APPLIED INFORMATION TECHNOLOGY AND COMPUTER SCIENCE
e-ISSN: 2773-5141

AITCS

Vol. 5 No. 2 (2024) 174-192
https://publisher.uthm.edu.my/periodicals/index.php/aitcs

# A Secure E-Voting Web-Based System utilizing Bcrypt hashing, Email One-Time Password and Email Verification for Politeknik METrO Tasek Gelugor

**Nethaneal Robert[1], Nor Bakiah Abd Warif[1*]**

[1] *Department of Information Security and Web Technology*
*Faculty of Computer Science and Information Technology*
*University Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, Malaysia*

*Corresponding Author: norbakiah@uthm.edu.my
DOI: https://doi.org/10.30880/aitcs.2024.05.02.010

**Article Info**

Received: 9 October 2024
Accepted: 16 October 2024
Available online: 15 December 2024

**Keywords**

E-Voting, Bcrypt hash, email One-Time Password, password authentication, email verification

**Abstract**

This project aims to develop a Secure E-Voting Web-Based System for Politeknik METrO Tasek Gelugor using the Bcrypt hash function, email one-time password for password recovery, and email verification. The current manual voting method lacks secrecy, authentication, and integrity, compromising effectiveness. The new system seeks to improve security and accessibility, ensuring voter confidentiality and process integrity. Utilizing the Agile methodology for iterative development and feedback, the system includes email verification, password authentication, Bcrypt hashing, and email-based one-time passwords. The expected outcome is a secure, effective system promoting democratic participation. The project underscores the potential for transforming electronic voting systems, enhancing voter engagement, and upholding democratic values at Politeknik METrO Tasek Gelugor.

## 1. Introduction

The rapid evolution of technology has transformed various aspects of our lives, including how elections and voting occur. To overcome the limitations of the current manual voting system at Politeknik METrO Tasek Gelugor and improve both security and accessibility, a Secure E-voting Web-Based System is being developed. This project focuses on designing, implementing, and testing the web application for online voting at the institution. The current system suffers from several drawbacks, including lack of confidentiality, absence of authentication or verification, and integrity issues in the vote counting process.

The primary goals are to design A Secure E-Voting Web-Based System utilizing Bcrypt Hashing, email-based one-time password and email verification for Politeknik Metro Tasek Gelugor., develop it using Visual Studio Code, and test its functionality. The project scope includes students, lecturers, and administrative staff involved in the voting process at Politeknik METrO Tasek Gelugor. Key issues like lack of confidentiality, absence of authentication or verification, and integrity concerns in vote counting will be addressed through password authentication, email verification, and Bcrypt hashing.

The expected outcome is a successfully implemented Secure E-Voting Web-Based System, ensuring enhanced security, confidentiality, and integrity of the voting process. This project is significant for modernizing the voting process, ensuring election integrity, and encouraging democratic participation among students. By implementing a web-based system, the project ensures wider accessibility and improved security, offering a robust solution to the current challenges.

## 2. Literature Review

This section explains about literature review that will be conducted as part of this project which discusses Time-based One-Time password authentication, password, email verification, and Bcrypt Hashing.

### 2.1 Bycrypt Hash

Bcrypt is a cryptographic hash function specifically designed for hashing passwords [1]. It enhances security by making the hashing process slower and more resistant to brute-force attacks through repeated iterations and the use of salt to protect against rainbow table attacks. Bcrypt uses 128-bit salt and produces a 192-bit hash. The hashing process involves generating salt, expanding the key using the Blowfish algorithm, mixing the internal state through multiple rounds, and producing the final hashed output, which includes both the salt and the hash. This method helps safeguard passwords against brute force and dictionary attacks.

### 2.2 Email One-Time Password

Email One-Time Password (Email OTP) is a two-factor authentication method that enhances account security. Instead of fixed passwords, Email OTP generates unique, time-sensitive codes that users enter along with their regular credentials during login [2]. These codes are created using the current time and a secret key and are typically valid for 30 or 60 seconds. Users receive these codes through authentication apps on their phones, which are synchronized with the server's clock. Email OTP reduces the risk of phishing and password interception, effectively protecting confidential information and enhancing overall security [3].

### 2.3 Email Verification

Email One-Time Password (Email OTP) is a two-factor authentication method that enhances account security. Instead of fixed passwords, Email OTP generates unique, time-sensitive codes that users enter along with their regular credentials during login [2]. These codes are created using the current time and a secret key and are typically valid for 30 or 60 seconds. Users receive these codes through authentication apps on their phones, which are synchronized with the server's clock. Email OTP reduces the risk of phishing and password interception, effectively protecting confidential information and enhancing overall security [3].

### 2.4 Password Authentication

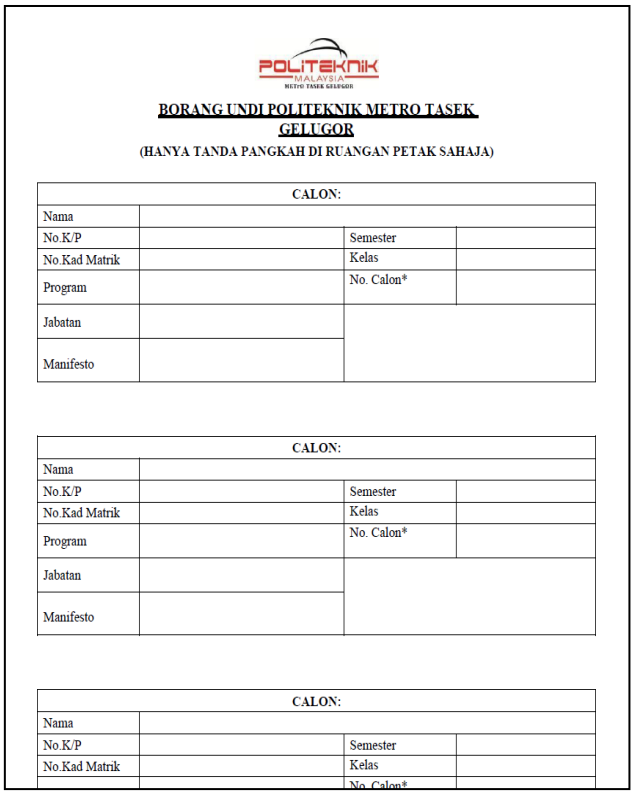
Password authentication remains the prevailing method of access control for both the Web and mobile devices, and its practicality and widespread use are unlikely to be superseded by alternative authentication methods in the foreseeable future [4]. A unique password that the user creates during registration is safely saved in the system using methods like hashing or encryption [5]. Users input their credentials during the login process, and the system returns the password hash that is saved for their account.

### 2.5 Study of Related System

This section provides a comprehensive analysis of current systems that are relevant to the project. This examination includes a detailed evaluation of their functionalities, strengths, and weaknesses, as well as a comparison with the proposed system. By understanding the existing solutions, this study aims to identify gaps, derive best practices, and establish a foundation for enhancing the new system's design and implementation.

#### 2.5.1 Existing System (Politeknik METrO Tasek Gelugor)

In the current manual voting system at Politeknik METrO Tasek Gelugor, students complete paper ballots expressing their candidate preferences. These ballots are collected and transported to the Higher Education Department (HEP) office, where votes are tallied and counted following established protocols. This traditional process presents challenges, particularly concerning voter privacy and reliable authentication. Ensuring secure handling practices during the transit of paper ballots from the polling place to the HEP office is crucial for maintaining electoral integrity. The candidate selection process is manual and crucial. This decision is based on evaluating candidates, including academic performance, grade point averages, and participation in Majlis Perwakilan Pelajar (MPP) activities. Despite being a tradition, the manual approach acknowledges challenges, prompting a shift to a more streamlined and secure e-voting system. The manual process occurs during PAK Sessions, guided by lecturers, emphasizing structured and supervised student participation in the democratic process. Figure 1 shows the template of the paper ballot that the students use to vote for the candidate.



POLITEKNIK MALAYSIA
METRO TASEK GELUGOR

**BORANG UNDI POLITEKNIK METRO TASEK GELUGOR**

**(HANYA TANDA PANGKAH DI RUANGAN PETAK SAHAJA)**

| CALON: | | | |
|---|---|---|---|
| Nama | | | |
| No.K/P | | Semester | |
| No.Kad Matrik | | Kelas | |
| Program | | No. Calon* | |
| Jabatan | | | |
| Manifesto | | | |

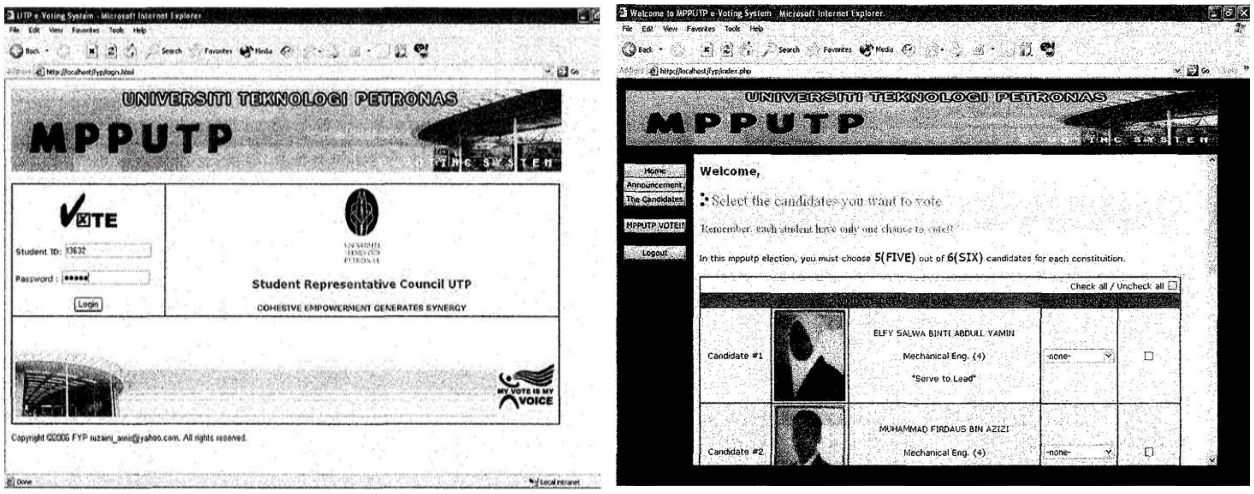
| CALON: | | | |
|---|---|---|---|
| Nama | | | |
| No.K/P | | Semester | |
| No.Kad Matrik | | Kelas | |
| Program | | No. Calon* | |
| Jabatan | | | |
| Manifesto | | | |

| CALON: | | | |
|---|---|---|---|
| Nama | | | |
| No.K/P | | Semester | |
| No.Kad Matrik | | Kelas | |
| | | No. Calon* | |

**Fig. 1** *New poll page in Pollie App*

## 2.5.2 Study of Electronic voting system (E-voting) for MPPUTP Election

The Electronic Voting System for MPPUTP elections, developed by Ruzaini Bt. Amir, is a web-based platform designed to replace traditional paper ballots with an internet-based voting method for increased accessibility and efficiency [6]. E-voting offers advantages like cost reduction, fewer errors, and improved accessibility for voters. The system addresses the inconvenience of the previous manual process, allowing students to vote online without the need to travel in person on election day. Comparing this system with the proposed Secure E-Voting Web-Based System for Politeknik METrO Tasek Gelugor, it becomes evident that both systems aim to enhance security and accessibility in the voting process. The MPP UTP system incorporates digital signatures to ensure the authenticity and verification of the entire voting process, overcoming authentication and verification challenges [6]. Figure 2 shows the interface in Electronic Voting System for MPPUTP.




**(a)** **(b)**

**Fig.2** *Electronic Voting System for MPPUTP. (a) login page; (b) e-voting ballot [6]*

### 2.5.3 Study of Pollie App

Pollie is an easy-to-use app for creating and sharing polls. You can customize polls with graphics, voting deadlines, and vote limits. No user accounts are needed, making it accessible from anywhere. The app offers strong analytics, including statistics and graphs. It ensures reliable voting with features like password security and email verification. Users can personalize polls, thank voters, and redirect user to a website after voting. Figure 3 shows the Pollie app page.

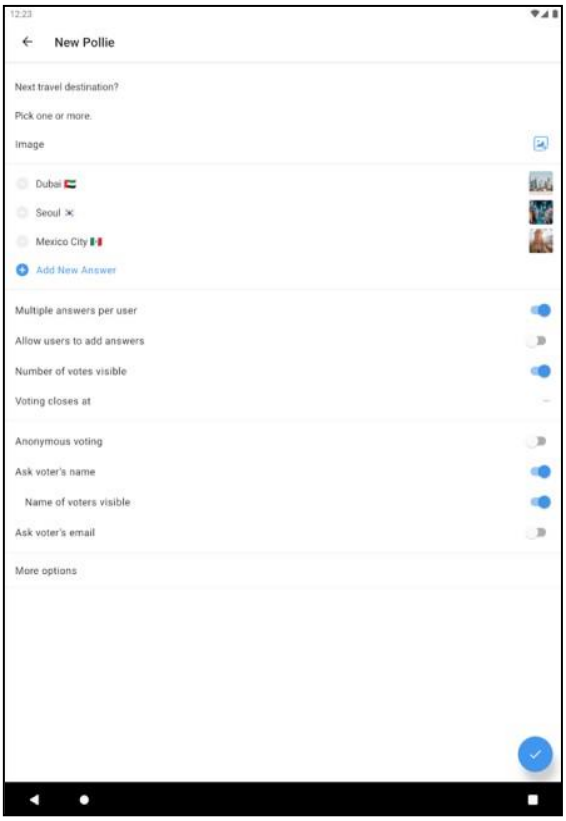



**Fig. 3** *New poll page in Pollie App*.

### 2.5.4 Comparison Table with proposed system

Table 1 shows the comparison between all three existing system and proposed system. The comparison table assesses various voting systems, highlighting their security protocols and authentication methods. While the manual system lacks reliable authentication, the MPPUTP utilizes digital signatures for verification, addressing this concern. However, the Pollie App falls short in providing any verification methods. In contrast, the proposed E-Voting System boasts robust authentication measures such as password authentication, OTP via email, and hashing for enhanced security. It ensures complete confidentiality and data anonymization by hashing the voter ID stored in the database, thereby preserving anonymity. This system represents a significant upgrade over existing methods, at Politeknik METrO Tasek Gelugor.

**Table 1** *Comparison table*

| Criteria | Existing manual system | Electronic Voting System (MPPUTP) | Pollie App | Propose E-Voting System |
|---|---|---|---|---|
| Platform | Paper ballot | Web-Based | Mobile Application | Web-based |
| Authentication/Verification module | No reliable authentication/ verification | Digital Signature | No | Yes (password authentication)(Email verification) |
| Data anonymization | No | Not stated | Optional | Yes (hashing) |

**Table 1** *Comparison table (cont.)*

| Criteria | Existing manual system | Electronic Voting System (MPPUTP) | Pollie App | Propose E-Voting System |
|---|---|---|---|---|
| Result Reporting | Manual | No | Yes (-simple pollform) | Yes (PDF) |
| Confidentiality invoting process | No | Not stated | No | Yes (hash voter id) |
| Accessibility | Requires physical presence | Web-based system accessible remotely | App based requires user to download the app. | Web-based system accessible remotely |
| Security measures | Limited security measures | Digital Signature | Password | Password, email OTP, email verification, Hashing |
| Overall System Enhancement | Limited modernization | Modern Web-basedsystem | Basic polling app | Significant modernizationwith hashing and real- time tracking |

## 3. Methodology

Agile software development refers to software development that is quick, active, and responsive which prioritizes adaptability, collaboration, and customer satisfaction [7]. Unlike traditional development models that follow a linear, sequential process, Agile divides the project into small increments called iterations. Figure 4 shows the phases within Agile methodology.


**Fig. 4** *Phases within Agile Methodology*

### 3.1 Planning Phase

The Agile methodology's initial step involves a comprehensive review of project scope, objectives, and approach. Key stakeholders, such as students, academic advisors, and HEP staff, are identified to establish a clear vision. A high-level project plan is created to outline major milestones.

## 3.2 Requirement Analysis Phase

The requirement analysis phase involves gathering, analyzing, and organizing the project's needs into a prioritized backlog. For the Secure E-Voting system, this includes secure authentication, vote confidentiality, and an efficient user interface

## 3.3 Design Phase

During the design phase, a detailed strategy is developed for implementing the prioritized requirements. This includes designing the application, user interface, database, and overall system architecture, with a focus on security features like Bcrypt hashing and email-based OTPs

## 3.4 Development/Implementation Phase

The development/implementation phase involves converting design ideals into a concrete and user-friendly Secure E-Voting system. Coding and architectural development processes are initiated, aiming to complete the system's basic functionality by the end of the first iteration or sprint. Hardware, software, and programming language requirements are determined for system development. Bcrypt hashing is used to securely encrypt voter IDs, making them difficult to decipher or tamper with. Email-based one-time passwords (OTPs) are implemented for enhanced authentication, requiring users to verify their identity through their email. Additionally, email verification during registration ensures the validity of user accounts. Secure login mechanisms, encrypted communication using SSL. These features collectively strengthen the system's security, ensuring confidential andsecure voting processes.

## 3.5 Testing Phase

Functional and user acceptance testing ensures each feature functions as specified and meets end-user expectations. Trial runs are conducted to ensure system functionality and test security features such as email OTP, email verification, and hashing

## 3.6 Deployment Phase

The Deployment involves the successful distribution of the Secure E-Voting system to end-users. Release preparation, user training, and software deployment into the production environment ensure a smooth deployment process. Maintenance post-deployment helps in defect elimination and functionality preservation, allowing for feedback collection and future improvements

# 4. System Analysis and Design

This section focuses on the specific functionality that users may anticipate from the Secure E-Voting System. It delineates the core features and capabilities that users can interact with, ensuring a comprehensive understanding of the system's behavior. Table 2 and Table 3 shows the functional requirement and non-functional requirement respectively with its description.

**Table 2** *Functional requirement*

| Requirement | Description |
|---|---|
| User Authenticationand verification | Users should be able to verify their email and log in securely using voter id and password. |
| Voting process | Display a list of candidates for voting.<br>Allow students to select and confirm their vote. |
| Security measures | Implement email verification and password authentication for an added layer of security during the voting process. Use hashing techniques to secure the voters id after voters casted their vote.<br>Implement email OTP authentication for forget password. |
| Result Display | Ensure That individual votes remain confidential. |
| Administrative Controls | Allow administrators to manage and monitor votingprocess.<br>Provide report generating control to audit voting results. |

**Table 3** *Non-functional requirement*

| Requirement | Description |
|---|---|
| Performance | The system should handle a specified number of simultaneous users without significant performance degradation. |
| Usability | The user interface should be intuitive and user-friendly. Users should be able to complete the voting processefficiently. |
| Security | Ensure that the system complies with industry standardsfor secure data transmission. Regularly update and patch security vulnerabilities. |
| Reliability | The system should be available and responsive during theentire voting duration. |
| Compatibility | The system should be compatible with a range of devices. |
| Regulatory compliance | Ensure that the system complies with relevant data protection and privacy regulations. Provide comprehensivedocumentation for administrators. |

## 4.1 Design Phase

The Secure E-Voting Web-based System streamlines the voting process for both students and administrators. After being registered by the admin, students must verify their identity via an email link each time before logging in. After the verification and login process, students log in to the system to cast their votes. view candidate details and cast their votes. They also have the option to view the vote count for each candidate.

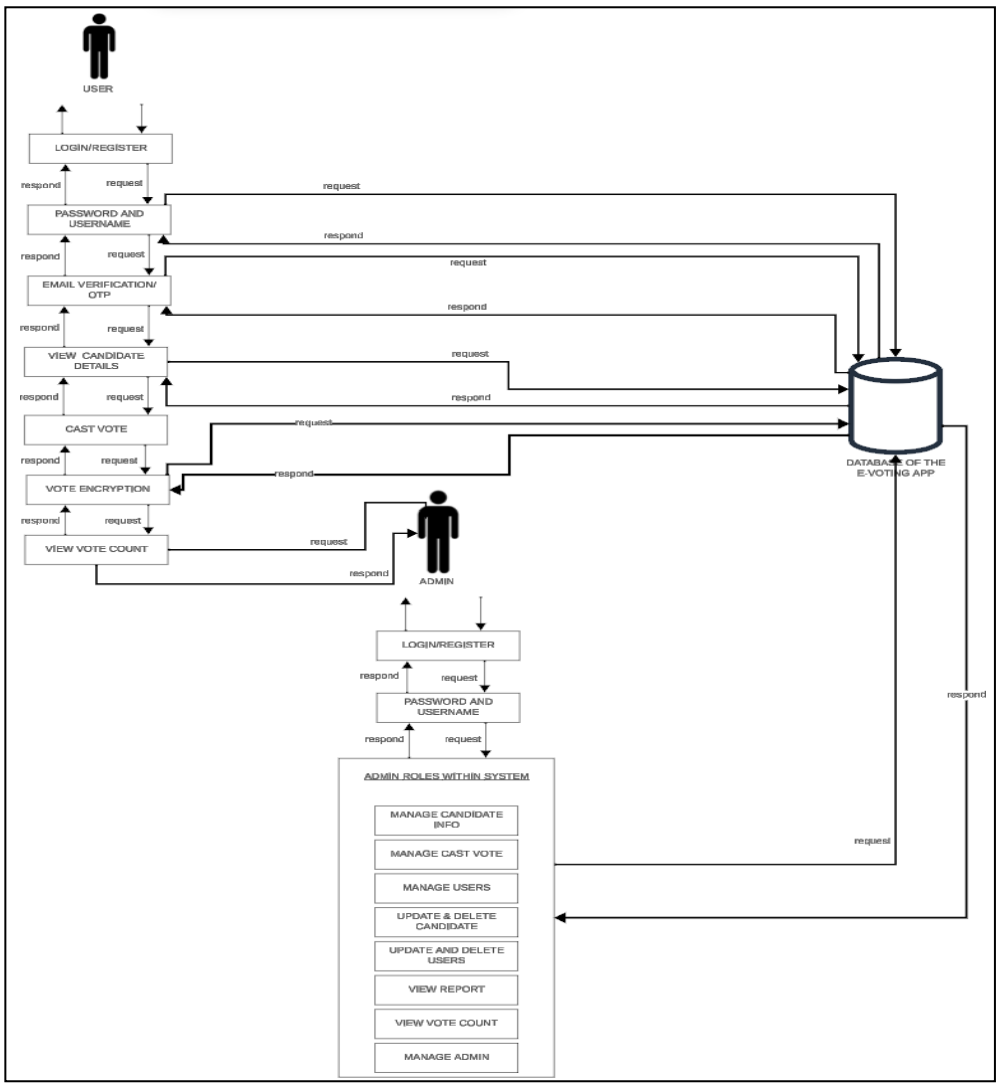



**Fig. 5** *System Architecture*

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 174-192

For administrators, the system offers tools to manage candidates, voters, admins, and positions. Administrators are equipped with tools for managing candidates, voters (students), and positions, among other administrative tasks. They can generate reports within the system to gain insights into the voting process and access vote count information for monitoring and analysis. This comprehensive functionality ensures a smooth and secure experience for both voters and administrators throughout the e-voting process.

## 4.2 Use-Case Diagram

A use case diagram serves as a visual representation of the system's roles and interactions. A use case diagram visually represents the roles and interactions within the system, providing a clear blueprint for system development. Stakeholders can use this diagram to ensure alignment with the system's functional requirements and to facilitate effective communication throughout the development process.

Admins initiate student registration, which involves email verification for identity validation. Once verified, users log in using their voter ID and password, with the system authenticating credentials before granting access to voting functionalities. Users can then select their preferred candidates, and the system securely records their votes, allowing users to view the current tally through the "View Vote Count" feature.

For administrators, logging in with authorized credentials grants access to a range of administrative features. Admins can manage users, candidates, and voting processes, review cast votes for issues or disputes, and generate reports summarizing election results. This comprehensive depiction elucidates the system's functionality and facilitates stakeholders' understanding and agreement regarding the expected behavior. It serves as a crucial blueprint for system development, ensuring that all parties involved have a clear understanding of the system's roles and interactions, thus contributing to a smooth and efficient implementation process. The use case diagram in Figure 6 outlines the roles and interactions of users and administrators within the system.

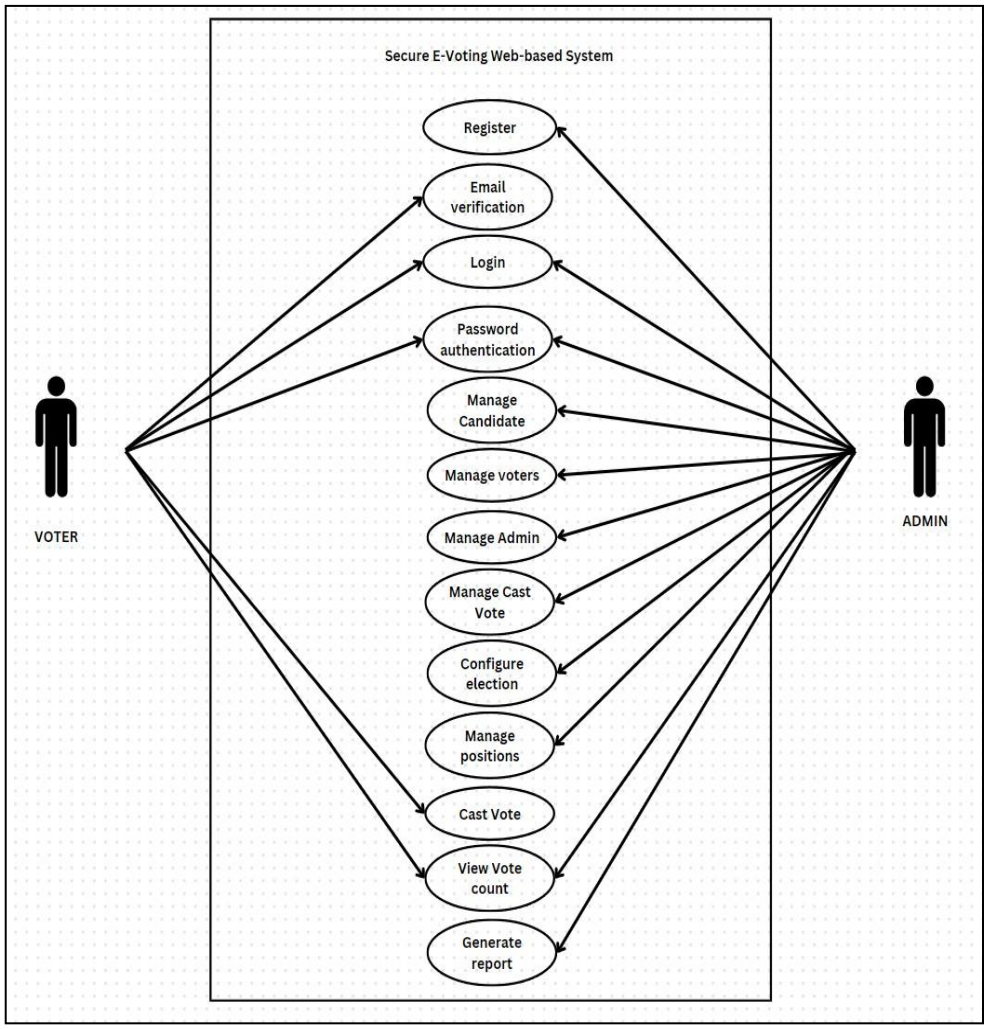



**Fig. 6** *System Architecture of the Secure E-Voting Web-based System*.

## 4.3 Sequence Diagram

Figure 7 and 8 depicts of the flow of activities for voter and admin of the proposed Secure E-Voting Web- based System. The student roles within the system involve several key processes. The User Verification & Login Sequence details the steps for user verification and authentication, granting access to the voting functions. The Cast Vote Sequence explains how students select candidates, ensuring secure and anonymous voting. Lastly, the View Vote Count Sequence shows how students can view the current vote count, ensuring transparency in the voting process.

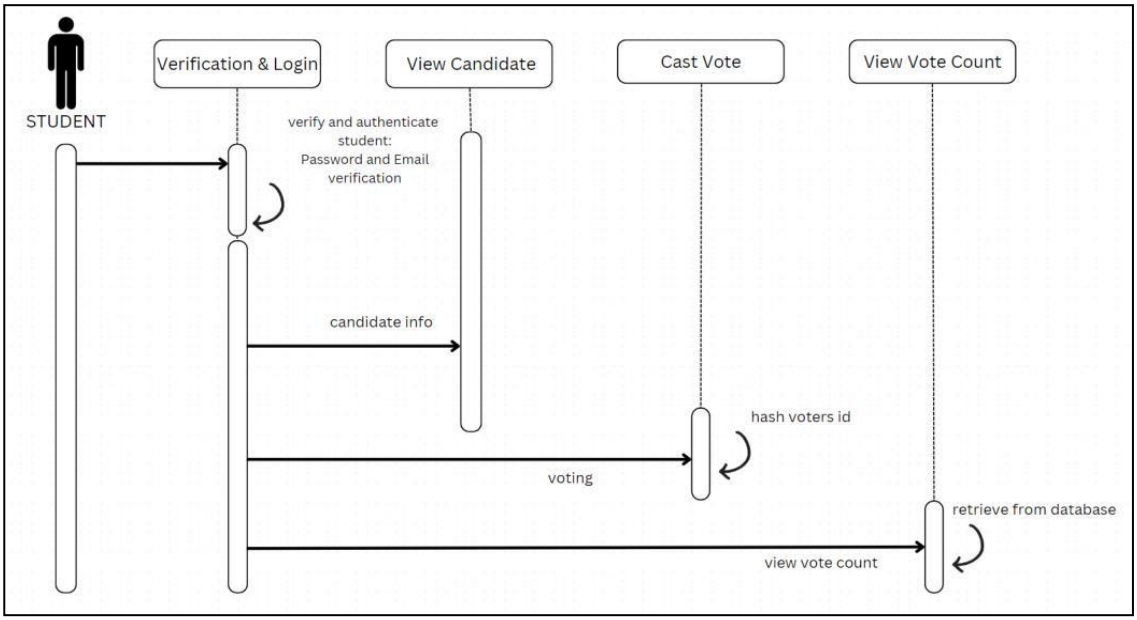



**Fig. 7** *Sequence Diagram for voter*

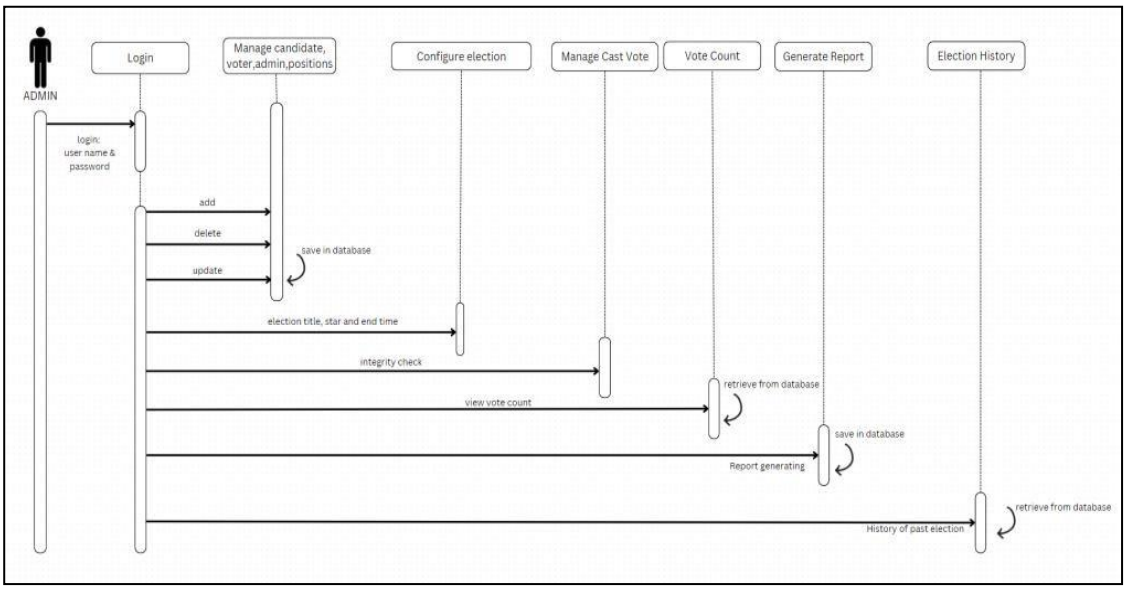



**Fig. 8** *Sequence Diagram for admin*

The procedure for admin starts with the administrator providing credentials for authentication. After successful login, the admin accesses administrative functions. The "Manage candidate, voter, admin, position" sequence outlines how the admin can add, edit, or remove entries. The system updates the database accordingly. The "Configure election" sequence allows the admin to set the election title and its start and end times.

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 174-192

## 4.4 Activity Diagram

An activity diagram is particularly useful for illustrating the dynamic aspects of a system by showing the order in which activities are executed. It provides a high-level view of how different tasks or actions interact and transition from one to another, making it a valuable tool for understanding and communicating complex processes in a graphical and easily understandable format.

The activity diagram in Figure 9 depicts the entire workflow of the system, starting with admin registration of voters and administrators, followed by login procedures. Admins access the dashboard for various management tasks. For voters, email verification and password authentication are necessary for login. After successful login, voters view and vote for candidates, then view vote totals. Both voting and administrative sessions end with a logout. The diagram provides a clear overview of system processes, facilitating understanding and communication of complex workflows in a graphical format.

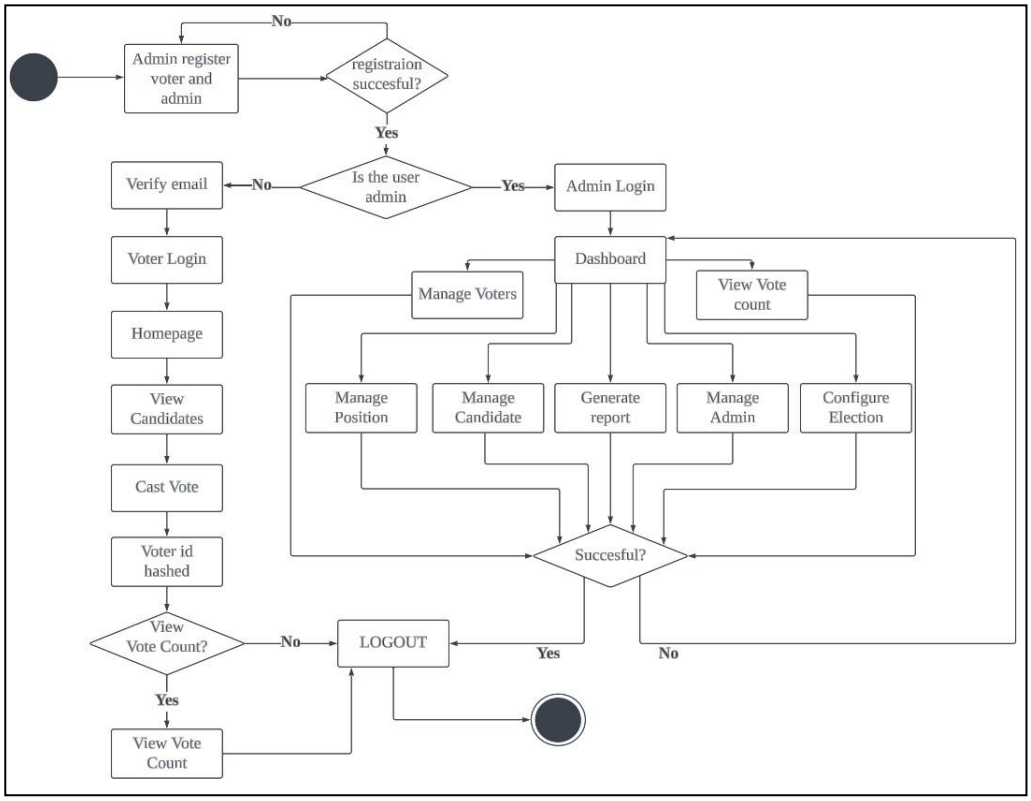



**Fig. 9** *Activity diagram for the Secure E-Voting Web-based System*.

## 4.5 Entity Relation Diagram (ERD)

The Entity-Relationship Diagram (ERD) is a visual representation used in database design to show the links between different entities in a system. Admins manage system records, voters, and candidates with unique IDs, usernames, passwords, and emails. Voters have unique IDs, passwords, and voter IDs, with OTP requests for password resets. Candidates have IDs, names, photos, and visions. Positions have IDs, descriptions, and maximum votes, with multiple candidates competing for each position. Votes link voters and candidates, ensuring traceability. History records system events, linked to admins, while OTP requests verify voters. This structured model ensures efficient management, integrity, and transparency in the e-voting system. Figure 10 shows the relationships between each entity within the system.

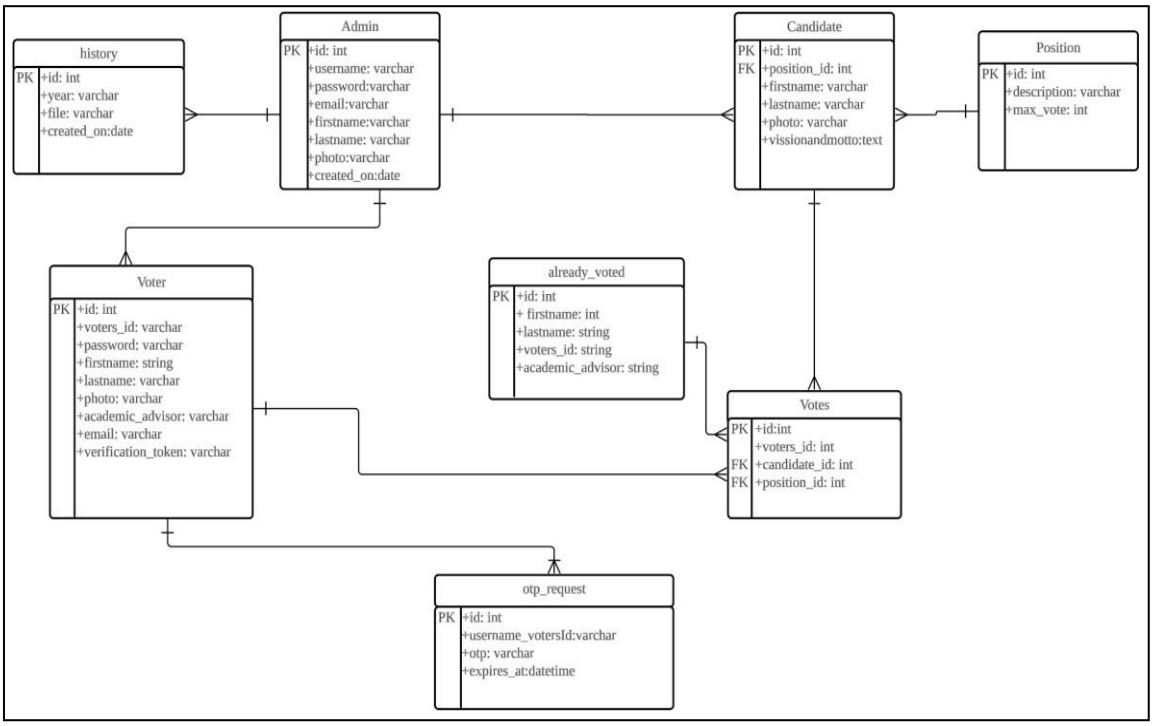



**Fig. 10** *ERD.*

## 5. Implementation and testing

This section covers the implementation and testing of the E-voting system, ensuring security and usability. Technical issues were addressed, and thorough testing was conducted to ensure functionality and reliability, preparing the system for deployment.

### 5.1 Security Implementation

This section outlines the implementation of various modules in the E-voting system. Dividing the system into modules ensures structured control over all aspects of the voting process, simplifying development, testing, and maintenance. Key modules include Login, Dashboard, Manage Positions, Register Candidate, Register Voters, Manage Votes, View Vote Count, Register Admin, Election History, and Set Election Title.

#### 5.1.1 Bcrypt hash

Bcrypt hashing enhances security by generating a unique salt for each password, making it challenging for attackers to guess passwords using precomputed hash tables. In the e-voting system, Bcrypt is not only used to secure user passwords but also to anonymize votes by hashing the "voter_id," ensuring voter privacy and preventing votes from being attributed to specific individuals. Figure 11 and 12 shows the bcrypt hash in system.

```
$candidate = $_POST[$position];
// Hash the voter's ID only for the votes table
$hashed_voters_id = password_hash($voter['voters_id'], PASSWORD_DEFAULT);
// Insert into votes table with hashed voter's ID
$sql_array[] = "INSERT INTO votes (voters_id, candidate_id, position_id) VALUES ('$hashed_voters_id', '$candidate', '$pos_id')";
```

**Fig. 11** *The Bcrypt hash implementation.*

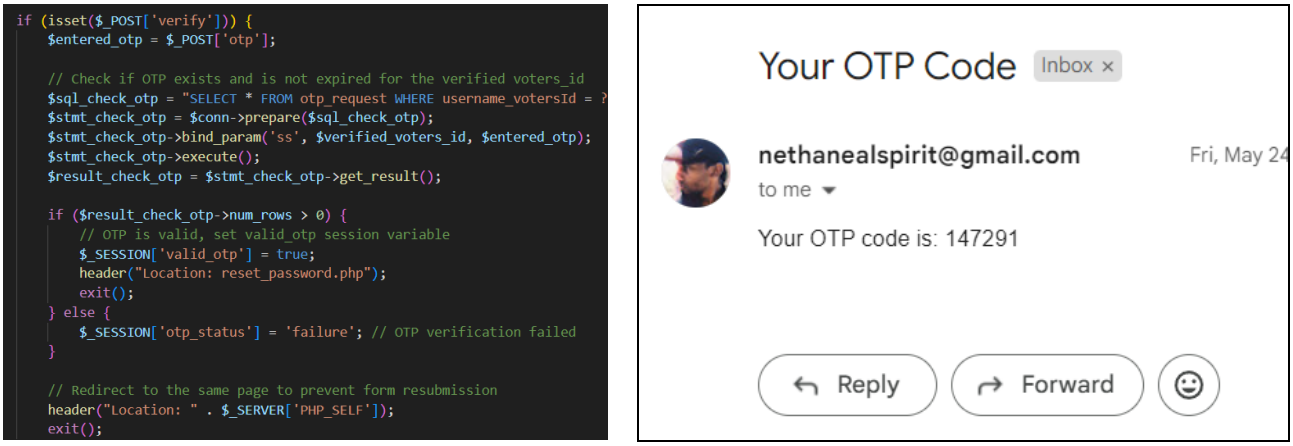
| voters_id | candidate_id ▲ 1 | position_id |
|---|---|---|
| $2y$10$npka6erGYyKLK3FcOB.yD.k4JeE7oZN9FIX7RFR5JML... | 1 | 1 |
| $2y$10$WjUrOPalebLdMGA1Xfj8De4P34OK7kdZ1bHxFjdKpkb... | 2 | 2 |
| $2y$10$HhD2mujKoUrhIbu3J0TB1uGtEGlMsb5c0715HDi3TvM... | 3 | 4 |
| $2y$10$UTuyRLrygSh1lu/6vAatMeaFso9F0C4uSrJBySRcoEE... | 4 | 3 |

**Fig. 12** *The hash value which is stored in votes table*

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 174-192

### 5.1.2 Email-based OTP

The One-Time Password (OTP) enhances user authentication during password recovery in the e-voting system. Users receive a unique OTP via email with a limited validity period, typically 5 minutes. By using OTP, the system ensures only authorized users can reset passwords, reducing phishing risks and enhancing overall security. This approach strengthens the system's security framework, ensuring user account integrity and resistance to potential security attacks. Implementation of Email-based OTP in the system is shown in Figure 13.

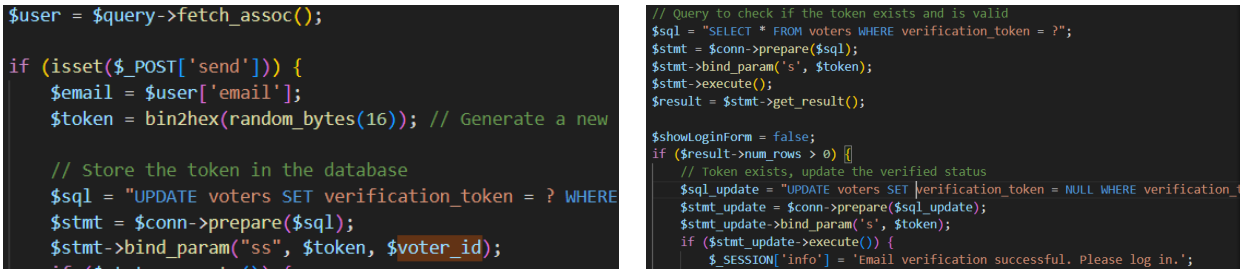



**(a)** **(b)**

**Fig.13** *Email OTP (a) Validation process of the OTP; (b) OTP that is sent to the email.*

### 5.1.3 Email-based verification

Email-based verification is a key security measure in the E-voting system, ensuring the authenticity of voter and preventing unauthorized access. When users click 'vote now,' they enter their voter ID, and if the account exists, they are redirected to send a verification email. A unique verification token is sent to their email and stored in the database. Users click the link in the email, which takes user to the login page, verifying their identity if the token is valid. This process ensures only legitimate voters can vote and helps with password recovery by sending a one-time password (OTP) to the verified email for resets. Figure 14 shows the email verification process within the code and Figure 15 illustrates the verification link.




**(a)** **(b)**

**Fig.14** *The email verification process (a) token is generated and stored in database.; (b) token is compared with database*

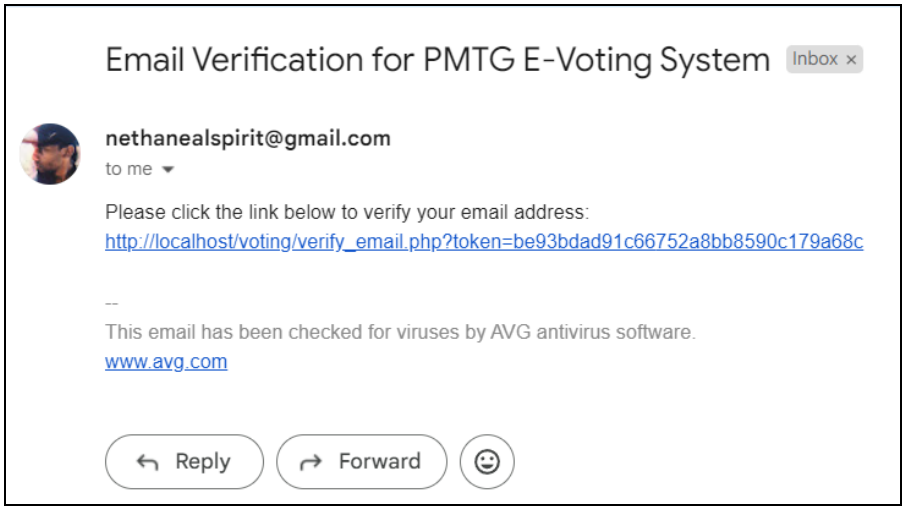



**Fig. 15** *The verification link in the email*

## 5.2 Module Implementation

Module Implementation section outlines the implementation of various modules in the e-voting system, enhancing efficiency, security, and usability. Dividing the system into discrete modules simplifies development, testing, and maintenance, covering all aspects from authentication to vote counting.

### 5.2.1 Login

The Login Module authenticates users (voters and administrators) via a secure interface where they input their credentials (username/voter ID and password). Bcrypt hashing secures passwords, protecting login details. Error handling is included for incorrect credentials. Upon successful authentication, users access the system based on their role. Figure 17 illustrates the login module for voters and administrators.


**(a)** **(b)**

**Fig.17** *Login Module. (a) voter page; (b) admin ballot.*

### 5.2.2 Dashboard Module

The Dashboard Module serves as the central hub for admin after logging in. This module provides an overview of the election status, including the number of registered voters, candidates, and votes cast. Dashboard module also consist of manage votes, configure election and other administrative task. The dashboard features quick links to other modules, enabling easy navigation and efficient task management. The header of the page also contains the admin settings where admin can change their details, name, password and logout. Figure 18 illustrates the dashboard module for admin.

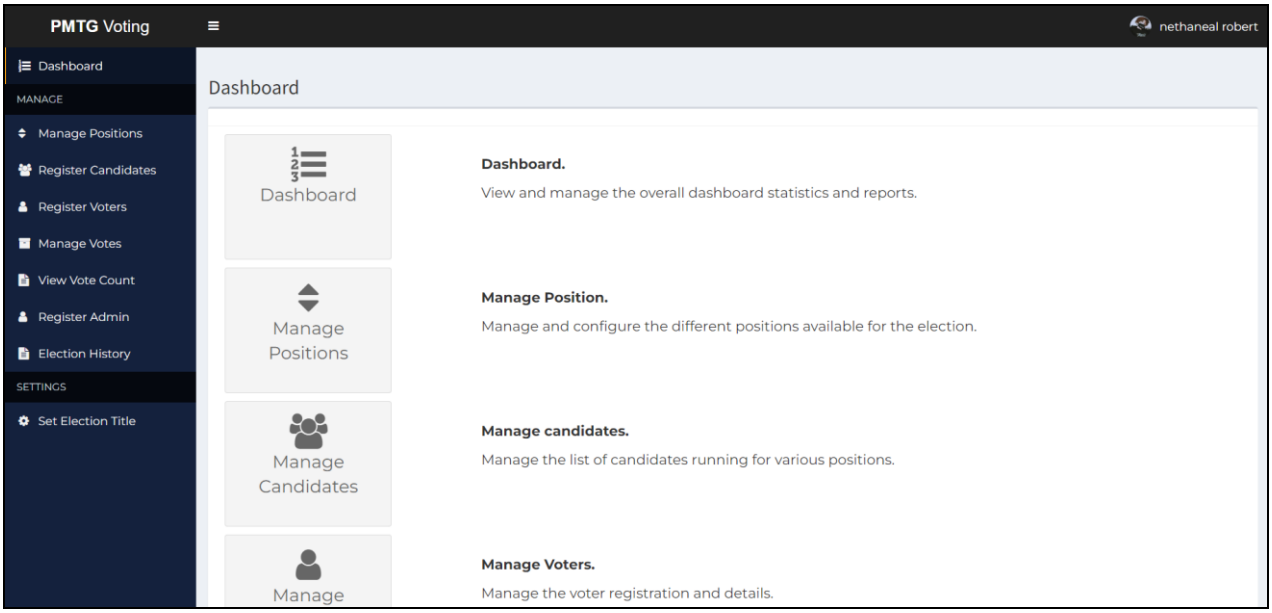
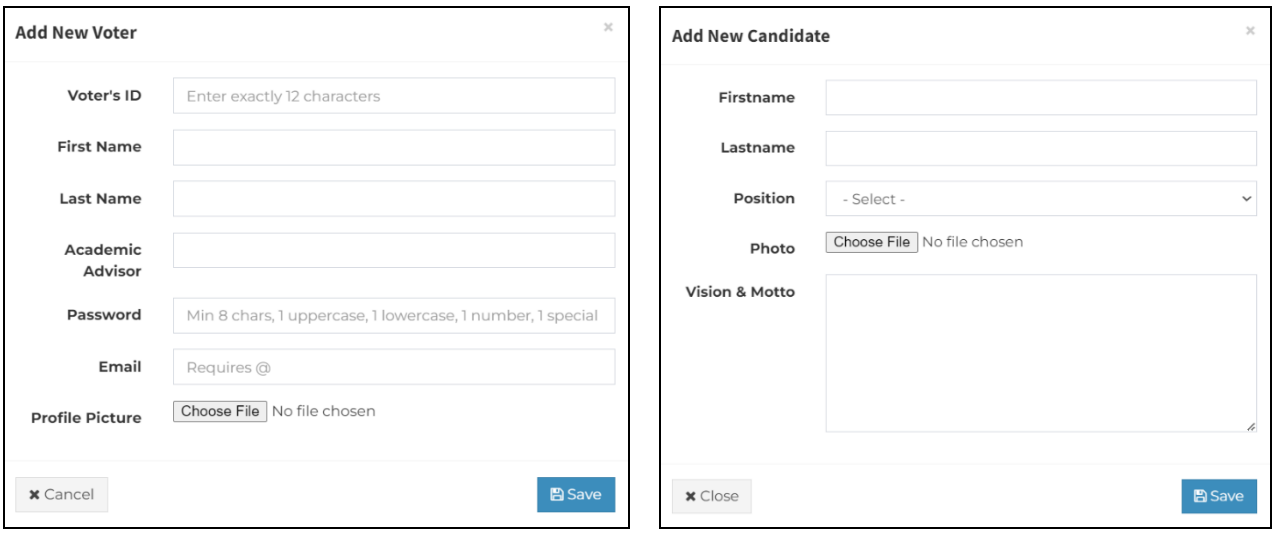



**Fig.18** *Dashboard Module.*

### 5.2.3 Register Module

The Login Module is responsible for authenticating users, including both voters and administrators. This module provides a secure interface where admin and voter can enter their credentials (username/voter id and password). It uses Bcrypt hashing to secure passwords, ensuring that login details are protected against unauthorized access.

The login process includes error handling for incorrect credentials. Upon successful authentication, users are granted access to the system based on their role, either as voters or administrators. Figure 19 shows the login module for voters and admin respectively. The login module for voter is not accessible directly within the system instead through an email verification step which enable voter to login.




**(a)** **(b)**

**Fig. 19** *Registration Module. (a) register voter; (b) register candidate.*

### 5.2.4 Manage Candidate Module

The Candidate module allows admin to manage candidate information by adding, deleting, and editing details. Only admins can register candidates, with eligibility based on CGPA and participation rate in MPP activities. The module ensures data accuracy and completeness through form validation and input sanitization. Figure 20 displays the Manage Candidate module in the admin interface.

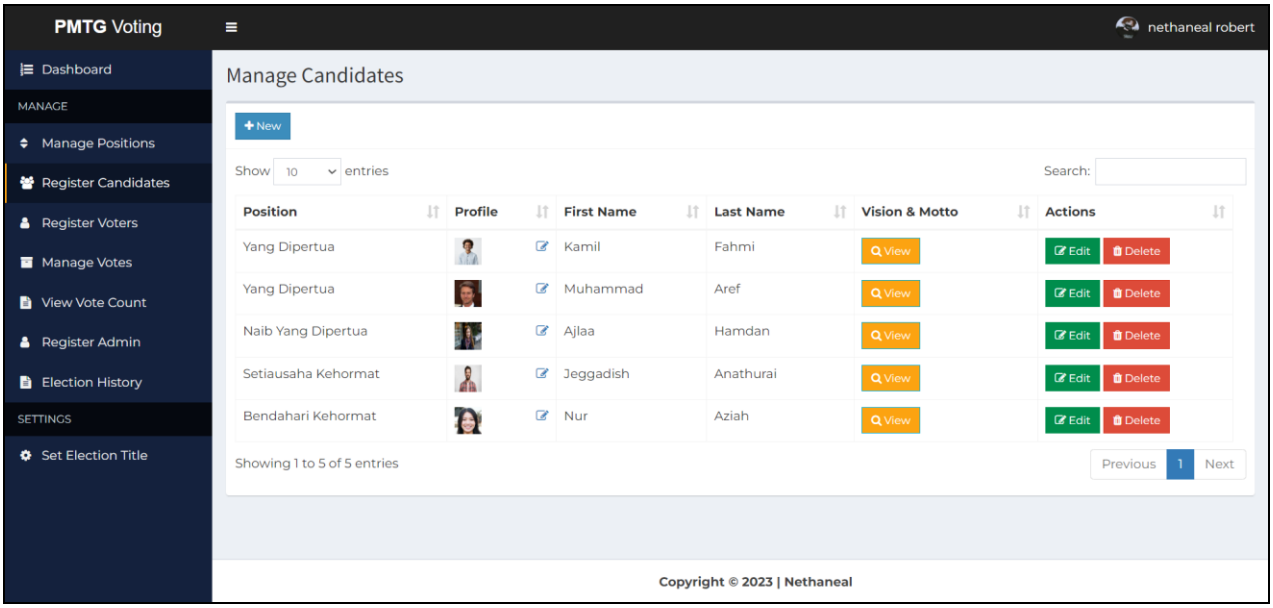



**Fig. 20** *Manage Candidate module*.

## 5.2.5 Manage Voters Module

The Manage Voters Module oversees voter management in the system, ensuring accurate registration and maintenance. Admins can update voter information and detect/prevent duplicates and fraudulent entries. Unique identification and verified emails prevent multiple entries, fostering a secure voting environment. Figure 21 showcases the Manage Voters module in the admin interface.

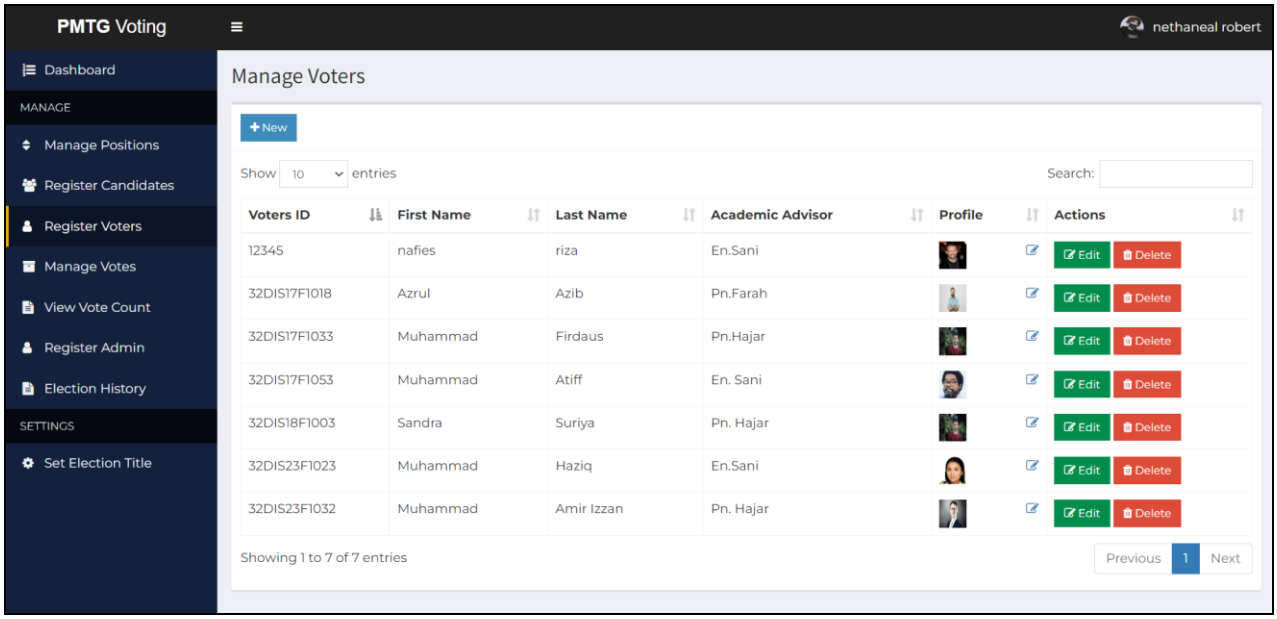
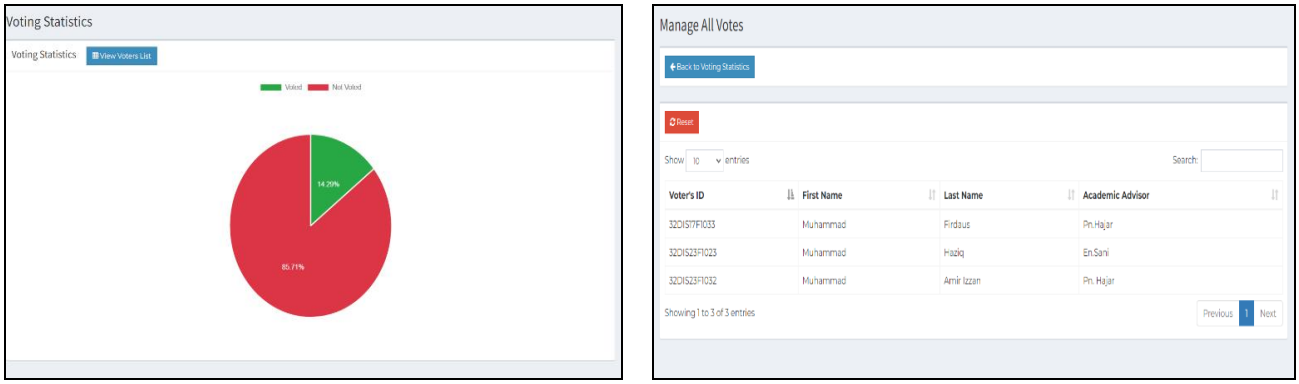



**Fig. 21** *Manage Voters module.*

## 5.2.6 Manage Votes Module

Manage Votes Module tracks and manages all votes cast in the election, ensuring accurate recording and secure storage. It features a pie chart displaying the ratio of students who have voted versus those who haven't. Furthermore, the "View Vote List" option leads to a detailed page that shows which individual participants voted in the current election, allowing the administrator to track voter involvement.. Figure 22 shows the Manage votes module.

Applied Information Technology and Computer Science Vol. 5 No. 2 (2024) p. 174-192



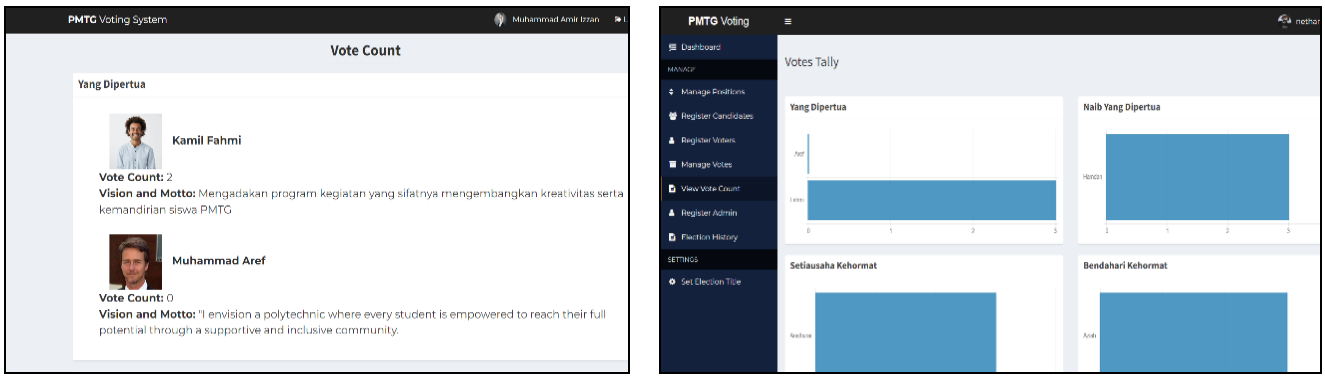

**(a)** **(b)**

**Fig.22** *Manage Votes Module. (a) Pie chart; (b) Voter's list.*

### 5.2.7 View Vote Count Module

The View Vote Count Module is critical for ensuring transparency and delivering real-time updates on election results to both voters and administrators. This module provides a detailed snapshot of the current vote tallies, keeping all stakeholders updated throughout the voting process.

Admins access detailed data and analytics, aiding in monitoring and decision-making during the voting process. Historical data enables comparison with past elections for insights into voter behavior. For voters, the module offers real-time vote totals for each candidate and position, fostering transparency and trust in the system's integrity. Figure 23 showcases the View Vote Count module for both voters and administrators




**(a)** **(b)**

**Fig.23** *View Vote Count module (a) voter; (b)admin.*

The View Vote Count Module is vital for transparency, offering real-time updates on election results to voters and administrators. It provides a detailed overview of current voting tallies, keeping all stakeholders informed during the voting process.

### 5.2.8 Election History Module

The Election History Module is a crucial component of the e-voting system that ensures a comprehensive archive of all past electoral activities. This module maintains detailed records of previous elections, including essential information such as the election title, candidates, and results. By preserving this data, the module enables administrators to view and analyze historical election data, providing valuable insights and supporting informed decision-making for future elections.

The Election History Module in which this module allows administrators to upload generated election results in various PDF format for further analysis or sharing with other admins. This feature ensures that administrators can easily disseminate election results and analyses. Figure 24 illustrates the election history module in admin interface.

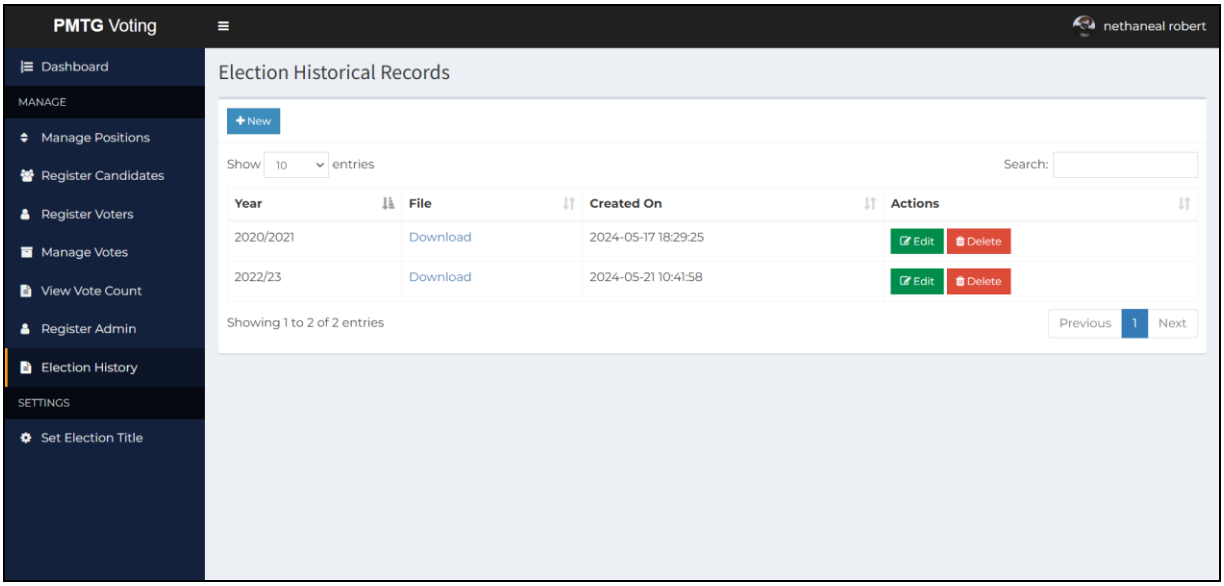



**Fig. 24** *Election History module*

## 5.2.9 Set Election Title Module

The Set Election Title Module as shown in Figure 25 enables administrators to define and manage election details, including the election title and start/end times. This module ensures clarity and organization in distinguishing between different elections. Administrators can set precise timestamps for the election, marking the voting period's start and end for efficient scheduling and time management.

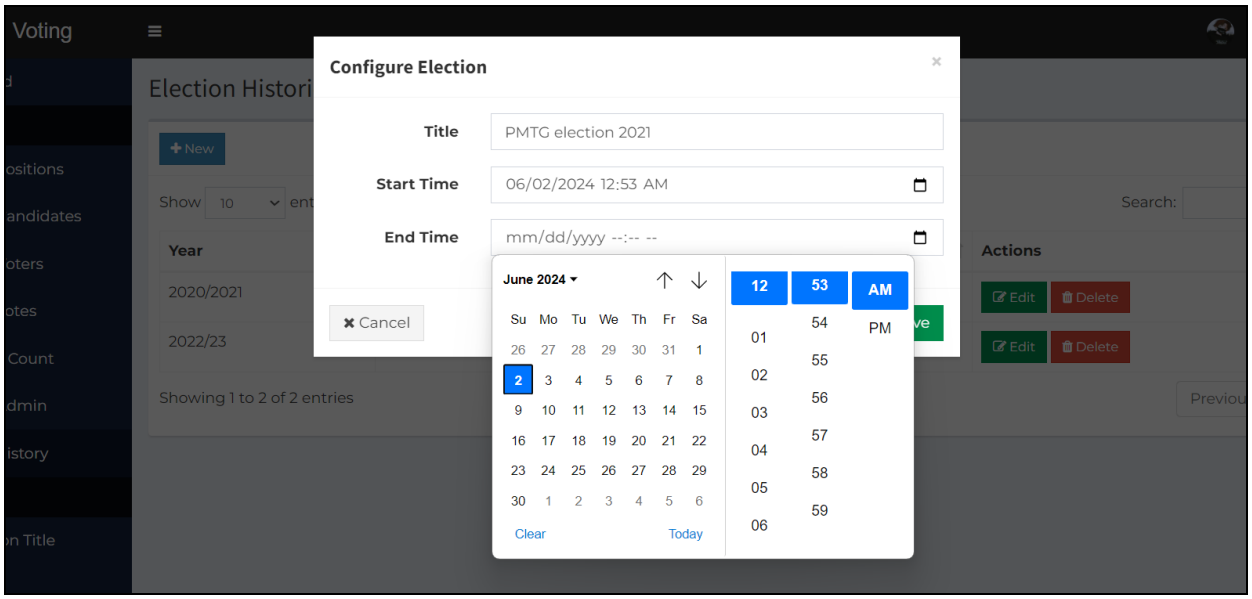



**Fig. 25** *Set Election Title module*

## 5.2.10 Voting Module

The Voting Module which is shown in Figure 26 is crucial for the E-voting system, offering a user-friendly interface for voters to select candidates securely. It ensures robust security measures to prevent tampering and multiple voting attempts. After submitting votes, voters receive immediate confirmation and can view the vote count. Additionally, the module displays the election timeline, including start and end times, to keep voters informed and engaged throughout the process. These features streamline the voting process while enhancing security, transparency, and reliability.

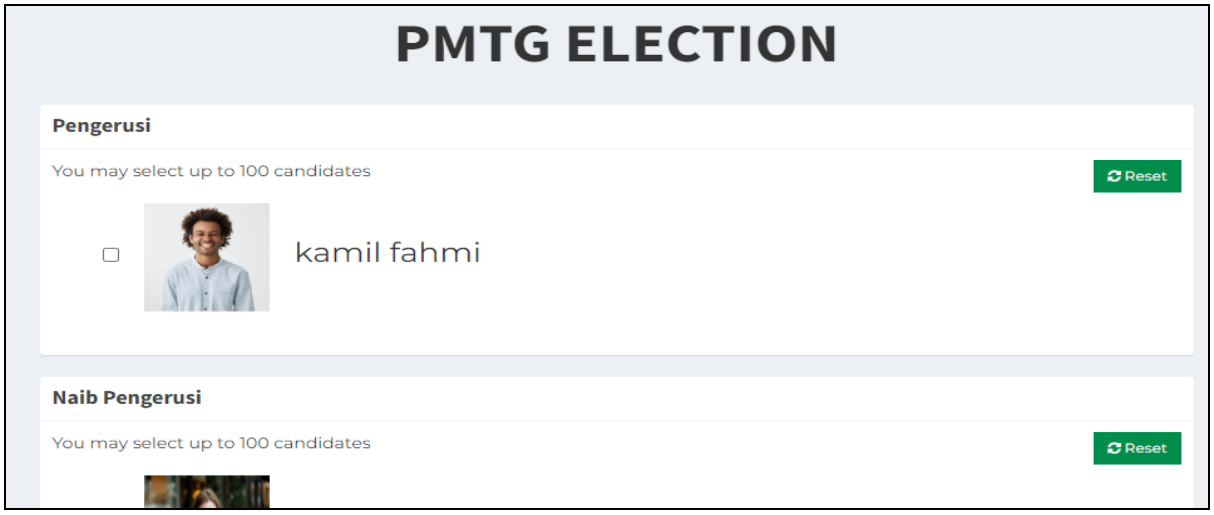



**Fig. 26** *Voting module*

## 5.3 Testing

Functional testing, security testing, and user testing are integral components of ensuring the effectiveness, reliability, and user satisfaction of the E-voting system. Functional testing verifies that all software components align with requirement specifications, while security testing focuses on identifying and mitigating potential vulnerabilities. User testing, conducted via Google Forms with two administrators and thirteen voters, evaluates both the system's interface and functionality, confirming adherence to user expectations. These testing processes validate the E-voting system's robustness, security, and usability, ensuring readiness for deployment. Table 4 and Table 5 shows the voter and admin functional testing results

**Table 4** *Voter functional testing*

| Description | Pass | Fail |
|---|---|---|
| System can be executed from start to end. | 13 | 0 |
| Voters can login and logout from the system. | 13 | 0 |
| Voter able to vote within voting period. | 13 | 0 |
| Voters able to select preferred candidate. | 13 | 0 |
| Voters able to submit vote without any error. | 13 | 0 |
| Voters able to view vote count only after voting and within time frame of the election. | 13 | 0 |

**Table 5** *Admin functional testing*

| Description | Pass | Fail |
|---|---|---|
| The system can be executed from start to end. | 2 | 0 |
| Admin able to login and logout from the system. | 2 | 0 |
| Admin able to navigate to respective pages in dashboard. | 2 | 0 |
| Admin able to view, add, update and delete positions. | 2 | 0 |
| Admin able to view, register, update and delete another admin. | 2 | 0 |
| Admin able to view, register, update and delete voters. | 2 | 0 |
| Admin able to view, register, update and delete candidates. | 2 | 0 |
| Admin able to view the percentage of voted students and who has voted. | 2 | 0 |
| Admin able to view vote count of the candidates. | 2 | 0 |
| Admin able to view, add, update and delete election history records. | 2 | 0 |
| Admin to able to configure election. | 2 | 0 |
| Admin able to reset password after successful OTP verification. | 2 | 0 |

### 5.3.1 User Acceptance Test

User testing was conducted with separate forms for administrators and voters which include two admin and thirteen voters. In section A, users rated the system's interface on a scale from one to five, while section B evaluated functionality. Results confirmed the E-voting system's compliance with user requirements, with feedback indicating satisfaction. Figure 27 displays ratings of the user of the system. The testing outcome

indicated successful user function testing. Voter testing showed that the method was easy to use and efficient, with voters able to navigate the system seamlessly.

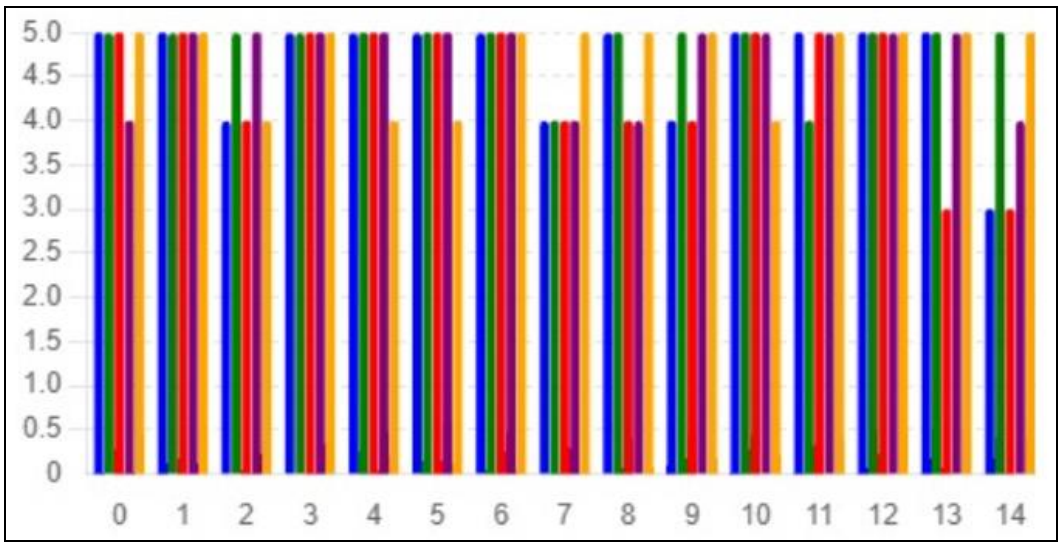



**Fig. 27** *User Acceptance test results*

## 6. Conclusion and Future Works

This section documents the journey of developing the Secure E-Voting Web-Based System for Politeknik METrO Tasek Gelugor. Objective Achievement: The project successfully met its objectives of designing a modular system using Visual Studio Code and testing its functionality thoroughly. System Advantages: The system enhances secrecy, authentication, and voting via the implementation of Bcrypt, password authentication, email-based OTP and email verification. System Limitations: Email verification after registration, centralized candidate registration, and reliance on single-factor authentication pose challenges. Future Works: Improvements include email verification after login, expanding user roles for candidate registration, and implementing multi-factor authentication. Conclusion: Despite drawbacks like email verification issues, the system achieves its objectives, providing modularity, security, and functionality.

## Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, for its support.

## References

[1] Z. Li, D. Wang, and E. Morais, "Quantum-Safe Round-Optimal Password Authentication for Mobile Devices," IEEE Trans Dependable Secure Comput, vol. 19, no. 3, pp. 1885–1899, 2022, doi: 10.1109/TDSC.2020.3040776.

[2] A. R. Reyes, E. Festijo, A. Roy, L. Reyes, E. D. Festijo, and R. P. Medina, "Enhanced Multi-factor Out-of-Band Authentication En Route to Securing SMS-based OTP," 2019. [Online]. Available: https://www.researchgate.net/publication/331949136

[3] V. Agrawal, R. K. Paliwal, P. Sharma, and A. Shrivastava, "Web security using user authentication methodologies: CAPTCHA, OTP and User Behaviour Authentication." [Online]. Available: https://ssrn.com/abstract=3360306

[4] M. Hassan, Z. Shukur, and M. K. Hasan, "An Improved Time-Based One Time Password Authentication Framework for Electronic Payments," International Journal of Advanced Computer Science and Applications, vol. 11, pp. 359–366, Nov. 2020, doi: 10.14569/IJACSA.2020.0111146.

[5] Ruzaini Amir, "Electronic Voting System (E-Voting) of MPPUTP Election."

[6] A. Sinha and P. Das, "Agile Methodology Vs. Traditional Waterfall SDLC: A case study on Quality Assurance process in Software Industry," in 2021 5th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), 2021, pp. 1–4. doi: 10.1109/IEMENTech53263.2021.9614779.