

Some of the capabilities of PHP include:

* Interact with HTML forms: PHP can display an HTML form and process the information that the user types in.
* Communicate with databases: PHP can interact with databases to store information from the user or retrieve information that is displayed to the user. PHP is particularly strong in its ability to interact with databases. PHP supports pretty much every database. You tell PHP the name of the database and where it is and PHP handles the details. It connects to the database, passes your instructions to the database, and returns the database response to you.
* Generate secure Web pages: PHP allows the developer to create secure Web pages that require users to enter a valid username and password before seeing the Web page content (*Jalade, 2007*). These functionalities of PHP make me to use it in my application.

**3.4.3** JavaScript

In the beginning was HTML, and all was well, and its capability to include text and graphics of different sizes within the same document was quite compelling. As time passed, however, pages became increasingly sophisticated, so the two major Web browser companies, Netscape and Microsoft, each began to develop a scripting language for use on Web pages, a language that would allow programs to be run on the visitors' systems as they viewed the-pages. For Microsoft, it was Visual Basic Script, the same scripting language found in the Microsoft Office Suite, among others. For Netscape, it was a scripting language that looked vaguely like the popular new Java object-oriented language. Netscape won, Microsoft lost, and JavaScript is the Web's de facto scripting language (Taylor, 2004).

Some of the Features of JavaScript Are:

* JavaScript gives HTML designers a programming tool - HTML authors are normally not programmers, but JavaScript is a scripting language with a very simple syntax! Almost anyone can put small portion of code into their HTML pages
* JavaScript can react to events - A JavaScript can be set to execute when something happens, like when a page has finished loading or when a user clicks on an HTML element. This gives more flexibility.
* JavaScript can read and write HTML elements - A JavaScript can read and change the content of an HTML element
* JavaScript can be used to validate data - A JavaScript can be used to validate form data before it is submitted to a server. This saves the server from extra processing. This particular function will be extensively utilized throughout this project.
* JavaScript can be used to detect the visitor's browser - A JavaScript can be used to detect the visitor's browser, and - depending on the browser - load another page specifically designed for that browser
* JavaScript can be used to create cookies - A JavaScript can be used to store and retrieve information on the visitor's computer.

Much more interesting than variable assignment statements (JavaScript instructions that assign a value to a specified variable) are expressions, which are the real building blocks of JavaScript. Expressions can evaluate to a Boolean (as in "if this condition is true, then") or can evaluate to a string or numeric expression. Based on the above-mentioned features of JavaScript I make use of in my application.

* + 1. **Cascading style sheets (CSS):**

Cascading style sheet **(CSS)**is the aspect of web design which allows the modification of web pages in a website. It enables a user to re-define how many markups can be rendered. It also essentially allows us to create its own tags to transform a web page into exactly the format layout we seek. CSS can be specified with the specified tags in the document body or at the top of the document within the block or combined with name or containers in the document body. Some features of CSS are:

* Better control of fonts including color.
* Better control of block-level layout (indents, margins, alignment, etc.)
* Better control of inline layout, particularly with regard to diagrams and related text.

**3.4.5 MySQL Workbench**

MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

**3.4.5 APACHE**

Apache is a web server which is commonly used with PHP and MSQL, it very important because it serves as the server where the application runs. Apache is an open-source HTTP Web server software (Source code is freely available and can be shared.). It is currently the most popular web server on the NET. It is usually run Unix operating system version like Linux or BSD, but it can also be run on Windows. It is a full-featured server with many powerful add-ons freely available. Apache operates on your Web server. Its major work is parsing any file requested by a browser and display the correct results according to the codes within the files.

Apache is to a certain extent powerful and executes or achieves virtually any task that a Webmaster may require.

**CHAPTER FOUR**

**SYSTEM IMPLEMENTATION AND TESTING**

**4.0 Introduction**

This chapter explores various factors that are necessary for the implementation of the proposed system. It also focuses on system implementation and system testing of Online Voting System. The purpose of this chapter is to implement the system requirements, design specifications and models from the previous chapters.

**4.1 Implementation**

Implementation is the stage where the theoretical design of the system is turned into a working system. System implementation is the last stage of the system development which involves the conversion of the system requirement into an executable system.

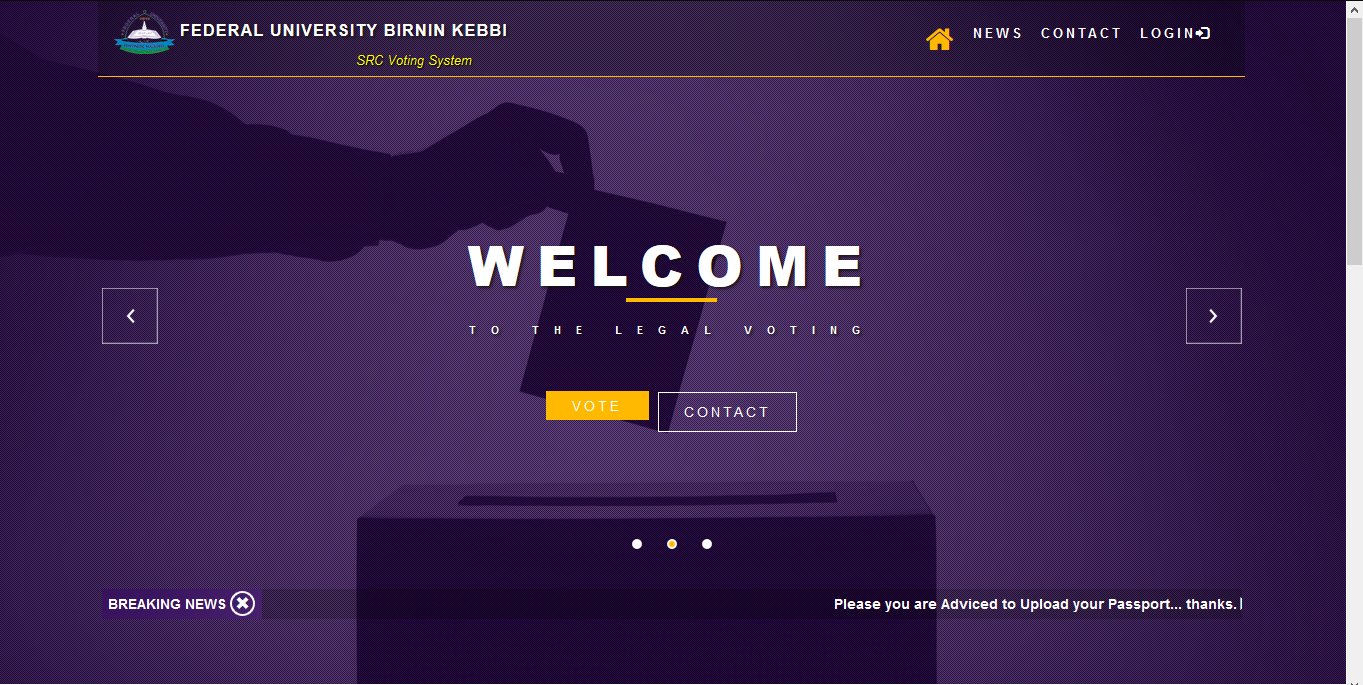
**4.2 Site Development**

This voting system is successfully designed with the view of providing possible remedies toward solving some of the problems surrounding Student Representative Council (SRC). The following is the description some of the modules designed.

**4.3 System Module**

The system module contains all the system pages that carry out specific functions. The modules of these systems are:

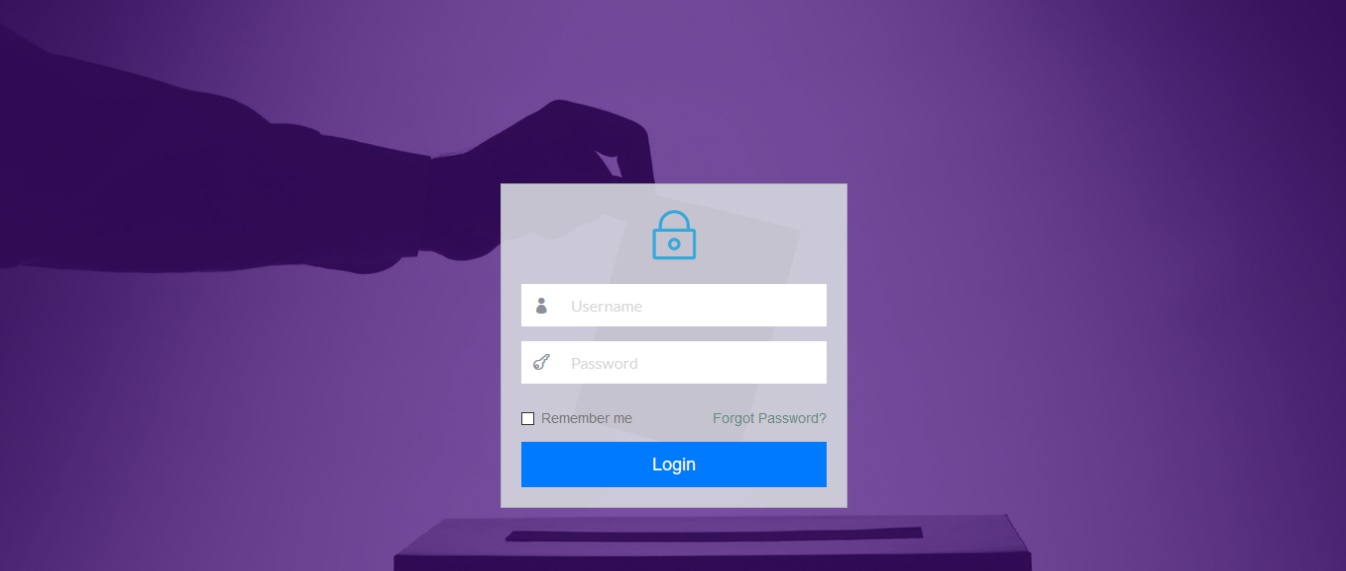
**Index page (Home Page):**

**` `**

**Fig 4.1 Homepage**

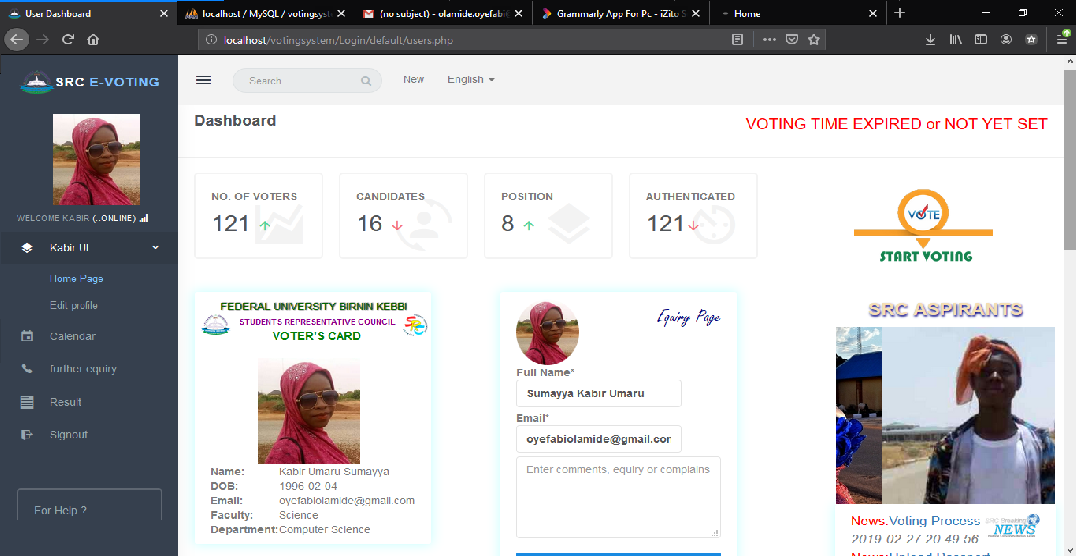
**Fig. 4.1** shows the main homepage (or landing page) of the application. The page contains welcome address and links to navigate to other pages.

**Login page**

****

**Fig. 4.2 Login page**

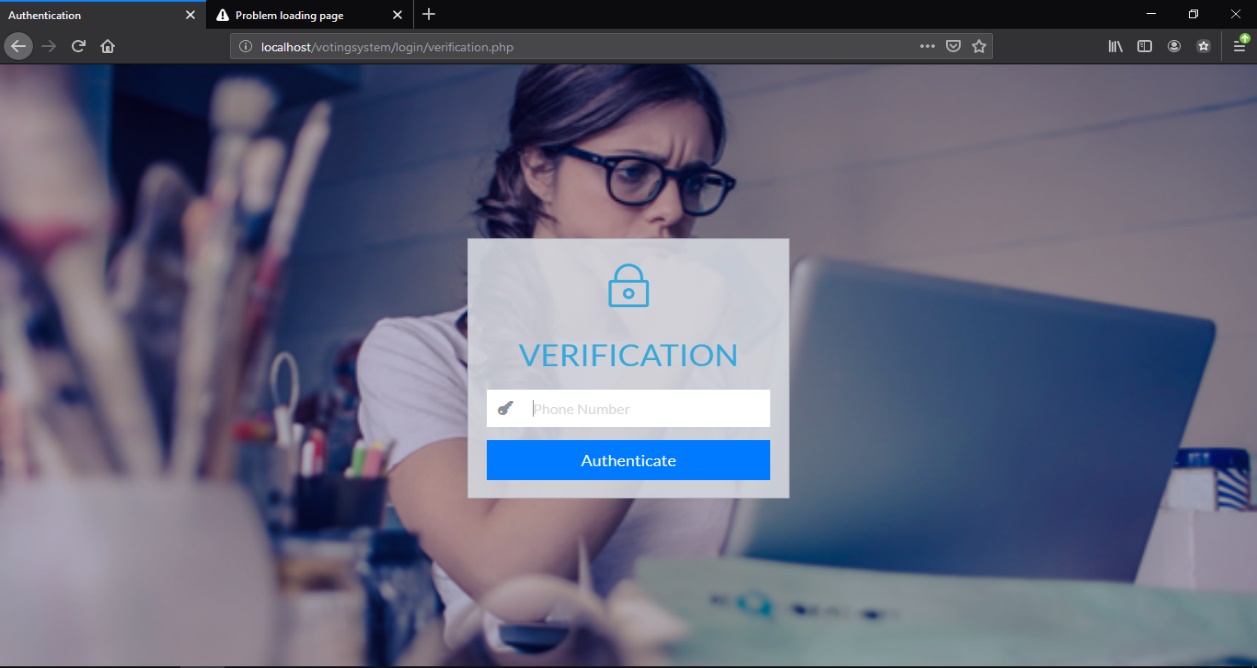
**Fig 4.2** shows the Login page where the user enters the Correct Username and Password before having access to the system

**Student page (Voter)**

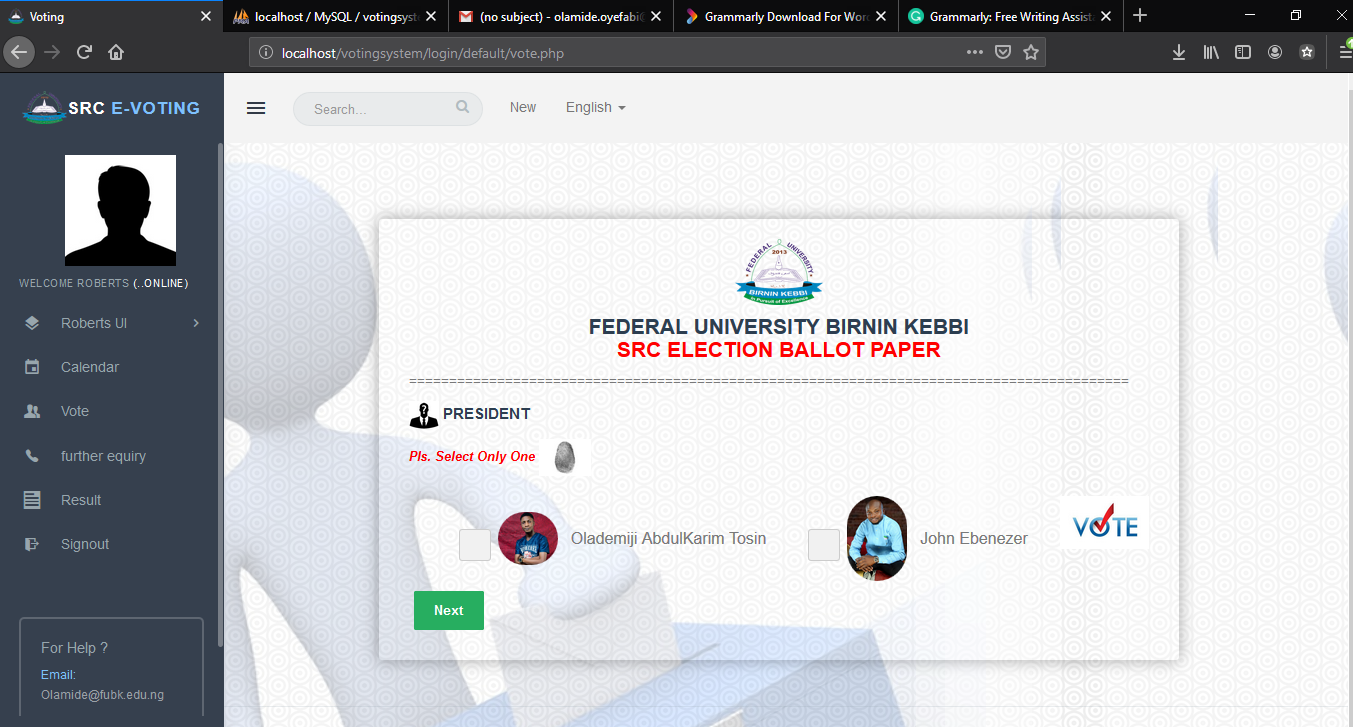
**Fig 4.3 Student page**

**Fig. 4.3** Shows the Student page which is the voter where the student (voter) have access to other pages to visit other pages and Start voting by clicking on the start voting button. The page contains news tab, SRC aspirants’ names and pictures with the post contesting for and links to navigate to other pages.

**Verification**

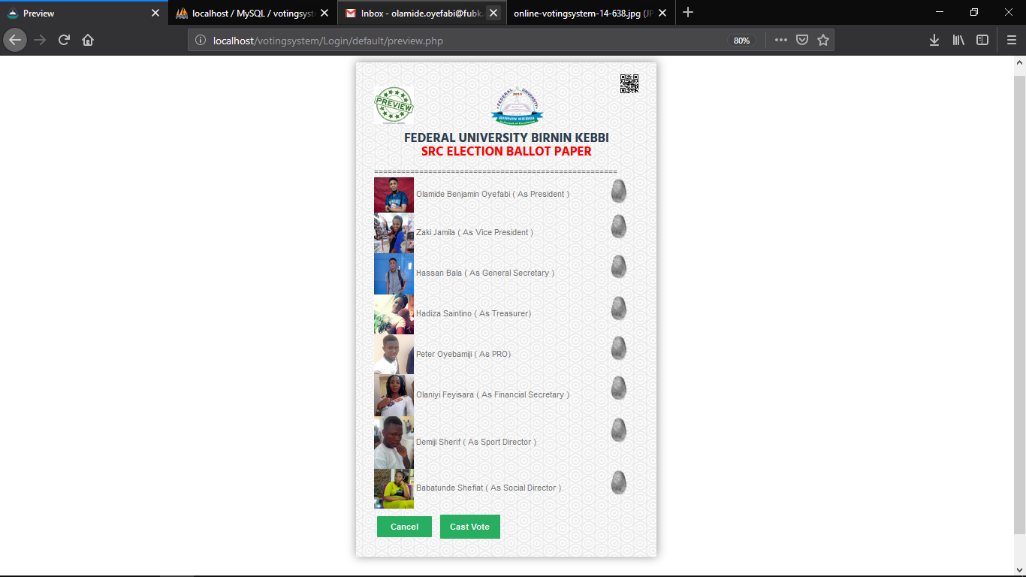


**Fig. 4.4** show the verification page where the user is verified by sending an OTP to his registered number and when the user enters the OTP then he will be allowed to proceed to voting page.

**Voting page**

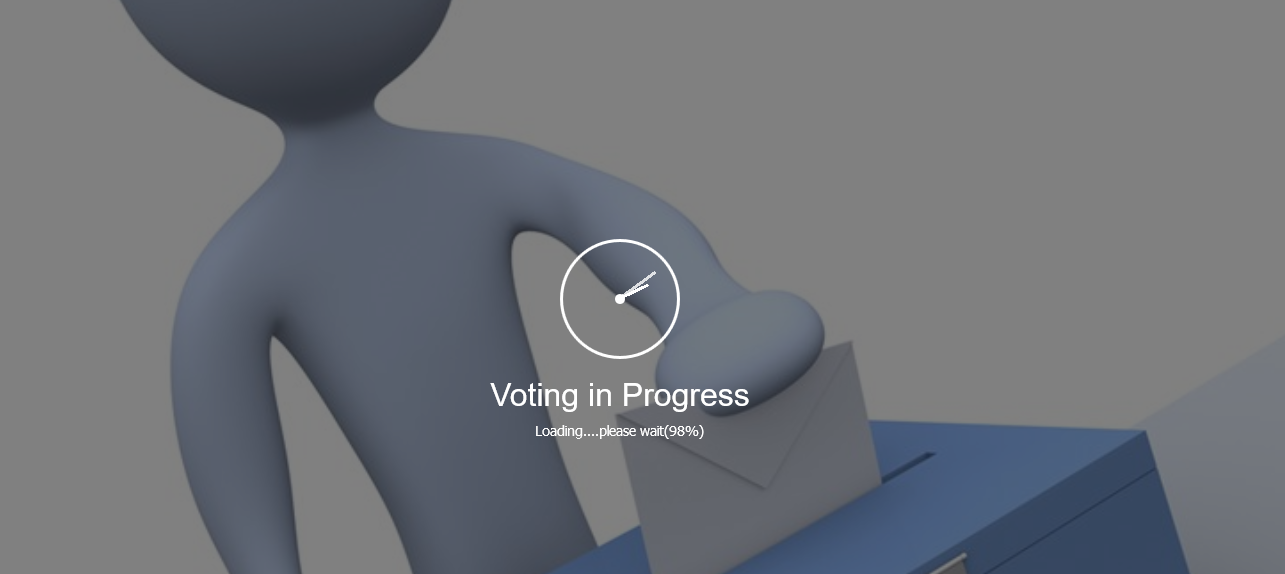
**Fig. 4.5 Voting page**

**Fig. 4.5** Shows the voting page where the voter selects his candidate for each Position and click on the next button to move to the next position where he/she can as all select candidate until it gets to the final position but the user can as well click previous button to change candidate before he/she clicks the submit button for casting.

**Preview page**

**Fig 4.6 Preview page**

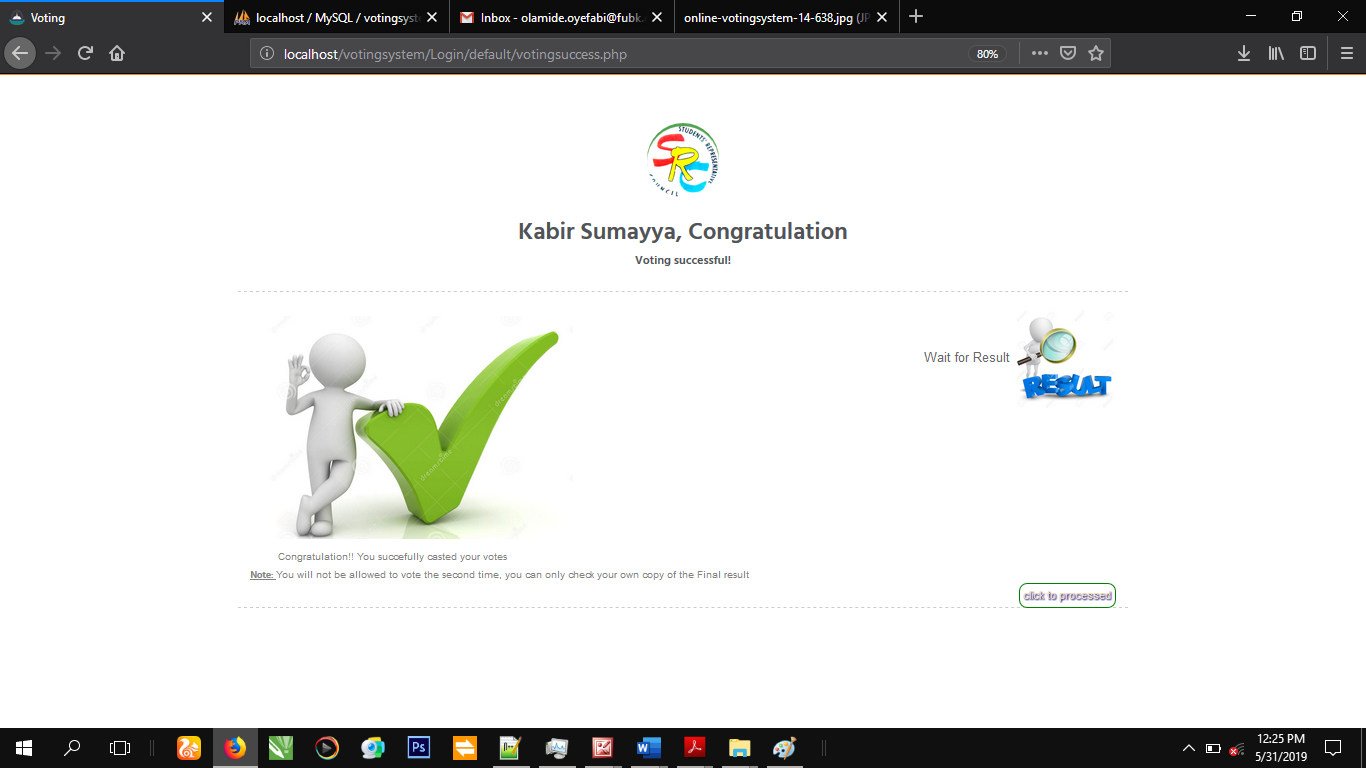
**Fig. 4.6** Show preview of the candidate the user has selected for each position. The Full name, picture and the position of each candidate the user selected will appear for verification before casting of votes and the user can also cancel the preview and start all over in case of wrong selection.

**Casting of votes page**

**Fig 4.7 Casting of vote page**

**F.g. 4.7** when the cast button is clicked on the preview page, this page popup loading in the votes input the database.

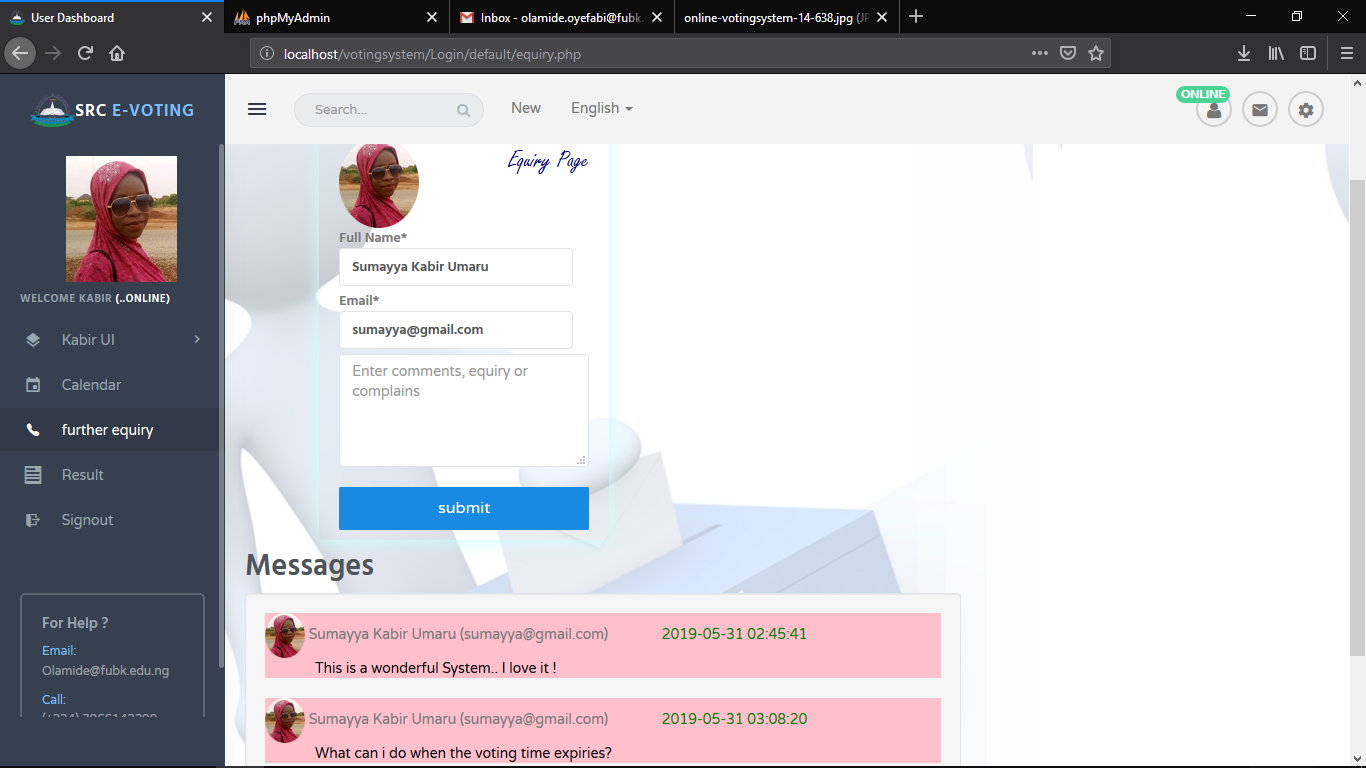
**Votes Cast success page**



**Fig 4.8 Votes Cast Success page**

Fig. 4.8 shows that the voter has successfully cast his or her vote and will not be allowed to vote again.

**Comment Page**

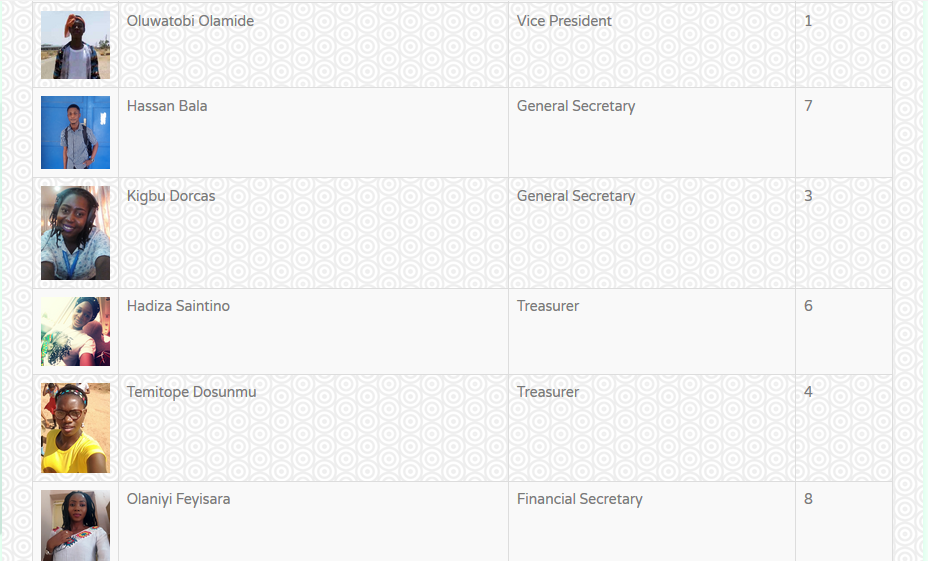


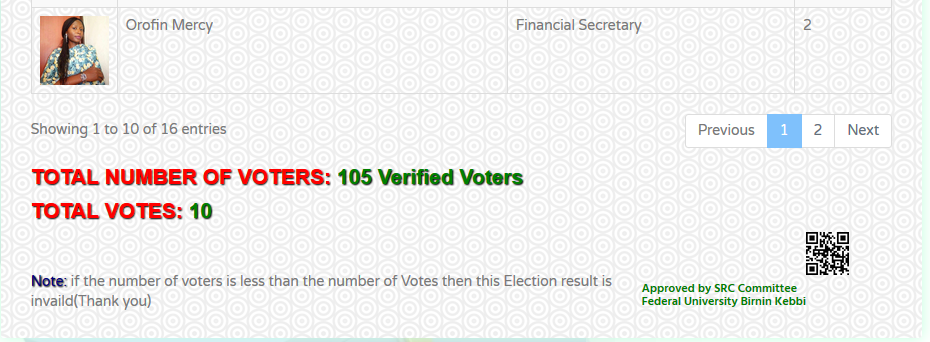
**Fig. 4.9 Comment Page**

**Fig. 4.9** Show the comment page where the user can comment on the page

**Election Result**



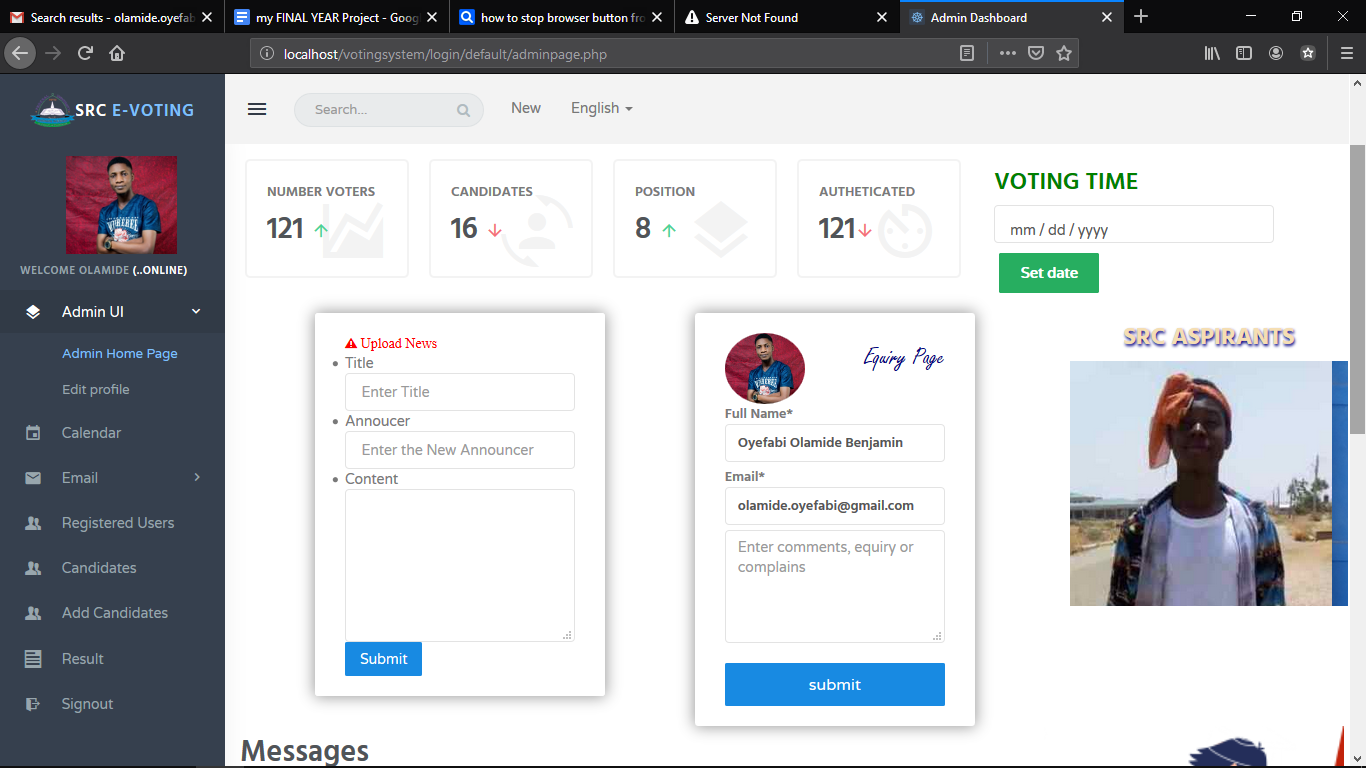




**Fig 4.10 Result page**

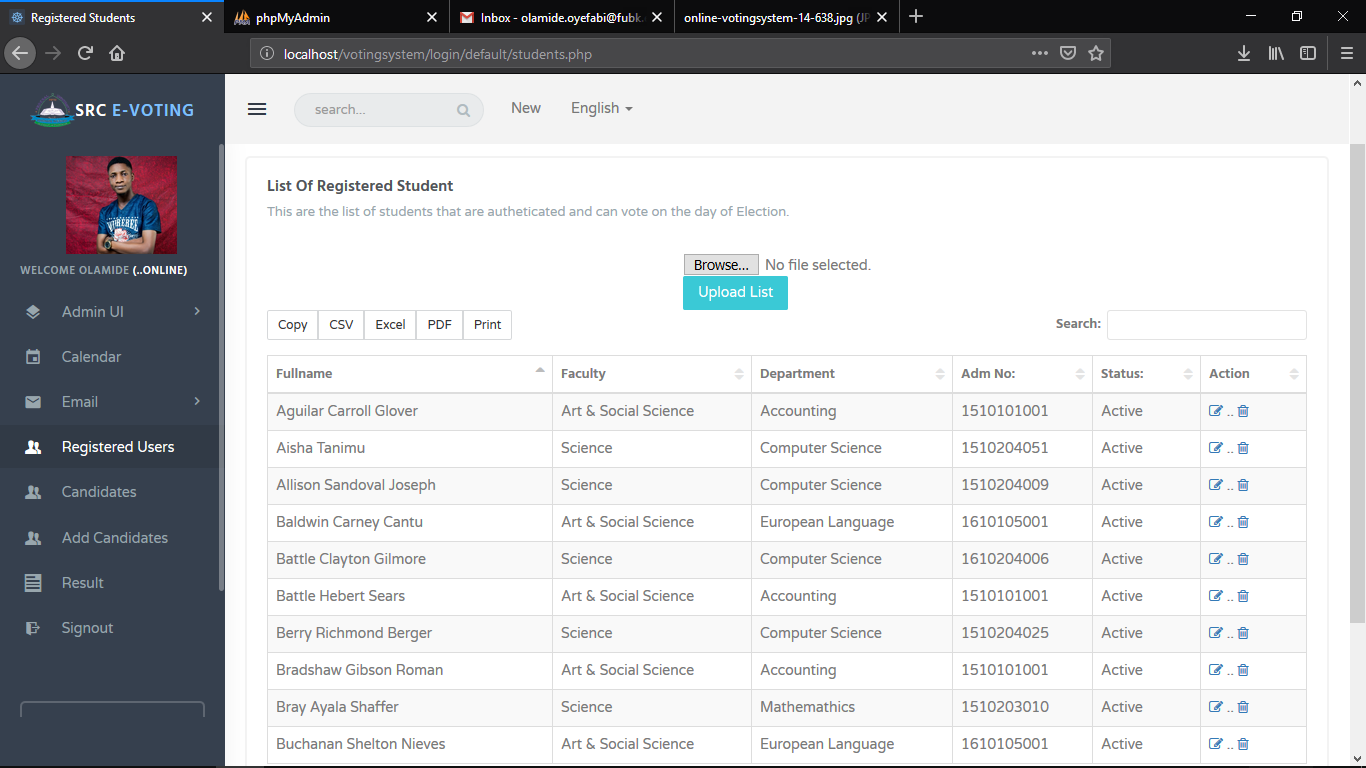
**Fig 4.10** shows the result of the election after the Voting time expiry

**Admin Page**



**Fig. 4.11Admin Page**

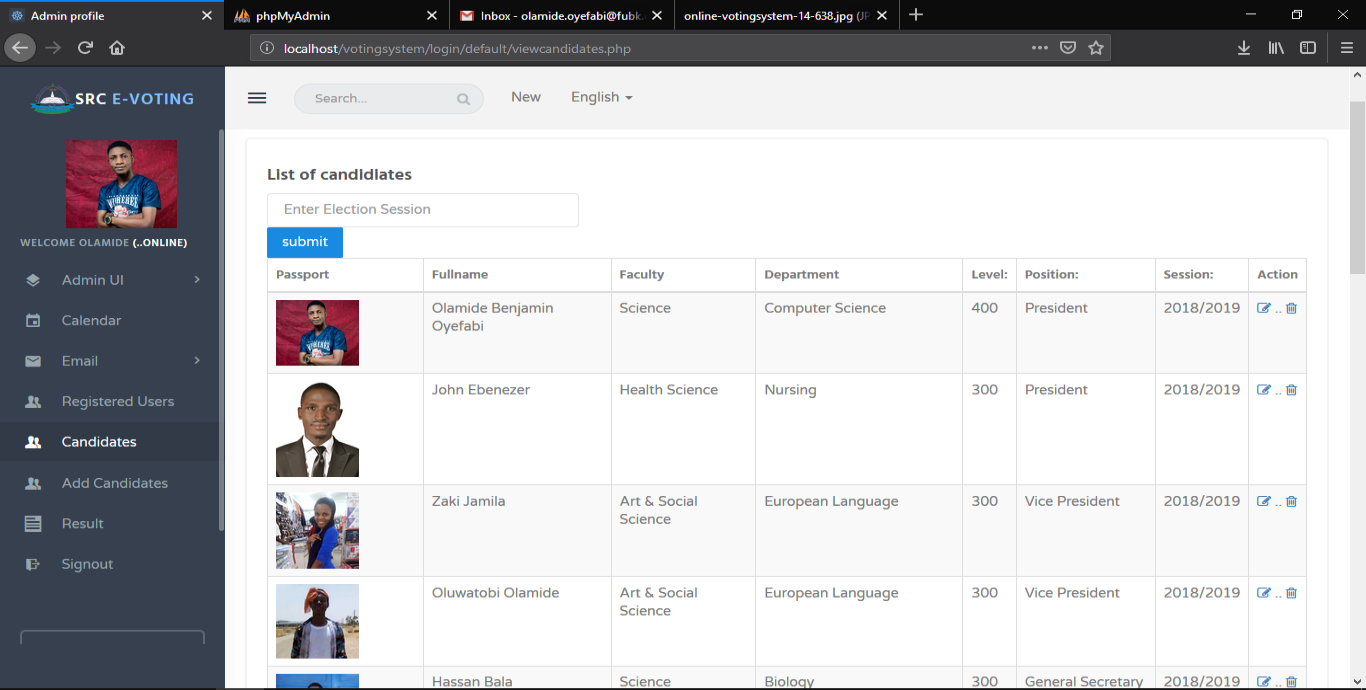
**Fig. 4.9** Shows the Admin page of the application where admin will have access to visit other pages and perform their actions. The page contains Number of Voters, Candidate positions and authenticated User and also with links to navigate to other pages

**Registered Students**

**Fig.4.12 Registered Students**

**Fig.4.10** shows the details of the Registered Students and the admin can perform edit of user profile or even delete student using the Action button by the side of each student details.

**List of Candidate**



**Fig. 4.13 List of registered candidates**

**4.11** shows the list of candidates registered with their aspiring Position. Admin can also edit or Delete any candidate using the action button at each side of candidate details.

**Add Candidate**

**Fig 4.14. Adding Candidate Page**

**Fig. 4.12** Show where the admin adds a candidate with his or her aspiring position.

**4.4 Source Code**

The source code for the proposed SRC Online Voting System is also attached in Appendix A.

**4.5 Testing**

Testing is the process that is carried out to ensure that the system conforms to the specification and meets the requirements of the clients and administrators. Testing had been conducted not only in the end but also during the development of the prototype system. Functional and interface testing were carried out for the module or for the whole system. Each and every link had been checked to make sure all the links are working correctly. Interface testing is carried out to identify that the interface works correctly and faults are not created because of interface errors.

**4.5.1 Unit Testing**

Unit Testing is to test software in terms of a unit, a module, a function, a specific section of code. This testing occurs while the software is being developed and before completion (David Fletcher, (2000). For Unit Testing, test cases are designed to verify that an individual unit implements all design decisions made in the unit's design specification. A thorough unit test specification should include positive testing where the unit does what it is supposed to do, and also negative testing where the unit does not do anything that it is not supposed to do. Table 4.1 shows the Unit

**Testing for the Administrator login module.**

|  |  |  |
| --- | --- | --- |
| **Test Procedure** | **Output/Error** | **Analysis of the Test Result** |
| Login as admin with valid login details, i.e. username and password | Admin is redirected to the admin homepage. | Successfully redirected to the admin home after the system checks the validity of the login details, i.e. username and password |
| Login as admin with  invalid login details | Wrong login details | Login denied and an error message of login with wrong details is displayed |
| Login with empty fields | Required fields cannot be empty | Login denied and an error message is displayed requesting to fill the required fields |

**Table 4.1 Unit Testing for Admin Login**

**4.5.2 Integral Testing**

Upon completion of Unit Testing, Integration Testing will begin. The purpose is to ensure the distinct components of the application still work in accordance with user requirements. Test sets will be developed with the purpose of exercising the interfaces between the components. This activity is to be carried out clients and administrators. Integration test will be complete when actual results and expected results are either inline or differences are explainable or acceptable based on the user input.

**4.5.3 System Testing**

Upon completion of Integration Testing, System Testing will begin. During system testing, the complete system is configured in a controlled environment to validate its accuracy and completeness in performing the functions as designed. The system test will simulate production as it will occur in the “production-like” test environment and test every function of the system that will be required in production. It is also important that the validation of the system meets the functional and non-functional requirements.

**CHAPTER FIVE**

**SUMMARY, CONCLUSION AND RECOMMENDATION**

**5.0 Introduction**

This chapter discusses the outcome of this entire project i.e. summary, conclusion and future works of the research. Knowledge in terms of concept, theory, technical and practical aspects of SRC Online voting System has been gained.

**5.0 Summary**

Based on the research objectives that have been stated in the earlier chapter, the following are the achievements; The first objective has been achieved successfully, which is to develop a voting system that uses computer technology to simplify the voting process to replace the appointing method adopted in the school. Some of the similar existing systems were also explored and research gap was clearly stated in chapter 2​ after reviewing a significant number of recent findings. Then achieved objective was identifying the effective methodology to improve the existing systems. This objective has been highlighted in ​chapter3​, based on the information gathered, relevant features of various components and methods needed for the proposed SRC voting system has been designed.

However, the next objectives were achieved by implementing the system designs from the previous chapters. An interactive interface that allow students (voters) to easily elect their own candidate without much stress has been successfully implemented and tested; the implemented system can also authenticate users before the voting excise commences and after the voting excise date has expired, users can access the result of the election that has be conducted.

**5.1 Conclusion**

In the course of this research, it can be concluded that SRC Online voting system can help to increase number of voters as individuals will find it easier and more convenient to vote. It increases user-level security by using OTP that is send to the user's mobile number and mail to avoid blackmailing. It also helps to reduce manual processes and thereby reduce human errors while calculating votes. The result is evaluated fast.

**5.2 Recommendations**

Based on the findings of the study, it is recommended that the appointing method adopted should be replaced by-election in which students can elect their candidates and also, The Student Representative Council should be replaced with Student Union Government to make them more effective.

**References**

Manjusha Vijay Amritsar, et al (2016). Secure Online Voting System

Sanjay, K.., and Singh, M. (2013). " Design A Secure Electronic Voting System Using

Fingerprint Technique" International Journal of Computer Science Issues Vol. 10(4), pp 192-199.

Olaniyi, O.M, O.T Arulogun, E.O, Omidiora, A Omotoso, Ogungbemi O.B. (2012), ‖ Design

of A Secured Model for Electronic Voting System Using Stegano-Cryptographic Approach ‖, Proceedings of the 7th International Conference on ICT Applications, Application of ICT to Teaching, Research, and Administration (AICTTRA 2012), National Defense College Abuja, pp 84-89.

Olayemi M. Olaniyi, Taliha A. Folorunso, Aliyu Ahmed, Olugbenga Joseph (2016), Design

of secure Electronic Voting System using fingerprint Biometrics and Crypto-watermarking Approach. Federal University of Technology/ Computer Engineering Department, Minna, Nigeria.

Berners, S. T., & Web, W. W. (2018). History of the Web, (March 1989), 1–5.

Gemino, A. & Parker, D., (2009). "Use case diagrams in support of use case modeling:

Deriving understanding from the picture", *Journal of Database Management*, 20(1),

pp.1-24.

David, F., (2000). *Software Testing and Quality Assurance,* 31/7/2000, Pp. 2-4.

Whitten, J.L *et al* ., (2002). *Systems Analysis and Design Methods,* 5th Edition, 2002.

Shafi‘í, M, Adebayo, O. S, Damian, O., Mohammed, D. (2013). "The Design and

Development of Real-Time E-Voting System in Nigeria with Emphasis on Security and Result Veracity, International Journal of Network and Computer Security. MECS (http://www.mecs-press.org/) Vol. 5, pp 9-18

D. Akinmosin D., G.G.O. Egbedokun G.G.O. and Ibitowa F.O (2011),‖An Extended

Multifactor Authentication in Mobile Financial Transaction Using User Authentication Module with Multilayered Encryption Algorithms‖, African Journal of Computer and Information Technology ICT (Journal of IEEE Nigeria Computer Section), Vol. 4 (2), pp 17-24.

Gallegos G., Gomez R., and Sanchez G, (2010), "Electronic Voting using Identity-Based

Cryptography", Fourth International Conference on Digital Society, pp 31-36.

Enokela, J. A. (2010), ―Security of Programs and Data for an Electronic Voting System‖,

Pacific Journal of Science and Technology, Vol. 11(2), pp. 283-287.

Gunjal B., and Mali S, (2013). ―Secure e-voting system with Biometric and Wavelet-Based

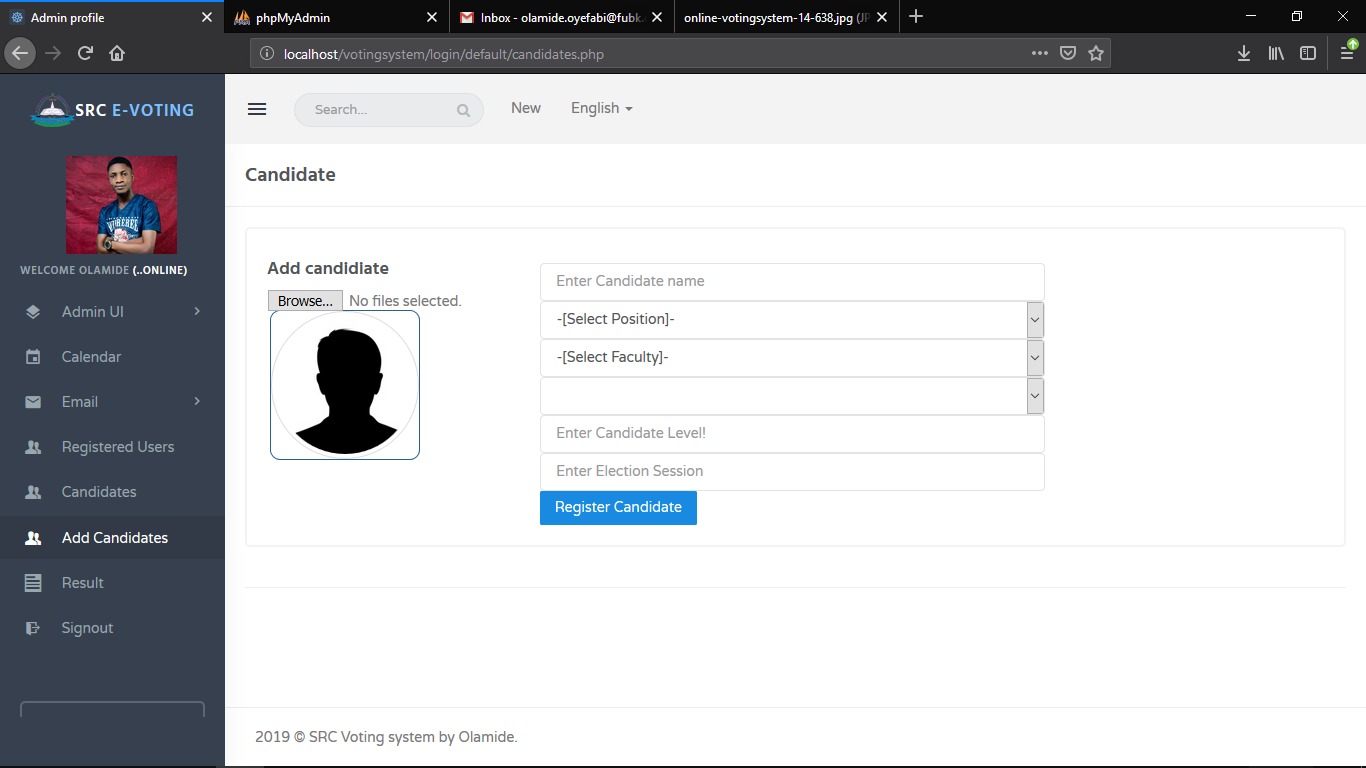
Watermarking Technique in Ycgcb color space. ‖n IEEE International Conference on Information Technology, pp 1-6

Sable, S. and Bombale U. L. (2014), "Cryptography Based Secured E-voting System Using

ARM for Cell Phone and Internet Application", International Journal of Computer Engineering and Applications, Vol. 7 (1), pp. 1-6.

Prashanth P., and Swaroop S, (2012), "Requirements for an Electronic Voting System."

Department of Computer Science the Johns Hopkins University, pp. 1-2.

**Appendix I**

**Source code (Html with little Php Code)**

**<!DOCTYPE html>**

**<html lang="en">**

**<?php error\_reporting(0);**

**session\_start();**

**if(!isset($\_SESSION['id'])){**

**//not loggin**

**header('Location:../login.php'); exit(); }**

**require('../conn/connection.php');**

**$userid=$\_SESSION['id'];**

**$sql="SELECT \* from profile join users join department join faculty ON profile.userid=users.id and profile.departmentid=department.id and profile.facultyid=faculty.id where profile.userid=:userid";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':userid', $userid);**

**$stmt->execute();**

**$rows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$row=$rows[0];**

**$sql="SELECT \* from news order by date DESC ";**

**$stmt=$pdo->prepare($sql);**

**$stmt->execute();**

**$nrows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$casql="SELECT \* from candidates JOIN profile, postions where candidates.userid=profile.userid and candidates.postionid=postions.id order by postionid";**

**$stmt=$pdo->prepare($casql);**

**$stmt->execute();**

**$candidates=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$sql="SELECT \* from users";**

**$stmt=$pdo->prepare($sql);**

**$stmt->execute();**

**$rows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$count=count($rows);**

**$sql="SELECT \* from postions";**

**$stmt=$pdo->prepare($sql);**

**$stmt->execute();**

**$rows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$countp=count($rows);**

**$sql="SELECT \* from candidates";**

**$stmt=$pdo->prepare($sql);**

**$stmt->execute();**

**$rows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$countd=count($rows);**

**$comment="SELECT \* from comments join users join profile ON comments.userid=users.id and comments.userid=profile.userid where comments.userid=users.id ORDER by time DESC";**

**$stmt=$pdo->prepare($comment);**

**$stmt->execute();**

**$comments=$stmt->fetchAll(PDO::FETCH\_ASSOC); ?>**

**<head>**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<meta name="description" content="A fully featured admin theme which can be used to build CRM, CMS, etc.">**

**<meta name="author" content="Coderthemes">**

**<link rel="icon" href="../../images/logo.png" />**

**<!-- App title -->**

**<title>User Dashboard </title>**

**<!--Morris Chart CSS -->**

**<link rel="stylesheet" href="../plugins/morris/morris.css">**

**<!-- App css -->**

**<link href="assets/css/bootstrap.min.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/core.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/components.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/icons.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/pages.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/menu.css" rel="stylesheet" type="text/css" />**

**<link href="assets/css/responsive.css" rel="stylesheet" type="text/css" />**

**<link rel="stylesheet" href="../plugins/switchery/switchery.min.css">**

**<script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>**

**<script src="https://oss.maxcdn.com/libs/respond.js/1.3.0/respond.min.js"> </script>**

**<script type="text/javascript">**

**function noBack()**

**{**

**window.history.forward()**

**}**

**noBack();**

**window.onload = noBack;**

**window.onpageshow = function(evt) { if (evt.persisted) noBack() }**

**window.onunload = function() { void (0) }**

**</script>**

**<!-- Start WOWSlider.com HEAD section -->**

**<link rel="stylesheet" type="text/css" href="engine1/style.css" />**

**<script type="text/javascript" src="engine1/jquery.js"></script>**

**<!-- End WOWSlider.com HEAD section -->**

**</head>**

**<body class="fixed-left" >**

**<!-- Begin page -->**

**<div id="wrapper">**

**<!-- Top Bar Start -->**

**<div class="topbar">**

**<!-- LOGO -->**

**<div class="topbar-left">**

**<a href="" class="logo"><img src="../../images/logo.png" style="width:20%"><span style="font-size:70%">SRC<span> E-Voting</span></span></a>**

**</div>**

**<!-- Button mobile view to collapse sidebar menu -->**

**<div class="navbar navbar-default" role="navigation">**

**<div class="container">**

**<!-- Navbar-left -->**

**<ul class="nav navbar-nav navbar-left">**

**<li>**

**<button class="button-menu-mobile open-left waves-effect">**

**<i class="mdi mdi-menu"></i>**

**</button>**

**</li>**

**<li class="hidden-xs">**

**<form role="search" class="app-search">**

**<input type="text" placeholder="Search..."**

**class="form-control">**

**<a href=""><i class="fa fa-search"></i></a>**

**</form>**

**</li>**

**<li class="hidden-xs">**

**<a href="#" class="menu-item">New</a>**

**</li>**

**<li class="dropdown hidden-xs"> <a data-toggle="dropdown" class="dropdown-toggle menu-item" href="#" aria-expanded="false">English**

**<span class="caret"></span></a>**

**<ul role="menu" class="dropdown-menu">**

**<li><a href="#">German</a></li>**

**<li><a href="#">French</a></li>**

**<li><a href="#">Italian</a></li>**

**<li><a href="#">Spanish</a></li>**

**</ul>**

**</li>**

**</ul>**

**</div><!-- end container -->**

**</div><!-- end navbar -->**

**</div>**

**<!-- Top Bar End -->**

**<!-- ========== Left Sidebar Start ========== -->**

**<div class="left side-menu">**

**<div class="sidebar-inner slimscrollleft">**

**<!--- Sidemenu -->**

**<div id="sidebar-menu"> <ul>**

**<li class="menu-title"><img src="../uploads/<?php echo $row['passport']?>" style="width:60%; position: relative; left:25%"><br><br> welcome <?php echo $row['firstname']?> <span style="color:white">(..ONLINE) <i class="glyphicon glyphicon-signal"></i></span></li>**

**<li class="has\_sub">**

**<a href="javascript:void(0);" class="waves-effect"><i class="mdi mdi-layers"></i><span> <?php echo $row['firstname']?> UI </span> <span class="menu-arrow"></span></a>**

**<ul class="list-unstyled">**

**<li><a href="users.php">Home Page</a></li>**

**<li><a href="userprofile.php">Edit profile</a></li>**

**</ul>**

**</li>**

**<li>**

**<a href="userscalendar.php" class="waves-effect"><i class="mdi mdi-calendar"></i><span> Calendar </span></a>**

**</li>**

**<!--<li><a href="loading.html" id="clck" class="waves-effect"><i class="ion-person-stalker "></i><span> Vote </span></a></li>**

**<li><a href="equiry.php" class="waves-effect"><i class="ion-ios7 telephone"></i><span> further equiry </span></a></li>**

**<li><a href="usersvotingresult.php" class="waves-effect"><i class="ion-android-note"></i><span> Result </span></a></li>**

**<li><a href="../../index.php" class="waves-effect"><i class="ion-log-out "></i><span> Signout </span></a></li>**

**</ul>**

**</div>**

**<!-- Sidebar -->**

**<div class="clearfix"></div>**

**<div class="help-box">**

**<h5 class="text-muted m-t-0">For Help ?</h5>**

**<p class=""><span class="text-custom">Email:</span> <br/> Olamide@fubk.edu.ng</p>**

**<p class="m-b-0"><span class="text-custom">Call:</span> <br/> (+234) 7066142299</p>**

**</div>**

**</div>**

**<!-- Sidebar -left -->**

**</div>**

**<div class="content-page">**

**<!-- Start content -->**

**<div class="content" >**

**<div class="container" >**

**<div class="row" >**

**<div class="col-xs-12">**

**<div class="page-title-box">**

**<h4 class="page-title">Dashboard</h4>**

**<ol class="breadcrumb p-0 m-0">**

**<div id="countdown"> </div>**

**</ol>**

**<p id="demo" style="float:right;font-size:20px; margin-top:0px; color: Red;"></p>**

**<div class="clearfix"></div>**

**</div>**

**</div>**

**</div>**

**<!-- end row -->**

**<div class="row" >**

**<div class="col-lg-2 col-md-4 col-sm-6">**

**<div class="card-box widget-box-one">**

**<i class="mdi mdi-chart-areaspline widget-one-icon"></i>**

**<div class="wigdet-one-content">**

**<p class="m-0 text-uppercase font-600 font-secondary text-overflow title="Statistics">No. Of Voters</p>**

**<h2><?php echo $count;?> <small><i class="mdi mdi-arrow-up text-success"></i></small></h2>**

**</div>**

**</div>**

**</div><!-- end col -->**

**<div class="col-lg-2 col-md-4 col-sm-6">**

**<div class="card-box widget-box-one">**

**<i class="mdi mdi-account-convert widget-one-icon"></i>**

**<div class="wigdet-one-content">**

**<p class="m-0 text-uppercase font-600 font-secondary text-overflow" title="User Today">Candidates</p>**

**<h2><?php echo $countd;?> <small><i class="mdi mdi-arrow-down text-danger"></i></small></h2>**

**</div>**

**</div>**

**</div><!-- end col -->**

**<div class="col-lg-2 col-md-4 col-sm-6">**

**<div class="card-box widget-box-one">**

**<i class="mdi mdi-layers widget-one-icon"></i>**

**<div class="wigdet-one-content">**

**<p class="m-0 text-uppercase font-600 font-secondary text-overflow" title="User This Month">Position</p>**

**<h2><?php echo $countp;?> <small><i class="mdi mdi-arrow-up text-success"></i></small></h2>**

**</div>**

**</div>**

**</div><!-- end col -->**

**<div class="col-lg-2 col-md-4 col-sm-6">**

**<div class="card-box widget-box-one">**

**<i class="mdi mdi-av-timer widget-one-icon"></i>**

**<div class="wigdet-one-content">**

**<p class="m-0 text-uppercase font-600 font-secondary text-overflow" title="Request Per Minute">Authenticated</p><h2> <?php echo $count;?><small><i class="mdi mdi-arrow-down text-danger"></i></small></h2>**

**</div>**

**</div>**

**</div>**

**<div class="col-lg-2 col-md-4 col-sm-6" >**

**<div class="card-box widget-box-one" style="width:300px; border-color:white">**

**<a href="../verification.php" id="click" ><button id="button" >Start Voting</button> </a>**

**</div>**

**</div>**

**<!-- end col -->**

**<!-- Start WOWSlider.com BODY section -->**

**<div id="wowslider-container1" style="width:280px; float:right; z-index: 3;">**

**<h3 style="text-align:center; font-weight:bold; color:wheat; text-shadow:1px 2px 2px navy">SRC ASPIRANTS</h3>**

**<fieldset style="width: ; float:left; margin-left:1%; background:url()" >**

**<center><span style="color: #006400; text-shadow: 1px 0px 1px black; font-size: 100%;" media="screen"> FEDERAL UNIVERSITY BIRNIN KEBBI</span> <br><center><img src="../../images/logo.png" style="width:15%; float:left; margin-left:-5%"><span style="color:#8B008B; font-weight:bold; font-size:80%"> STUDENTS REPRESENTATIVE COUNCIL <img src="../../images/src.png" style="width:12%; float:right; margin-right:-6%"><br> <span style="color:green; font-size:130%"><b> VOTER'S CARD</b></span></span></center> </a>**

**<br>**

**<div>**

**</div>**

**<a href="userprofile.php"><img src="../uploads/<?php echo $row['passport']?>" style="width:50%; float:left; margin-left:23%; "> </a>**

**<div >**

**<table style="width: 100%;" cellpadding="10%">**

**<tr>**

**<th>Name: </th>**

**<td> <?php echo $row['firstname']." ". $row['othername']." ". $row['lastname'];?></td>**

**</tr>**

**<tr>**

**<th>DOB: </th>**

**<td><?php echo $row['dob'];?></td>**

**</tr>**

**<tr>**

**<th>Email: </th>**

**<td><?php echo $row['email'];?></td>**

**</tr>**

**<tr>**

**<th>Faculty:</th>**

**<td><?php echo $row['faculty'];?></td>**

**</tr>**

**<tr>**

**<th>Department:</th>**

**<td><?php echo $row['department'];?></td>**

**</tr>**

**</table>**

**</fieldset>**

**<fieldset style=" float:left; margin-left:8%; background:url()" >**

**<form method="POST" action="equiryreg.php">**

**<h5 style="float:right; color:navy; font-size:1.8em; font-family:Freestyle Script;">Equiry Page</h5>**

**<img src="../uploads/<?php echo $row['passport']?>" style="border-radius:50%; width:80px" >**

**<input type="hidden" name="passport" value="<?php echo $row['passport']?>">**

**<input type="hidden" name="userid" value="<?php echo $row['userid']?>">**

**<label>Full Name\***

**<input class="form-control" name="name" type="text" value="<?php echo $row['lastname']." ".$row['firstname']." ". $row['othername']?>" >**

**</label>**

**<label>Email\***

**<input class="form-control" name="email" type="text" value="<?php echo $row['email']?>" > </label>**

**<textarea class="form-control" cols="15" rows='4' placeholder="Enter comments, equiry or complains " name="message"></textarea> <br>**

**<input class="btn btn-primary btn-lg btn-block" value="submit" type="submit" name="submit">**

**</form>**

**</fieldset>**

**<div class="container">**

**<div class="row row-centered pos">**

**<div class="col-lg-8 col-xs-12 col-centered">**

**<h2>Messages</h2>**

**<div class="well">**

**<?php foreach($comments as $comment){ ?>**

**<div style="background: pink">**

**<img src="../uploads/<?php echo $comment['passport']?> " style="border-radius:50%; width:40px">**

**<?php echo $comment['firstname']." ".$comment['lastname']." (".$comment['email'].")"; <span style="color:green; margin-left:50px; float:right"><?php echo $comment['time']?></span>**

**<br>**

**<span style="color:black; margin-left:50px;"><?php echo $comment['message']?></span>**

**</div>**

**<br/> <?php } ?>**

**</div>**

**</div>**

**<img src="../img/ss.png" style="float:right; width:300px">**

**</div>**

**</div> </div>**

**<script>**

**<?php**

**$sql="SELECT \* from votingdate where id=1";**

**$stmt=$pdo->prepare($sql);**

**$stmt->execute();**

**$daterows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$daterow=$daterows[0];**

**$date=$daterow['date']; ?>**

**// Set the date we're counting down to**

**var countDownDate = new Date("<?php echo $date; ?>").getTime();**

**// Update the count down every 1 second**

**var x = setInterval(function() {**

**// Get todays date and time**

**var now = new Date().getTime();**

**// Find the distance between now and the count down date**

**var distance = countDownDate - now;**

**// Time calculations for days, hours, minutes and seconds**

**var days = Math.floor(distance / (1000 \* 60 \* 60 \* 24));**

**var hours = Math.floor((distance % (1000 \* 60 \* 60 \* 24)) / (1000 \* 60 \* 60));**

**var minutes = Math.floor((distance % (1000 \* 60 \* 60)) / (1000 \* 60));**

**var seconds = Math.floor((distance % (1000 \* 60)) / 1000);**

**// Display the result in the element with id="demo" document.getElementById("demo").innerHTML = "Remaining Time: " +days + "days " + hours + "h "**

**+ minutes + "m " + seconds + "s ";**

**// If the count down is finished, write some text**

**if (distance < 0) {**

**clearInterval(x);**

**document.getElementById("demo").innerHTML = "VOTING TIME EXPIRED or NOT YET SET"; $('#click').click(function(e){ e.preventDefault();});**

**function showalert(){**

**alert('Ooops...!');**

**}**

**}**

**}, 1000);**

**</script>**

**</div>**

**<br><br>**

**<br><br>**

**</div> <!-- container -->**

**</div> <!-- content -->**

**</div>**

**<footer class="footer text-right">**

**2019 © Voting System by Olamide(Horlitech).**

**</footer>**

**<!-- Right Sidebar -->**

**<div class="side-bar right-bar">**

**<a href="javascript:void(0);" class="right-bar-toggle">**

**<i class="mdi mdi-close-circle-outline"></i>**

**</a>**

**<h4 class="">Settings</h4>**

**<div class="setting-list nicescroll">**

**<div class="row m-t-20">**

**<div class="col-xs-8">**

**<h5 class="m-0">Notifications</h5>**

**<p class="text-muted m-b-0"><small>Do you need them?</small></p>**

**</div>**

**<div class="col-xs-4 text-right">**

**<input type="checkbox" checked data-plugin="switchery" data-color="#7fc1fc" data-size="small"/>**

**</div>**

**</div>**

**<div class="row m-t-20">**

**<div class="col-xs-8">**

**<h5 class="m-0">API Access</h5>**

**<p class="m-b-0 text-muted"><small>Enable/Disable access</small></p>**

**</div>**

**<div class="col-xs-4 text-right">**

**<input type="checkbox" checked data-plugin="switchery" data-color="#7fc1fc" data-size="small"/>**

**</div>**

**</div>**

**<div class="row m-t-20">**

**<div class="col-xs-8">**

**<h5 class="m-0">Auto Updates</h5>**

**<p class="m-b-0 text-muted"><small>Keep up to date</small></p>**

**</div>**

**<div class="col-xs-4 text-right">**

**<input type="checkbox" checked data-plugin="switchery" data-color="#7fc1fc" data-size="small"/>**

**</div>**

**</div>**

**<div class="row m-t-20">**

**<div class="col-xs-8">**

**<h5 class="m-0">Online Status</h5>**

**<p class="m-b-0 text-muted"><small>Show your status to all</small></p>**

**</div>**

**<div class="col-xs-4 text-right">**

**<input type="checkbox" checked data-plugin="switchery" data-color="#7fc1fc" data-size="small"/>**

**</div>**

**</div>**

**</div>**

**</div>**

**<!-- /Right-bar -->**

**</div>**

**</body>**

**</html>**

**Appendix II**

**(Source code for Authentication)**

**<!DOCTYPE html>**

**<html lang="en">**

**<?php**

**error\_reporting(0);**

**require('conn/connection.php');**

**session\_start();**

**if(!isset($\_SESSION['id'])){**

**//not loggin**

**header('Location:../login.php');**

**exit();**

**}**

**function createRandomPassword() {**

**$chars = "003232303232023232023456789";**

**srand((double)microtime()\*1000000);**

**$i = 0;**

**$pass = '' ;**

**while ($i <= 7) {**

**$num = rand() % 33;**

**$tmp = substr($chars, $num, 1);**

**$pass = $pass . $tmp;**

**$i++; }**

**return $pass;**

**}**

**$message=createRandomPassword();**

**$userid=$\_SESSION['id'];**

**$psql="SELECT \* from profile join users join department join faculty ON profile.userid=users.id and profile.departmentid=department.id and profile.facultyid=faculty.id where profile.userid=:userid";**

**$stmt=$pdo->prepare($psql);**

**$stmt->bindParam(':userid', $userid);**

**$stmt->execute();**

**$prows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$prow=$prows[0];**

**$sql="SELECT \* FROM authentication where userid=:userid";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':userid', $userid);**

**$stmt->execute();**

**$rows=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**$row=$rows[0];**

**if($row['userid']==$userid){**

**$sql="UPDATE authentication SET pin=:pin where userid=:userid";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':pin', $message);**

**$stmt->bindParam(':userid', $userid);**

**$stmt->execute();**

**}**

**else{**

**$sql="INSERT INTO authentication(userid, pin)values(:userid, :pin)";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':pin', $message);**

**$stmt->bindParam(':userid', $userid);**

**$stmt->execute(); }**

**$owneremail="olamide.oyefabi@fubk.edu.ng";**

**$subacct="OLA";**

**$subacctpwd="horlitech1212";**

**$sendto=$prow['phone'];**

**$mail=$prow['email'];**

**/\* destination number \*/**

**$sender="SRC"; /\* sender id \*/**

**/\* message to be sent \*/**

**/\* create the required URL \*/**

**$url = "http://www.smslive247.com/http/index.aspx?"**

**. "cmd=sendquickmsg"**

**. "&owneremail=" .$owneremail**

**. "&subacct=" .$subacct**

**. "&subacctpwd=" .$subacctpwd**

**. "&message=".$message**

**. "&sender=".$sender**

**. "&sendto=".$sendto**

**. "&msgtype=". 0;**

**/\* call the URL \*/**

**if ($f = @fopen($url, "r"))**

**{**

**$answer = fgets($f, 255);**

**if (substr($answer, 0, 1) == "+")**

**{**

**echo "SMS to $dnr was successful.";**

**}**

**else**

**{**

**echo "<script>alert('OTP Sent')</script>";**

**}**

**}**

**else**

**{**

**echo "<script>alert('Bad network please try again.')</script>"; }?>**

**<head>**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1.0">**

**<meta name="description" content="Creative - Bootstrap 3 Responsive Admin Template">**

**<meta name="author" content="GeeksLabs">**

**<meta name="keyword" content="Creative, Dashboard, Admin, Template, Theme, Bootstrap, Responsive, Retina, Minimal">**

**<link rel="shortcut icon" href="img/favicon.png">**

**<title>Authentication</title>**

**<!-- Bootstrap CSS -->**

**<link href="css/bootstrap.min.css" rel="stylesheet">**

**<!-- bootstrap theme -->**

**<link href="css/bootstrap-theme.css" rel="stylesheet">**

**<!--external css-->**

**<!-- font icon -->**

**<link href="css/elegant-icons-style.css" rel="stylesheet" />**

**<link href="css/font-awesome.css" rel="stylesheet" />**

**<!-- Custom styles -->**

**<link href="css/style.css" rel="stylesheet">**

**<link href="css/style-responsive.css" rel="stylesheet" />**

**<script type="text/javascript">**

**function noBack()**

**{**

**window.history.forward()**

**}**

**noBack();**

**window.onload = noBack;**

**window.onpageshow = function(evt) { if (evt.persisted) noBack() }**

**window.onunload = function() { void (0) }**

**</script>**

**</head>**

**<body class="login-img3-body" style="background-image: linear-gradient(rgba(10,0,80,0.3), rgba(10,0,80,0.3)), url('img/cover.png') ; background-size:cover ">**

**<div class="container">**

**<form class="login-form" method="POST" action="verificationexe.php">**

**<div class="login-wrap">**

**<p class="login-img"><i class="icon\_lock\_alt"></i> <br><span style="font-size:0.7em">VERIFICATION</span></p>**

**<div class="input-group">**

**<span class="input-group-addon"><i class="icon\_key"></i></span>**

**<input type="text" class="form-control" placeholder="Verification PIN" name="OTP" autofocus required>**

**</div>**

**<input class="btn btn-primary btn-lg btn-block" type="submit" name="submit" value="Authenticate"/>**

**<a class="btn btn-primary btn-lg btn-block" style="background:transparent" href="verification.php">RESEND OTP</a>**

**</div>**

**</form>**

**<div class="text-right">**

**</div>**

**</div>34**

**</body>**

**</html>**

**Appendix III**

**(Source code for voting page)**

**<form id="msform" name="joe" action="preview.php" method="POST" onsubmit="return sapatka()" >**

**<!-- progressbar -->**

**<fieldset >**

**<?php**

**$position="President";**

**$sql="SELECT \* from candidates JOIN profile JOIN postions ON candidates.userid=profile.userid and candidates.postionid=postions.id where position =:position";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':position', $position);**

**$stmt->execute();**

**$presidents=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**foreach($presidents as $president){**

**$president\_list .= "<option value='".$president['userid']. "'>".$president['lastname'].' '.$president['othername'].' '.$president['firstname']."</option>";**

**} ?>**

**<center><img src="../../images/logo.png" style="width:13%; "/></center><img src="../../images/barcode.png" style="float:right; width:5%">**

**<h1 class="fs-title" style="text-align:center; font-weight:bold; font-size:150%">Federal University Birnin Kebbi<br><span style="color:red; "> SRC ELECTION BALLOT PAPER</span></h1>**

**==========================================================================================**

**<h2 class="fs-title" style=" font-weight:bold" ><img src="../../images/id.png" style="width:10%"> PRESIDENT</h2>**

**<h3 class="fs-subtitle">Pls. Select Only One</h3>**

**<select name="President" style="height: 51px;">**

**<option value="0">I Have no Candidate</option>**

**<?php echo$president\_list; ?>**

**</select>**

**<input type="button" value="Next" class="next action-button" name="next">**

**</fieldset>**

**<fieldset>**

**<?php**

**$position="Vice President";**

**$sql="SELECT \* from candidates JOIN profile JOIN postions ON candidates.userid=profile.userid and candidates.postionid=postions.id where position =:position";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':position', $position);**

**$stmt->execute();**

**$Vpresidents=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**foreach($Vpresidents as $Vpresident){**

**$Vpresident\_list .= "<option value='".$Vpresident['userid']. "'>".$Vpresident['firstname'].' '.$Vpresident['othername'].' '.$Vpresident['lastname']."</option>";**

**} ?>**

**<center><img src="../../images/logo.png" style="width:13%; "/></center><img src="../../images/barcode.png" style="float:right; width:5%">**

**<h1 class="fs-title" style="text-align:center; font-weight:bold; font-size:150%">Federal University Birnin Kebbi<br><span style="color:red; "> SRC ELECTION BALLOT PAPER</span></h1>**

**==========================================================================================**

**<h2 class="fs-title" style=" font-weight:bold" ><img src="../../images/id.png" style="width:10%">VICE PRESIDENT</h2>**

**<h3 class="fs-subtitle">Pls. Select Only One</h3>**

**<select name="VPresident" style="height: 51px;">**

**<option value="0">I Have no Candidate</option>**

**<?php echo $Vpresident\_list; ?>**

**</select>**

**<input type="button" value="Previous" class="previous action-button" name="previous">**

**<input type="button" value="Next" class="next action-button" name="next">**

**</fieldset>**

**<fieldset>**

**<?php**

**$position="General Secretary";**

**$sql="SELECT \* from candidates JOIN profile JOIN postions ON candidates.userid=profile.userid and candidates.postionid=postions.id where position =:position";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':position', $position);**

**$stmt->execute();**

**$secretarys=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**foreach($secretarys as $secretary){**

**$secretary\_list .= "<option value='".$secretary['userid']. "'>".$secretary['firstname'].' '.$secretary['othername'].' '.$secretary['lastname']."</option>";**

**} ?>**

**<center><img src="../../images/logo.png" style="width:13%; "/></center><img src="../../images/barcode.png" style="float:right; width:5%">**

**<h1 class="fs-title" style="text-align:center; font-weight:bold; font-size:150%">Federal University Birnin Kebbi<br><span style="color:red; "> SRC ELECTION BALLOT PAPER</span></h1>**

**==========================================================================================**

**<h2 class="fs-title" style=" font-weight:bold" ><img src="../../images/id.png" style="width:10%">SECRETARY</h2>**

**<h3 class="fs-subtitle">Pls. Select Only One</h3>**

**<select name="Secretary" style="height: 51px;">**

**<option value="0">I Have no Candidate</option>**

**<?php echo $secretary\_list; ?>**

**</select>**

**<input type="button" value="Previous" class="previous action-button" name="previous">**

**<input type="button" value="Next" class="next action-button" name="next">**

**</fieldset>**

**<fieldset>**

**<?php**

**$position="Treasurer";**

**$sql="SELECT \* from candidates JOIN profile JOIN postions ON candidates.userid=profile.userid and candidates.postionid=postions.id where position =:position";**

**$stmt=$pdo->prepare($sql);**

**$stmt->bindParam(':position', $position);**

**$stmt->execute();**

**$treasurers=$stmt->fetchAll(PDO::FETCH\_ASSOC);**

**foreach($treasurers as $treasurer){**

**$treasurer\_list .= "<option value='".$treasurer['userid']. "'>".$treasurer['firstname'].' '.$treasurer['othername'].' '.$treasurer['lastname']."</option>";**

**} ?>**

**<center><img src="../../images/logo.png" style="width:13%; "/></center><img src="../../images/barcode.png" style="float:right; width:5%">**

**<h1 class="fs-title" style="text-align:center; font-weight:bold; font-size:150%">Federal University Birnin Kebbi<br><span style="color:red; "> SRC ELECTION BALLOT PAPER</span></h1>**

**==========================================================================================**

**<h2 class="fs-title" style=" font-weight:bold" ><img src="../../images/id.png" style="width:10%">TREASURER</h2>**

**<h3 class="fs-subtitle">Pls. Select Only One</h3>**

**<select name="Treasurer" style="height: 51px;">**

**<option value="0">I Have no Candidate</option>**

**<?php echo $treasurer\_list; ?>**

**</select>**

**<input type="button" value="Previous" class="previous action-button" name="previous">**

**<input type="button" value="Next" class="next action-button" name="next">**

**</fieldset>**