**ICAST: STREAMLINING ELECTIONS THROUGH AN INTEGRATED**

**WEB-BASED E-VOTING SYSTEM**

A System Proposal presented to the

Faculty of the College of Computer and Information Sciences,

Polytechnic University of the Philippines

In partial fulfillment for the course

INTE 40163 – Capstone Project 1

***Proponents***

Chavez, Jabez Victor P.

Legaspi, John Raymond T.

Maldecir, Lenard Dean V.

Yandoc Francis Emmanuel S.

BSIT 3-4

**September 2023**

TABLE OF CONTENTS

[CHAPTER 1 - INTRODUCTION 1](#_Toc82007528)

[1.1. Project context 1](#_Toc82007529)

[1.2. Technical Background 2](#_Toc82007530)

[1.2.1. Equipment/Hardware 2](#_Toc82007531)

[1.2.2. Software 2](#_Toc82007532)

[1.2.3. Peopleware/Manpower 3](#_Toc82007533)

[1.2.4. Network Infrastructure/Architecture 3](#_Toc82007534)

[1.2.5. Storage, Backup and Recovery Procedure 3](#_Toc82007535)

[1.2.6. Security Procedures 4](#_Toc82007536)

[1.2.7. Policies and Procedures 4](#_Toc82007537)

[1.2.8. Data and Process 5](#_Toc82007538)

[1.3. Problem Analysis 6](#_Toc82007539)

[1.3.1. Fishbone Diagram 6](#_Toc82007540)

[1.3.2. Problem and Solution Statement 7](#_Toc82007541)

[1.3.3. Problem – Requirements Matrix 8](#_Toc82007542)

[1.4. Purpose and Description 8](#_Toc82007543)

[1.5. Specific Objectives 9](#_Toc82007544)

[1.6. Scope and Limitations 9](#_Toc82007545)

[CHAPTER 2 – REVIEW OF RELATED LITERATURE/SYSTEMS 10](#_Toc82007546)

[CHAPTER 3 – METHODOLOGY 15](#_Toc82007552)

[3.1. Requirements Analysis 15](#_Toc82007553)

[3.1.1. Requirements – Features Matrix 15](#_Toc82007554)

[3.1.2. Use Case Diagrams 16](#_Toc82007555)

[3.1.3. Use Case Reports 21](#_Toc82007559)

[3.1. Design Specifications 28](#_Toc82007560)

[3.1.1. Activity Diagram 28](#_Toc82007561)

[3.1.2. Class Diagram 32](#_Toc82007562)

[3.1.3. GUI Design 33](#_Toc82007563)

[3.1.4. Database Schema 50](#_Toc82007564)

[3.1.5. Data Dictionary 51](#_Toc82007565)

[3.2. Development Methodology 71](#_Toc82007566)

[3.2.1. Process Model 71](#_Toc82007567)

[3.2.2. Development Tools 72](#_Toc82007568)

[3.3. Test Methodology/Procedures 74](#_Toc82007569)

[3.4.1 Testing Plan 74](#_Toc82007570)

[3.4.2 Scope of Testing 74](#_Toc82007571)

[3.4.3 Testing Procedures 75](#_Toc82007572)

[3.4. System Requirements 75](#_Toc82007573)

[3.5.1 Hardware Requirements 75](#_Toc82007574)

[3.5.2 Software Requirements 76](#_Toc82007575)

[3.5. Quality Plan 76](#_Toc82007576)

[3.6. Evaluation Plan 80](#_Toc82007577)

[REFERENCES 81](#_Toc82007578)

# CHAPTER 1 - INTRODUCTION

## **Project context**

Nowadays, in many educational institutions, manual ballot counting for elections, although a conventional approach, has been demonstrated to be time-consuming, susceptible to errors, lacking a centralized database for comprehensive voter records, and limited in its ability to reach every participant effectively.

To bridge this gap, the capstone project proposes the implementation of an e-voting system designed for elections. By leveraging the capabilities of automation and technology, the proposed e-voting system endeavors to mitigate the inherent inefficiencies and limitations entailed in manual ballot counting operations. The central objective of the e-voting system is to provide a seamless and comprehensive platform where participants can cast their votes conveniently and securely.

With the help of IVOTE++, which stands as an existing voting system that caters for election for various organizations, also a solid foundation upon which the iCast is being developed. The system will represent a significant upgrade and innovation that will introduce new features to enhance the voting experience. iCast is designed to build upon the strengths of IVOTE++, it will introduce an account system for users to ensure confidentiality and vote validity. The platform will also bring remarkable flexibility, allowing for seamless adjustments to meet specific candidate requirements during elections. Moreover, iCast includes a data analysis component that delves into the results, filtering and interpreting voters' choices to provide a deeper understanding of election dynamics.

iCast will simplify voting procedures, eliminate time-consuming manual counting, and provide instant and accurate results. The system will also enhance accessibility by allowing participants to cast their votes remotely and promoting higher voter turnout. Additionally, the implementation of this e-voting system will create a centralized database for all votes. This centralized approach will streamline data management, enhance transparency, and minimize the risk of errors or discrepancies. By maintaining accurate records, the system will ensure the integrity of the electoral outcomes and provide a reliable source of information for post-election analysis.

|  |  |
| --- | --- |
| **Side-by-Side Features of iVOTE++ and iCast** | |
| **iVOTE++** | **iCast** |
| Unique Passcode Generation (physically given face to face before the election starts) | Unique Passcode Generation |
| Level-based Voter Routing | Level-based Voter Routing |
| Party Affiliations | Party Affiliations |
| Built-in Survey System | Built-in Survey System |
| Real-time Voting Statistics (Only admin view the results) | Real-time Voting Statistics |
| Generation of Survey Results | Generation of Survey Results |
|  | **Filing of Candidacy** |
|  | **Customization of Candidacy Requirements** |
|  | **Data Analysis (Chart of Partial Voting Results)** |
|  | **Party Registration Certificate** |
|  | **Online Voting** |

*Table 1. Features of iVOTE++ and iCast*

* 1. Technical Background
     1. Equipment/Hardware

|  |  |
| --- | --- |
| Personal Computer/Laptop | |
| Category | Specifications |
| **Processor** | AMD Ryzen™ 5 3600 @3.6GHz up to 4.2GHz |
| **Installed RAM** | 16.00 GB |
| **Storage** | 256 GB SSD |
| **Graphics** | Dedicated GPU |
| **Network Connectivity** | Ethernet/Wireless |
| **Others** | Monitor and Printer |

*Table 2. Equipment/Hardware*

* + 1. Software

|  |  |
| --- | --- |
| Category | Specifications |
| **Operating System** | Windows 10 Professional 64 bit |
| **Productivity Tools** | Microsoft Office Professional |
| **Web Browser** | Google Chrome |

*Table 3. Software*

* + 1. Peopleware/Manpower

|  |  |
| --- | --- |
| Position | Responsibilities |
| **UI/UX Designer** | Focuses on designing interfaces that provide an intuitive and enjoyable voting experience for users. |
| **Server-side Developer** | Managing software systems, problem-solving, meeting user needs, project management, team leadership, and supervising the members assigned tasks. |
| **Technical Support Specialist** | Aids users who encounter issues with registration, voting, or any other aspect of the system. |
| **Administrator** | Manages the system's database, including the maintenance, and optimization. |

*Table 4. Peopleware/Manpower*

* + 1. Network Infrastructure/Architecture

Since the goal is to replace the manual tradition of election and implement an automated one, and as the elections are being done by manual operations, therefore there is no network infrastructure in place.

* + 1. Storage, Backup and Recovery Procedure

To keep all of the data that was processed by the system, the system will use a database management system (DMBS), specifically a MySQL server. Considering it can provide data integrity, flexibility, and SQL-based querying, MySQL server was selected as the storage.

The system will perform regular backups as part of its backup strategy to make sure the database has every necessary record. In order to safeguard the data required to prevent catastrophes like system failures, on-site and off-site backups will also be put into action.

A step-by-step composed recovery plan will be implemented as part of our recovery strategy in the event that a system failure happens. This plan will instruct users on how to reinstate and restore the database in the event of an error.

* + 1. Security Procedures

The iCast is fortified by a well implemented security procedures, encompassing user authentication, data encryption, role-based permissions and regular audits. By adhering to precise server security measures, access controls, and privacy protection, we ensure a secure environment where user data remains confidential and the electoral process maintains its integrity.

* + 1. Policies and Procedures

1. The digital platform or suite of tools that facilitates voting, decision-making, and management processes within the system.
2. All users are eligible to participate in the voting while the administrators are eligible in the decision-making processes.
3. All registered users of the organization who will be using the system are eligible to access and participate in iCast.
4. Any aspirant user can submit their application for candidacy through iCast.
5. Initiatives must undergo a review process by a designated committee to ensure alignment with the organizations’ objectives and values.
6. Approved initiatives will proceed to the voting stage.
7. The voting parameters, such as voting method (e.g., electronic ballots, surveys) and voting duration, will be determined for each initiative.
8. The voting and management system will implement security measures to protect user data, prevent unauthorized access, and ensure the integrity of the voting process.
9. Personal information and voting records will be handled in compliance with data protection regulations.
10. After the voting period, results will be tabulated and analyzed.
11. The administration will review the outcomes and make informed decisions based on the voting results.
12. Partial and final results will be displayed to foster transparency during and after the election.
    * 1. Data and Process **(if applicable)**

* **Voting Elections:** Using their authentication credentials, eligible voters sign into the electronic voting platform. For each position, voters choose their preferred candidates. The mechanism makes sure that each voter can only cast one vote, and only for the positions for which they are eligible.
* **Filing of Candidacy:** Candidates file their candidacy through an online form. Data entered includes personal information, positions running for, and other required details. Submitted data is validated for completeness and accuracy.
* **Tally of Ballots:** The process of tallying votes within the iCast is a meticulous and transparent procedure designed to ensure accuracy and fairness. Once the voting phase concludes, the system displays the cast votes for each candidate and position. Algorithms and secure protocols are employed to verify the legitimacy of votes, detect any irregularities, and prevent any unauthorized manipulation.
* **Political Party Registration**: Political parties provide their registration details through an online form. Information includes party name, logo, and other relevant information. The submitted data is reviewed and validated by election administrators. It will print a PDF file to display the name of the candidate and what party they belong.
* **Data Analytics**: Filters the results of the election to make sure that the provided data are accurate and reliable.
* **Flexibility and Customization**: The administrative team of the organization possesses the capability to fully customize the requirements of the election, tailoring them to specific requirements, preferences, and unique needs, thereby ensuring a seamlessly user-friendly experience for all the organization that is involved in the electoral process.
* **Generation of Survey Results and Voting Results**: Following the conclusion of the elections, iCast exhibits its remarkable efficiency by compiling and processing the voting data, culminating in the generation of a concise and comprehensive summarized PDF file presenting the results, all accomplished with the utmost convenience through the simple action of a single click. This feature not only streamlines the post-election administrative tasks but also enhances transparency and accessibility for the organization involved in the electoral process.
  1. Problem Analysis
     1. A diagram of a process

        Description automatically generated with medium confidenceFishbone Diagram

## 

*Figure 1. Fishbone Diagram*

* + 1. **Problem and Solution Statement**

Traditional manual voting procedures in many settings provide several difficulties. These include the lengthy manual vote counting process, the possibility of mistakes, and the lack of a centralized voter database. This conventional procedure has the potential to deny participants their right to vote, lessen transparency, and obstruct the democratic process' effective functioning.

Our proposed solution addresses these issues by developing an electronic voting system with the capability of reducing dependence on human resources while providing step-by-step guidance. It includes the integration of a centralized database for secure voter information storage, automated vote tabulation for accuracy and efficiency, the implementation of data analysis of the tallied votes to filter the results, and the transition to electronic voting to expedite the process and ensure timely results.

* + 1. Problem – Requirements Matrix

|  |  |
| --- | --- |
| **Problem** | **Solution** |
| **Unreliable and Error-Prone Data Entry** | Voter information and ballot counts must be entered manually during manual voting, which can result in transcription and data management mistakes. Through automated digital records and direct online voting, iCast's solution will do away with error-prone data entry, lowering the possibility of inaccurate voter data and election outcomes. |
| **Voter and Ballot Confidentiality** | The secrecy of a voter's choices may not always be guaranteed by manual voting because other people may be able to observe their choices. By offering a secure online platform where voters can secretly cast their ballots, iCast ensures ballot confidentiality while maintaining the fairness of the electoral process. |
| **Storage and Retrieval of Actual Ballots** | Managing actual ballots can be tedious because it requires physical storage and needs to be protected for a long time. By maintaining digital records in a database system, iCast eliminates the need for physical ballot storage and streamlines the storage, retrieval, and security of election data. |
| **Limited Voters Engagement** | Because manual voting procedures lack the accessibility and convenience of online options, they might not increase voter participation and turnout. They may not be able to vote for a number of reasons, including disabilities or a personal issue. By offering iCast it encourages participation and enables voters to use the system at their convenience. |
| **Timely Results Reporting** | Election results announcement delays brought on by manual vote counting may affect the process' overall transparency and trust among voters. Through real-time vote counting and instant data analysis, iCast ensures that results are reported promptly, giving users quick access to reliable election results. |

*Table 5. Problem - Requirements Matrix*

* 1. Purpose and Description

The core purpose of this study is to implement an efficient electronic voting (e-voting) system that replaces the time-consuming manual procedures used in elections. This innovative system is designed to promote inclusivity, establish a centralized voter database, and offer an efficient that replaces the traditional manual methods previously employed. Our primary goal is to develop an e-voting solution that simplifies and automates tasks such as the filing of candidacy and registering political parties, ultimately transforming the way elections are conducted.

The intended outcomes of this endeavor are numerous. Through the implementation of iCast, we seek to reorganize the process of completing requirements of the candidates. By providing candidates with a user-friendly digital interface, we aim to enhance accuracy and reduce errors that often arise from manual data entry. This, in turn, contributes to maintaining the integrity of the election process by minimizing discrepancies in candidate information.

Additionally, iCast will serve as a cornerstone for managing certificates for political party registrations. By digitizing this process, we eliminate the need for labor-intensive paper submissions and create a secure online platform for parties who are officially registered. This approach streamlines the registration process and enables effective communication between election administrators and political parties, ensuring all requirements are met. Also, the system will also feature a flexible and customizable forms depending on what the election needs.

Beyond its administrative benefits, the e-voting system has the potential to enhance inclusivity. With a centralized voter database, eligible participants will have easier access to participate in elections through electronic voting. This accessibility not only encourages higher voter engagement but also reduces the chances of irregularities like double voting. Lastly, the real-time results will help the election to foster transparency and display the partial and final results of votes so that the users will be able to see it digitally and hassle free.

* 1. SPECIFIC OBJECTIVES

• Implement a secure and reliable platform that ensures the integrity and confidentiality of the voting process.

• Create a centralized database to store and manage all voter information, applicants filing for candidacy, facilitating efficient data management. analysis, and auditing.

• Streamline the election process by eliminating the time-consuming manual counting of ballots and introducing automated result tabulation.

• To develop forms that are flexible and customized to the specifications of a particular election

• Foster transparency by implementing real-time statistics results of the election and auditing of the electoral process to instill trust among the population.

• Lastly, to provide comprehensive documentation and guidelines for the system's implementation, maintenance, and future enhancements.

* 1. Scope and Limitations

The scope of the project is solely focused on implementing a web-based voting system that will take the place of the current manual elections. It focuses primarily on the attributes of automated results, facilitation of seamless digitization of political party registration certificates and easy candidacy filing that will eliminate the traditional way of paperwork associated with traditional methods. In addition, the feature of flexibility in terms of organizational demands, and finally the goal of transparency to all users in order to foster trust among them. Lastly, this empowers users to engage in the election process conveniently through a user-friendly online platform.

One constraint of the iCast is the reliance on technology and internet connectivity. To participate, users must have access to devices like computers or smartphones and stable internet connections. Also, device compatibility may also be an issue that may arise in the scenario. This reliance might inadvertently exclude individuals with limited technological resources, potentially leading to unequal participation. While the system employs and practices security measures, no digital system is entirely immune to cybersecurity threats.

# CHAPTER 2 – REVIEW OF RELATED LITERATURE/SYSTEMS

**Authenticated Web Based Voting System**

According to the research conducted by Ushashree and colleagues in 2021, an Authenticated E-Voting System (AEVS) was developed with a primary focus on security. The AEVS system was designed with the nation's welfare in mind, specifically to offer an opportunity for residents who are outside the country to participate in choosing their representatives.

Traditionally, the election of representatives involved either paper ballots or automated ballots. While there has been a significant shift from traditional to automated methods, the current system still has some limitations that need to be addressed. The objective of this paper is to introduce AEVS as a highly secure system, intended to replace traditional systems and provide more accurate and immediate election results.

AEVS boasts a high success rate compared to traditional systems, primarily because it prevents users from engaging in multiple registrations and proxy voting. The system places a strong emphasis on authentication, utilizing fingerprint recognition and the unique identification numbers provided by the government. It also addresses voter eligibility and related issues.

Voters are granted the convenience of casting their votes at their own convenience, using devices with an internet connection and their unique identification numbers. To enhance security, an additional authentication step involves answering a security question during the initial registration. Furthermore, a token is automatically generated for each registered individual as a reference. This token serves to prevent duplicate registrations using the same information. Each token is sent to the user's provided email address and has an expiration period, ensuring that it loses validity after a certain time frame.

**Web Based Application For Secure Online Voting System**

According to Ilakkiyavani and Shanthi's study conducted in 2018, Data Mining is a powerful analytical tool utilized for extracting knowledge from extensive datasets with a focus on security to facilitate effective decision-making. This technique has been employed in online voting systems to enhance transparency at the highest level and improve operational efficiency while minimizing the risk of data breaches. Additionally, it enables quicker access to secure data to support effective decision-making processes.

In the past, various methods, such as secret ballots or punch card systems, were used for electoral processes, requiring individuals to physically visit polling booths to cast their votes within the existing system. With the rapid advancement of Information Technology, particularly the proliferation of Android apps, modern voting systems have emerged. These systems allow voters to participate in an automated process, a capability only achievable through Voting Systems.

The proposed online system is especially advantageous as it enables individuals, even those living away from their hometowns, to cast their votes. Increasing voter turnout is a primary objective of this system. The key goal of the proposed system is to provide swift and efficient access to information.

**Responsive Web-Based E-Voting Information System**

According to Cahyo and Suryawinata's study conducted in 2022, rapid technological advancements have brought significant changes to various aspects of human life, including the process of voting. To adapt to this technological growth, voting has evolved to incorporate computer technology, commonly known as electronic voting or e-voting. One of the pressing challenges faced by election organizers, particularly during the COVID-19 pandemic, is the inability to conduct in-person voting.

This research utilized qualitative research methods and followed the System Development Life Cycle (SDLC) method, which involves several stages such as design, analysis, prototype development, and implementation. The e-voting application was developed using PHP programming language, HTML tags, a MySQL database as a backend server, and web hosting. Importantly, this e-voting application is web-based, allowing it to be accessed online.

The research concluded that the online e-voting application successfully met the requirements for the election of the chairman of RT.01 RW.03 Kludan Village. Through testing with 10 random voters and the completion of questionnaires, the application received an acceptance rating of 83%, categorizing it as "very good" in terms of usability and effectiveness.

**Implementation of Design Thinking for Web Based E-Voting Student Organization in Nahdlatul Ulama University of Yogyakarta**

As per Dewi et al. (2022), the aim of this study is to implement design thinking principles within the context of E-Voting for the Nahdlatul Ulama University Student Organization in Yogyakarta, through a website-based approach.

The method employed in this research is design thinking, which encompasses stages including empathizing, defining, ideating, prototyping, and testing. The outcome of this endeavor entails the development of the Student Organization E-Voting platform for Nahdlatul Ulama University in Yogyakarta, effectively realized through a website-based system, aligned with user requirements.

This initiative represents a novel and innovative approach, introducing the Student Organization E-Voting system for Nahdlatul Ulama University in Yogyakarta, employing website-based technology.

**An online e-voting prototype for student's union elections in Jordanian universities**

Obeidallah and Alzamel (2018) highlight that the voting system is among the numerous systems undergoing constant enhancements due to the technological revolution. Electronic voting, referred to as e-voting, is anticipated to enhance the reliability, accuracy, and appropriateness of the electoral process. E-voting is embraced to reinforce the principles of democracy and enhance the authenticity of election outcomes. The primary objective of their study is to introduce a recommended framework for an online e-voting system tailored for student council elections within Jordanian universities. The authors will outline the architecture of the proposed framework and underscore the advantages of adopting such a system, while also addressing potential challenges

**Online Voting System using Cloud**

According to the research conducted by Govindaraj et al. in 2020, voting, commonly associated with the political process, has traditionally been carried out using manual methods, where voters physically cast their ballots. However, manual voting systems can sometimes be susceptible to fraudulent practices. Hence, there is a pressing need to transition to an online voting system, thereby advancing from a manual to a digital voting system.

The primary objective of this research is to implement an online voting system that includes features related to the policies and schemes proposed by specific political parties. Voters can make their selections based on these features. The primary motivation for shifting from the traditional voting system to an online one is the convenience it offers. Online voting allows individuals to cast their votes from anywhere, saving time and effort.

This implementation was achieved using the C# programming language, Microsoft SQL Server 2012, and Microsoft Azure as a cloud platform.

**DATE: A Decentralized, Anonymous, and Transparent E-voting System**

According to "Lai, W.-J., Hsieh, Y.-C., Hsueh, C.-W., & Wu, J.-L. (2018), the establishment of a reliable electronic voting system necessitates the public availability of all pertinent information. This requirement, however, extends beyond mere transparency and encompasses concerns regarding individual privacy. Put differently, the focus is on ensuring that each vote is tabulated in an anonymous, accurate, and efficient manner. The research introduces an efficient electronic voting system aimed at reducing voters' reliance on governmental or authoritative entities. To achieve this, the proposed system leverages the Ethereum blockchain to ensure election transparency. Concurrently, the privacy of individual voters is safeguarded through the implementation of a robust ring signature mechanism.

Furthermore, the system incorporates an appealing self-tallying capability. This aspect ensures that any participant with access to the blockchain network can independently calculate the election outcome, thereby obviating the need for third-party involvement post-voting. Additionally, the research emphasizes the maintenance of voting result accuracy while minimizing the Ethereum gas costs incurred by individual participants. Notably, the system's predefined attributes of stealth address enhance its suitability for conducting large-scale online elections.

**A Survey on Web Based Application of Secure Online Voting System**

According to the study conducted by I. Ilakkiyavani and Amsaveni in 2018, Data Mining serves as an analytical tool utilized to extract valuable insights from extensive datasets, prioritizing data security to facilitate informed decision-making. The Online Voting System strategically employs data mining techniques to enhance transparency to the highest degree, optimizing operational efficiency, mitigating data tampering risks, and ensuring swift access to secure data for effective decision-making.

Historically, various methods such as punch card systems or the secret ballot approach were employed to conduct electoral processes, mandating individuals to physically visit polling stations to cast their votes within the existing voting system. However, with the rapid evolution of Information Technology, modern voting systems have emerged, enabling voters to participate in automated processes that are only achievable through such Voting Systems.

The proposed system is web-based, enabling even those residing outside their hometowns to participate in the voting process, with the primary objective of increasing voter turnout. The central aim of this system is to provide efficient and expeditious information retrieval.

**Privacy preserving E-voting cloud system based on ID based encryption**

According to Shankar, A., Pandiaraja, P., Sumathi, K., Stephan, T., & Sharma, P. (2020), many facets of contemporary life, including banking, transportation, voting, healthcare, and electricity, have embraced digitalization. In their research paper, the authors conducted an investigation into the existing voting systems, highlighting the security and operational flaws present in these systems. The drawbacks encompass issues like insufficient data security, vulnerability to vote tampering, susceptibility to attacks such as man-in-the-middle, masquerade, denial of service, and impersonation attacks. These vulnerabilities compromise security and allow unauthorized modifications to the election outcomes.

To tackle these challenges and vulnerabilities, the research paper proposes a protocol designed to address security concerns. The proposed system introduces a secure online e-voting mechanism developed to ensure the integrity of votes during result announcements, both in India and elsewhere. The system, referred to as the E-Voting Cloud System (ECS), operates across three distinct phases: registration, vote polling, and result announcements. In this setup, the election commission of India employs cloud computing to search and authenticate vote data. The protocol implements a cube data storage structure and a user-differentiated system to enhance security. Data encryption is utilized to safeguard voter information in the cloud and ECS, with candidates possessing the decryption keys. This design empowers individuals to verify the results published by ECS. The performance analysis underscores the robust security of the proposed system compared to existing methods. Consequently, the proposed protocol presents an efficient solution for secure data transfer in online e-voting applications.

**A Computerized Touchscreen Student Voting System for the Universities and Colleges**

In a study conducted by Pereyras in 2019, a computerized touchscreen student voting system refers to a type of computer-assisted voting system wherein student-voters select their preferences using a touchscreen computer interface. The research aimed to evaluate the efficiency of the current student voting systems utilized in various Universities and Colleges, with a specific focus on Pangasinan State University - Lingayen Campus, chosen as a representative sample. The ultimate objective was to propose the adoption of a computerized touchscreen student voting system.

The study employed the descriptive-developmental research method. Data collection techniques encompassed the use of questionnaires, checklists, casual interviews, and observations. The questionnaire data were organized and subjected to tabulation. Analysis of the data was conducted using statistical tools like weighted mean and t-test.

Regarding the effectiveness of the existing student voting system, student organization advisers perceived it as "Effective," while students perceived it as "Somewhat Effective." The research findings also indicated a significant disparity in how respondents perceived the effectiveness of the current student voting system. Furthermore, the proposed computerized touchscreen student voting system garnered high acceptance among students.

Taking into consideration the comprehensive findings and conclusions derived from this research, the author strongly recommends that the management of different Universities and Colleges, including the Pangasinan State University - Lingayen Campus, in coordination with the Students' Affairs Office and the University Student Council, seriously consider implementing the proposed computerized touchscreen student voting system in the near future.

**The Implementation of an Electronic Voting System for Supreme Student Council in the State College**

According to a study conducted by Matos in 2023, the research aimed to conceptualize and put into operation an electronic voting system tailored for the supreme student council at the State College of Zamboanga del Sur in the Philippines. The study had distinct objectives, which were as follows: (1) the automation of the election results, (2) the creation of a user-friendly module for voters to easily cast their votes, (3) the development of a module to assist election officers in managing candidates, and (4) the provision of essential printed reports required by the election officer.

To expedite application development, a rapid application development approach was adopted as the chosen methodology. The system was executed as a web-based application, employing the Hypertext Pre-Processor (PHP) programming language and MYSQL database. Subsequently, stakeholders conducted testing of the system, and it received an above-average rating in terms of usability. The newly established voting system has proven to be a valuable tool for the Office of Student Affairs and the Supreme Student Council during the election processes and canvassing.

**Utilization of uVote Student Electronic Voting System as Voter Education Initiative in Rinconada Secondary Schools**

In a study conducted on the Development Academy of the Phillipines, it was observed that the uVote, also known as the Universal Voting System, has been extended to secondary schools within the Rinconada district since the year 2018. Annually, spanning from January to March, a dedicated extension services team from the Camarines Sur Polytechnic Colleges (CSPC), led by the College of Computer Studies (CCS), oversees the deployment of the uVote system. This includes conducting training sessions for both students and teachers, equipping them with the knowledge and skills to independently utilize the e-voting system.

The decision to introduce the uVote system to secondary schools was rooted in research findings that suggest experiential voting during one's school years increases the likelihood of participating in subsequent elections, including those in adulthood. Furthermore, this initiative aims to serve as a means of educating young voters about their roles, responsibilities, and civic rights. By establishing positive election habits at an early age, it seeks to contribute to a well-informed and responsible electorate.

**A Comparative Study of Manual Voting System and E-voting System for Supreme Student Government Council Election of Paso De Blas National High School**

According to a study conducted by Boado et al. (2019), the annual Supreme Student Government council election is a significant event in Philippine schools. This election is organized by the co-curricular student organization, which has the authority to execute relevant programs and activities as mandated by the Department of Education. During this event, all students are required to cast their votes for candidates who seek to represent various positions on the student council. The primary objective of the Supreme Student Government is to shape and nurture student leaders while encouraging them to become vigilant and active citizens within their respective communities and country (Likigan, 2012).

Consequently, the use of manual election systems has become less favorable. This led the researchers to embark on the task of enhancing the existing voting system at Paso De Blas National High School, transitioning from a manual voting process to an E-Voting system.Paso De Blas National High School represents one of the institutions that still adheres to a manual voting process for its Supreme Student Government Council elections. This election encompasses positions including President, Vice President, Secretary, Treasurer, Auditor, Public Information Officer, and representatives for Grades 7 to 12 (Boado et al., 2019).

**iVOTE++ Voting Application and Survey System**

iVOTE++ is an innovative voting application software engineered to ensure the security and precision of electronic voting processes. Its primary objective is to revolutionize electoral procedures, encompassing the verification of voter and candidate eligibility. What sets iVOTE++ apart are its distinctive features, tailored to address the diverse needs of the organization's stakeholders.

Firstly, it employs a Unique Passcode Generation system, providing each voter with an exclusive passcode for seamless and secure voting. Secondly, iVOTE++ incorporates Level-based Voter Routing, allowing organizations to channel voters to specific candidate sets, in alignment with their hierarchical officer structure. Thirdly, the software embraces Party Affiliations, registering and assigning candidates to respective parties. This feature empowers voters with the option to cast their votes automatically for an entire party. Additionally, iVOTE++ boasts a sophisticated Built-in Survey System, enabling customized exit surveys post-voting, tailored to the organization's specific requirements.

Furthermore, administrators benefit from Real-time Voting Statistics displayed through an intuitive dashboard, offering insights into ongoing voter participation and partial voting results as elections progress.

Finally, iVOTE++ streamlines result dissemination by swiftly generating summarized PDF files of the election outcomes with a single click after the polls close.

**SYNTHESIS**

The studies discussed the challenges faced by current voting systems, including issues of security, privacy, participation, and accuracy. Researchers have emphasized the potential of blockchain technology to address these challenges by providing a secure and tamper-proof framework for recording and tallying votes. By decentralizing data storage and utilizing cryptographic techniques, blockchain-based voting systems aim to enhance trust and integrity in the election process.

The articles highlighted the importance of user participation and engagement, with some studies focusing on student elections within educational institutions. The introduction of web-based and cloud-based e-voting systems provides opportunities for greater inclusivity and ease of access, particularly in the digital age. Moreover, the integration of innovative features like interactive ballots, secure data encryption, and real-time result announcements demonstrates the evolution of election mechanisms to meet the demands of modern society.

In conclusion, the presented research collectively showcases the ongoing efforts to enhance the election process through technology-driven innovations. These studies emphasize the need for secure and participatory election systems, ultimately contributing to the strengthening of democratic processes in various contexts.

# CHAPTER 3 – METHODOLOGY

* 1. Requirements Analysis
     1. Requirements – Features Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **System**  **Features**  **User  Requirements** | Responsive on both web and mobile | Voter’s Feed | Vote Candidate | Account Management | View Live results | Apply for candidacy | Party list Registration Certificate | Request of use | Generate Reports |
| Account generation for students and Comelec | ✓ |  |  | ✓ |  |  |  |  |  |
| Allow other organizations to use the system |  |  |  |  |  |  |  | ✓ |  |
| View results in real time | ✓ |  |  |  | ✓ |  |  |  |  |
| Easy candidacy application | ✓ |  |  |  |  | ✓ |  |  |  |
| Party Registration Certificate | ✓ |  |  |  |  |  | ✓ |  |  |
| Statistical Report Generation | ✓ |  |  |  |  |  |  |  | ✓ |
| Easy information dissemination | ✓ | ✓ |  |  |  |  |  |  |  |

Table 5. Requirements – Feature Matrix

* + 1. Use Case Diagrams

**A diagram of a system

Description automatically generated**

Online Voting and Management System

*Figure 1. General Use Case Diagram*

**REGISTER**

**A diagram of a register

Description automatically generated**

*Figure 2. Detailed Use Case Diagram – Register*

The use case above shows a detailed process of how client admin and system admin will create an account in order to access the voting system.

**LOGIN**

**A diagram of a login

Description automatically generated**

*Figure 3. Detailed Use Case Diagram – Login*

The use case above shows the process of how the users, client admin, and system admin can login their account in order to access the system.

**REQUEST OF USE**

**A diagram of a process

Description automatically generated**

*Figure 4. Detailed Use Case Diagram – Request of Use*

The use case above shows the process of how a client may request access to use our system.

**VOTER’S FEED**

**A diagram of voting process

Description automatically generated**

*Figure 5. Detailed Use Case Diagram – Voter’s Feed*

The use case above shows the process of how a admins can edit, delete, or create announcements which the students see in the homepage.

**CREATE ELECTION**

A diagram of a person's election

Description automatically generated

*Figure 6. Detailed Use Case Diagram – Voter’s Feed*

The use case above shows the process of how an election is made.

**CAST VOTE**

**A diagram of a voting process

Description automatically generated**

*Figure 7. Detailed Use Case Diagram – Cast Vote*

The use case above shows the process of how users will cast their vote in the syste

**VIEW RESULTS**

**A screenshot of a diagram

Description automatically generated**

*Figure 8. Detailed Use Case Diagram – View Results*

The use case above shows the process of showing the live results after a user cast their votes.

**APPLY FOR CANDIDACY**

**A diagram of a application

Description automatically generated**

*Figure 9. Detailed Use Case Diagram – Apply for Candidacy*

The use case above shows the process of a student’s application for candidacy and the process of the client admin of approving the application.

**GENERATE REPORTS**

A diagram of a company

Description automatically generated

*Figure 10. Detailed Use Case Diagram – Generate Reports*

The use case above shows the process of statistical report generation.



### Use Case Reports

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-1 | |
| Use Case Name | Register | |
| Actor | Client Admin, System Admin | |
| Purpose | Allows users to create an account in order to login and use the system | |
| Brief Description | The feature allows users to create an account in order to login and use the voting system. | |
| Trigger | The admins needs to do registration | |
| Pre-Condition/s | N/A | |
| Post-Condition | • The user will be directed to the login page after account completion. | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click the “Register Account” on the main page | | 2. System will display the registration form. |
| 3. Users will fill up the registration form by entering the required details. | | 5. The system will generate an account if the necessary details are inputted, and the user is redirected to the login page. |
| 4.The user will click the submit button. | |  |

*Table 6. Use Case Report – Register*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-2 | |
| Use Case Name | Login | |
| Actor | User, Client Admin | |
| Purpose | Allows users to login their accounts and give access to the system. | |
| Brief Description | This allows users to login to their accounts and use the system. | |
| Trigger | The user needs to login | |
| Pre-Condition/s | • The user must have an existing account. | |
| Post-Condition | • The user will be directed to the home page after login is successful. | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will open “Login page”. | | 2. System will display login form. |
| 3.User will fill input email and password. | | 5.The user will be redirected to the homepage if the right details are entered. |
| 4. User will click login. | |  |
| Alternative Flow | | |
| User Action | | System Response |
| 1. User entered incorrect password or email. | | 2. System will prompt “Incorrect email or password”. |
| 3. The user could press “forgot password” if user forgot the password. | | 4. The system will display forgot password page. |
| 5. User can click “create account” if he/she has not created an account yet. | | 6. System will display account registration page. |

*Table 7. Use Case Report – Login*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-3 | |
| Use Case Name | Voter’s feed | |
| Actor | Client/System Admin | |
| Purpose | Allows users to see announcements. | |
| Brief Description | This is where announcements are posted, deleted, or edited. | |
| Trigger | The user wants to see announcements | |
| Pre-Condition/s | • The user must be an administrator | |
| Post-Condition | • Voter’s feed is updated. | |
| Normal Flow | | |
| User Action | | System Response |
| User accessed the voter’s feed. | | The System will display announcement feed. |
| 3. Comelec will click “New Announcement” button. | | The system allows the user to input text or images to the voter’s feed. |
| Admin will click “Publish” button. | | The system publishes the announcement that was created by the Admin. |
| Alternative Flow | | |
| User Action | | System Response |
| 1. The user will click “Edit” if editing is needed in the announcement. | | 2. The system enables the user to edit the announcement from the feed. |
| 3. The user will click “Delete” if announcement is to be deleted | | 5. The system deletes the announcement |

*Table 8. Use Case Report – Voter’s Feed*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-4 | |
| Use Case Name | Request of Use | |
| Actor | Client Admin | |
| Purpose | Allows users to fill up a request for using the system | |
| Brief Description | This is where educational institutions or other organizations can submit a request in order to use the system. | |
| Trigger | The user wants to submit a request to use the system. | |
| Pre-Condition/s | • The user must be an admin of the school institution or organization that is requesting to use the system. | |
| Post-Condition | • The request is approved by the system administrators | |
| Normal Flow | | |
| Client Admin Action | | System Response |
| 1. User will click “Request of Use” form | | 2. The System will display the Request of Use form |
| 3. User will fill up the Request of Use form | | 5. The system will display “Request submitted”. |
| 4. User will click the “Submit” button. | |  |
| Alternative Flow | | |
| System Admin Action | | System Response |
| 1. System will click “Request of Use” button | | 2. The System will display the request list. |
| 3. System will view the request forms | | 5. The system will prompt “Request approve or Rejected”. |
| 4. System admin can approve or reject requests. | |  |

*Table 9. Use Case Report – Request of Use*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-5 | |
| Use Case Name | Create Election | |
| Actor | Client Admin | |
| Purpose | Allows users to create an election | |
| Brief Description | This is where the client admin will create an election and add the positions that candidates are being elected. | |
| Trigger | The user wants to create an election | |
| Pre-Condition/s | • The user must be a client admin | |
| Post-Condition | • Election created successfully | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click “Create Election” button from homepage | | 2.The System will display Create election page |
| 3. User will add positions needed to be voted. | | 4.. The system will display the “Election created successfully” |
| 4.User will click “Save” button | |  |

*Table 10. Use Case Report – Create Election*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-6 | |
| Use Case Name | Cast Vote | |
| Actor | User | |
| Purpose | Allows users to vote in the election | |
| Brief Description | This is where the users will cast their votes for the candidate’s they want to vote for in the election | |
| Trigger | The user wants vote | |
| Pre-Condition/s | • The user must be given a unique passcode to be able to vote | |
| Post-Condition | • Votes are submitted and recorded | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click “Vote” button from homepage | | 2.The System will display voting page |
| 3. User will pick the candidate they want in Presidents. | | 4.. The system will display the “vote” prompt for the candidate selected in President. |
| 5.User will pick the candidate they want in Vice Presidents | | 6.. The system will display the “vote” prompt for the candidate selected in Vice President. |
| 7.Users will pick 6 candidates they want in Councilors. | | 8.. The system will display the “vote” prompt for the candidates selected in Councilors. |
| 9. User will click “Submit Votes” button | | 10. The system will record the vote. |

*Table 11. Use Case Report – Cast Vote*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-7 | |
| Use Case Name | View Results | |
| Actor | User | |
| Purpose | Allows users to vote see live results of the election | |
| Brief Description | This is where the students see the live results of the current election | |
| Trigger | The user wants to see the results | |
| Pre-Condition/s | • The user must already cast votes. | |
| Post-Condition | • Live results are displayed. | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click “See Live Results” button | | 2 The System will display the electoral results page |
| 3. User will pick from the “Colleges” page | | 4. The system will show the results of the college that was picked |

*Table 12. Use Case Report – View Results*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-8 | |
| Use Case Name | Apply for Candidacy | |
| Actor | Users | |
| Purpose | Allows users to apply for candidacy | |
| Brief Description | This is where the users can apply for candidacy in order to be elected in their preferred role. | |
| Trigger | The user wants to apply for candidacy | |
| Pre-Condition/s | • The user must be existing in the database | |
| Post-Condition | • The user will be in the lineup of official candidates for the election. | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click “Apply for Candidacy” button | | 2. The System will display the certificate of candidacy application page. |
| 3. User will fill up the certificate of candidacy form | | 3. The system will display confirmation for the application of candidacy. |
| 5. User will click the “Submit” button. | |  |
| Alternative Flow | | |
| Comelec Action | | System Response |
| 1. Client admin will click “Candidacy application” button | | 2. The System will display the candidate application list. |
| 3. Client Admin will view the application of the candidate | | 5. The system will prompt “Application approve or Rejected”. |
| 4. Client admin can approve or reject applications. | |  |

*Table 13. Use Case Report – Apply for Candidacy*

|  |  |  |
| --- | --- | --- |
| Use Case ID | UC-9 | |
| Use Case Name | Generate Reports | |
| Actor | Client Admin | |
| Purpose | Allows users to generate reports | |
| Brief Description | This is where the admin can generate statistical reports about the election | |
| Trigger | The user wants to see statistical report of the election | |
| Pre-Condition/s | • The voting period must ended. | |
| Post-Condition | • The statistical reports for the election are generated. | |
| Normal Flow | | |
| User Action | | System Response |
| 1. User will click “Generate Reports” button. | | 2. The System will display the voting results. |
| 3. User will click “Generate Report” button. | | 4. The system will generate a downloadable statistical report file. |

*Table 14. Use Case Report – Generate Reports*

* 1. Design Specifications
     1. Activity Diagram

REGISTER

A diagram of a flowchart

Description automatically generated

*Figure 11. Activity Diagram - Register*

**LOGIN**

A diagram of a software

Description automatically generated

*Figure 12. Activity Diagram - Login*

VOTER’S FEED

**A diagram of a flowchart

Description automatically generated**

*Figure 13. Activity Diagram – Voter’s Feed*

REQUEST OF USE

**A diagram of a work flow

Description automatically generated**

*Figure 14. Activity Diagram – Request of Use*

CREATE ELECTION

A diagram of a political election

Description automatically generated with medium confidence

*Figure 15. Activity Diagram – Create Election*

CAST VOTE

**A screenshot of a computer screen

Description automatically generated**

*Figure 16. Activity Diagram – Cast Vote*

VIEW RESULTS

**A diagram of a system

Description automatically generated**

*Figure 17. Activity Diagram – View Results*

APPLY FOR CANDIDACY

**A diagram of a process

Description automatically generated**

*Figure 18. Activity Diagram – Apply for Candidacy*

GENERATE REPORTS

**A diagram of a diagram

Description automatically generated**

*Figure 19. Activity Diagram – Generate Reports*

* + 1. A diagram of a company

       Description automatically generatedClass Diagram

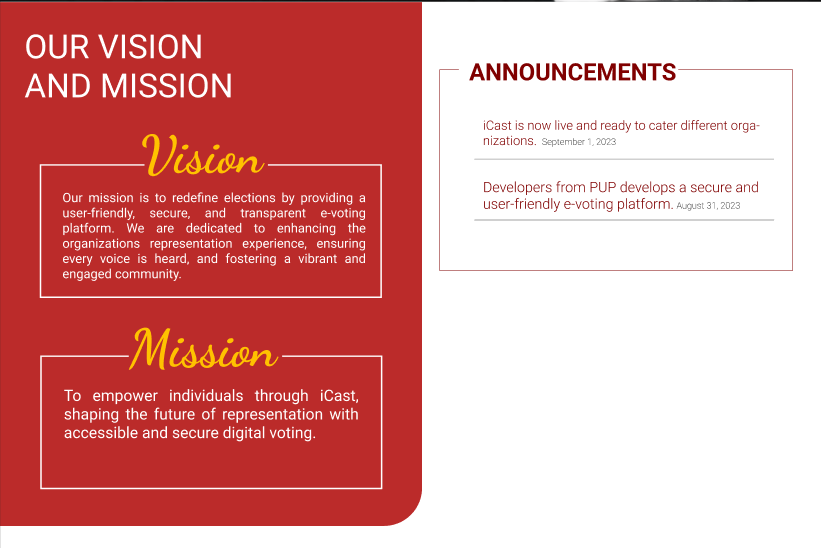
*Figure 20. Class Diagram*

* + 1. GUI Design

A screen shot of a voting site

Description automatically generated

*Figure 21. Landing Page*



*Figure 22. Vision and Mission/Announcement Page*

A close-up of a website

Description automatically generated

*Figure 23. About Us and Footer*

A red and white login screen

Description automatically generated

*Figure 24. Voter Login*

A screenshot of a login form

Description automatically generated

*Figure 25. Client Admin Login*

A screenshot of a computer

Description automatically generated

*Figure 26. System Admin Login*

A screen shot of a sign up form

Description automatically generated

*Figure 27. Client Admin Registration*

A screenshot of a computer

Description automatically generated

*Figure 28. System Admin Registration*

A screenshot of a video game

Description automatically generated

*Figure 29. View Partylists Page*

A screenshot of a web page

Description automatically generated

*Figure 30. View Candidates Page*

A screenshot of a computer

Description automatically generated

*Figure 31. Apply for Candidacy*

A screenshot of a computer

Description automatically generated

*Figure 32. Review of Application for Candidacy*

A screenshot of a voting screen

Description automatically generated

*Figure 33. Vote Page (Pre-Election Countdown)*

A screenshot of a voting screen

Description automatically generated

*Figure 34. Vote Page (Election Day)*

A screenshot of a video chat

Description automatically generated

*Figure 35. Voting Page*

A screenshot of a computer

Description automatically generated

*Figure 36. Review Votes*

A screenshot of a voting site

Description automatically generated

*Figure 37. Vote Notification and Redirection to Survey*

A screenshot of a survey

Description automatically generated

*Figure 38. iCast Survey Form*

A screenshot of a computer

Description automatically generated

*Figure 39. View Vote Information*

A screenshot of a voting form

Description automatically generated

*Figure 40. Voting Period Ended*

A screenshot of a computer

Description automatically generated

*Figure 41. View Results of Winners*

**Admin Login (Dashboard Panel)**

A screenshot of a computer

Description automatically generated

*Figure 42. Client Admin – Home Dashboard*

A screenshot of a computer

Description automatically generated

*Figure 43. Client Admin – View Total No. of Voters*

A screenshot of a voting box

Description automatically generated

*Figure 44. Client Admin – View Total No. of Candidates*

A screenshot of a computer

Description automatically generated

*Figure 45. Client Admin – View Partylists*

A screenshot of a computer

Description automatically generated

*Figure 46. Client Admin – View Elections*

A screenshot of a computer

Description automatically generated

*Figure 47. Client Admin – Manage Positions*

A screenshot of a computer

Description automatically generated

*Figure 48. Client Admin – Add Positions*

A screenshot of a computer

Description automatically generated

*Figure 49. Client Admin – Manage Voters*

A screenshot of a computer

Description automatically generated

*Figure 50. Client Admin – Add Voter*

A screenshot of a computer

Description automatically generated

*Figure 51. Client Admin – Upload CSV File (Importing User Database)*

A screenshot of a computer

Description automatically generated

*Figure 52. Client Admin – Manage Candidates*

A screenshot of a computer

Description automatically generated

*Figure 53. Client Admin – Customize Candidate Requirements*

A screenshot of a computer

Description automatically generated

*Figure 54. Client Admin – Add Fields for Requirements in Application for Candidacy*

A screenshot of a computer

Description automatically generated

*Figure 55. Client Admin – Manage Partylists*

A screenshot of a computer

Description automatically generated

*Figure 56. Client Admin – Add Partylists*

A screenshot of a survey

Description automatically generated

*Figure 57. Client Admin – Manage Survey Polls*

A screenshot of a computer

Description automatically generated

*Figure 58. Client Admin – Manage Survey Polls*

A screenshot of a computer

Description automatically generated

*Figure 59. Request for Use*

**System Admin**

A screenshot of a computer

Description automatically generated

*Figure 60. System Admin - Dashboard*

A screenshot of a computer

Description automatically generated

*Figure 60. System Admin – Manage Requests for Use*

A screenshot of a computer

Description automatically generated

*Figure 61. System Admin – Manage Client Admin*

* + 1. Database Schema

A diagram of a computer program

Description automatically generated with medium confidence

*Figure 62. Database Schema*

* + 1. Data Dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: UserTable Table Description: Contains information about the users | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| user\_id | int | No |  | Primary key | U-00001 |  |
| name | varchar(25) | No |  | Name of user | Jabez Victor Chavez |  |
| email\_add | varchar(25) | No |  | Email address of user | [jv@gmail.com](mailto:jv@gmail.com) |  |
| password | varchar(25) | No |  | Account password | \*\*\*\*\*\*\* |  |

*Table. Voter Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: VotesTable Table Description: Contains the details of the voting | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| vote\_id | int | No |  | Primary Key | VT-000001 |  |
| user\_id | int | No |  | Foreign key to user table | U-00001 |  |
| candidate\_id | int | No |  | Foreign key to candidate table | CND-00001 |  |
| position\_id | int | No |  | Foreign key to position table | POS-00001 |  |
| vote\_date | DATE |  |  | Date the vote was casted | yyyy-MM-dd |  |
| num\_of\_votes | int |  |  | Number of votes | 10000 |  |
| President\_votes | int |  |  | Holds the votes of the Presidents | 800 |  |
| Vp\_votes | int |  |  | Holds the votes of the VP's | 500 |  |
| Councilors\_votes | int |  |  | Holds the votes of the Councilors | 1000 |  |

*Table 15. Votes Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: PositionTable Table Description: Contains the details of the positions the candidates are running for | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| posistion\_id | int | No |  | Primary Key | POS-00001 |  |
| position-name | varchar(25) | No |  | The open position in the student council | President |  |
| description | varchar(100) | No |  | Describes the role or what it do | Leads the student council |  |

*Table 16. Position Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: CandidateTable Table Description: Contains the details of the candidates | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| candidate\_id | int | No |  | Primary Key | CND-00001 |  |
| party\_id | int |  |  | Foreign Key | PT-00001 |  |
| position\_id | int |  |  | Foreign Key | POS-00001 |  |
| surname | varchar(25) | No |  | Last Name | Chavez |  |
| first\_name | varchar(25) | No |  | First Name | Jabez Victor |  |
| middle\_name | varchar(25) | No |  | Middle name | Peñalosa |  |

*Table 17. Candidate Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: RequestofUseTable Table Description: Contains details of request from clients | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| request\_id | int | No |  | Primary key | R-00001 |  |
| name | int | No |  | Name of client | Edward Bane |  |
| details | varchar(255) | No |  | What is needed for the elections | Election will be 1 week with 7 positions |  |
| positions | varchar(30) | No |  | Positions needed in the election | President, External Vice president |  |

*Table 18. Request of Use Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: clientAdmin Table Table Description: Contains the details of the client admins | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| client\_id | int | No |  | Primary Key | COM-00001 |  |
| name | varchar(25) | No |  | Name of client admin admin | Sam Tang |  |
| email\_address | varchar(25) | No |  | Email address of client admin | [jv@gmail.com](mailto:jv@gmail.com) |  |

*Table 19. Comelec Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: partyTable Table Description: Contains information about the Party List | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| party\_id | int | No |  | Primary Key | PT-00001 |  |
| client\_id | int | No |  | Foreign Key | COM-00001 |  |
| party\_name | varchar(25) | No |  | Name of party | SAMASA |  |
| members | Varchar(255) | No |  | Party members | John Doe |  |
| num\_of\_candidates | int | No |  | Number of Candidates | 50 |  |

*Table 20. Party Info Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: SystemAdmin Table Description: Contains information about the main admins of the system | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| Sysadmin\_id | int | No |  | Primary key | PTM-00001 |  |
| request\_id | int | No |  | Foreign key | PT-00001 |  |
| client\_id | int | No |  | Foreign key | Benjamin |  |
| name | varchar(25) | No |  | Name of System Admin | Victor Chavez |  |
| email\_address | varchar(25) | No |  | Email address of the admin | jv@yahoo.com |  |

*Table 21. Party Members Table*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table Name: votersfeedTable Table Description: Contains information about the Party List | | | | | |
|  |
| Column | Type | Null | Default | Description | Sample Data |  |
| post\_id | int | NO |  | Primary key | PS-00001 |  |
| comelec\_id | int | NO |  | Foreign key | COM-00001 |  |
| post\_created | DATE |  |  | Date post created | yyyy-MM-dd |  |
| post\_updated | DATE |  |  | Date post updated | yyyy-MM-dd |  |
| image | int |  |  | included image on announcement | update.png |  |
| message | varchar(255) | NO |  | body of the announcement | Voting now open! |  |

*Table 22. Voter’s Feed Table*

* 1. Development Methodology
     1. **A screen shot of a computer

        Description automatically generated**Process Model

*Figure 62. Process Model*

The waterfall methodology is well-suited for this capstone project due to its linear and sequential approach, which aligns with the systematic implementation of an e-voting system.

The team chose the Waterfall method as the developmental research methodology in this system. This means that the developmental process will be one directional which means that there will be no adapting to the client’s sudden requests.

* + 1. Development Tools
* Figma

Figma is a collaborative web-based design tool that allows teams to create, share, and comment on designs in real time. It is a powerful tool that can help teams of all sizes create great designs. It is easy to learn and use, and it is constantly being updated with new features.

* MySQL

MySQL is an open-source relational database management system (RDBMS). An RDBMS is a software application that allows users to create, update, and manage data in a database. MySQL is one of the most popular RDBMSs in the world, and it is used by a wide variety of websites, applications, and organizations.

* Python

Python is a general-purpose, high-level programming language. It is known for its simple, easy-to-read syntax and its powerful libraries. Python is often used for web development, data science, and machine learning.

* PHP

PHP is chosen for iCast's backend because it excels in server-side scripting, seamlessly integrates with web servers, offers robust frameworks for development, supports various databases, provides security features, benefits from a strong developer community, and is cost-effective.ad

* HTML

HTML stands for HyperText Markup Language. It is a markup language used to create web pages. HTML is a text-based language that uses tags to define the structure and content of a web page.

* CSS

CSS stands for Cascading Style Sheets. It is a style sheet language used to control the presentation of web pages. CSS is used to control the appearance of HTML elements, such as their color, font, size, and layout.

* JavaScript

JavaScript is a scripting language that is used to create interactive web pages. It can be used to add dynamic content, animations, and user interaction to web pages. JavaScript is also used in a variety of other applications, such as mobile apps, game development, and machine learning.

* 1. Test Methodology/Procedures

The last step in the software developmental process is the testing phase. In this phase, the system is tested in order to ensure that all the criteria of the client are met and supports the user scenarios. This will also serve as a verification that the proposed solution is effective for the users.

### 3.3.1 Testing Plan

* Unit Testing- This test will be conducted in order to prove the system’s functionality and reliability.
* Blackbox Testing – The tester will focus on how the system meets the requirements proposed in the program specifications.
* System Testing – This test will ensure that the system meets the determined requirements and functionality.
* Usability Testing – This test evaluates the user interface for intuitiveness and ease of use.
* Acceptance Testing – This test will be conducted in order to gauge how the users will accept the system and assess if there will be issues raised.
* Beta Testing – The test is primarily done by users not part of the organization. It is a test that is performed by the end-users of the system.

### 3.3.2 Scope of Testing

The team will ensure that the system requirements are understood and conduct a series of handling tests for each process and maintenance.

### 3.3.3 Testing Procedures

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID |  | User |  |
| Description |  | | |
| Expected Result |  | | |
| Test Case ID | Test Steps | Expected result | Outcome (Pass or Fail) |
|  |  |  |  |

* 1. System Requirements

### 3.4.1 Hardware Requirements

|  |  |  |
| --- | --- | --- |
| System Unit | | |
| Hardware Components | Recommended Specifications | Minimum Specifications |
| Processor | Intel® Core™ i5-1235UL Processor (12mb cache, 4.40GHz) | Intel® Core™ i5-8400H Processor (8mb cache, 4.20GHz |
| Installed memory (RAM) | 16 GB | 8GB |
| Storage | 1TB | 256GB |
| Internet Speed | 25mbps | 10mbps |

*Table 23. Hardware Requirements – System Unit*

### 3.4.2 Software Requirements

|  |  |  |
| --- | --- | --- |
| Mobile Device | | |
| Hardware Components | Recommended Specifications | Minimum Specifications |
| Processor | Qualcomm Snapdragon 8 Gen 2(8mb cache, 2GHz) | Qualcomm Snapdragon 782G (4mb cache, 2.7GHz) |
| Installed memory (RAM) | 8GB | 4GB |
| Storage | 256Gb | 64GB |
| Internet Speed | 25mbps | 10mbps |

*Table 24. Hardware Requirements – Mobile Device*

* 1. Quality Plan

The quality plan for the iCast will evaluate the activities and processes of the system. This will ensure that the system complies with the software quality standards, meet the client's expectations, and fulfill the needs of the university.

## Criteria for Evaluation

To test and ensure the quality of the system, the proponents will follow the ISO/IEC 25010 standards, which is an international standard that provides a detailed description and evaluation model for the quality of software products.

## ISO/IEC 25010 Software Product Quality Standards

|  |  |
| --- | --- |
| **Software Quality** | **Description** |
| **Functional Suitability**   * Suitability * Accuracy * Security | The system's administrator will ensure that its features adequately cater to users' activities while maintaining compliance with organizational norms and procedures. For a better user experience, some system parts will also be able to communicate with one another. |
| **Reliability**   * Maturity * Fault Tolerance * Recoverability | A real-time voting system defines its capacity to reliably and precisely process and record votes in a timely way, ensuring that the outcomes are reliable and accurate representations of the voters' intentions. |
| **Usability**   * Understandability * Learnability * Operability * Attractiveness | To make the application simple to use, simple to learn and understand, and appealing to the users' eyes, the system will be designed with user interface design standards in mind. |
| **Performance Efficiency**   * Time Behavior * Resource Utilization | The system's administrator will make sure that it works effectively on a variety of gadgets and doesn't need high-end hardware because it only needs to function on browsers. |
| **Maintainability**   * Analyzability * Changeability * Stability * Testability | To make it simpler to read, identify, and correct errors in the code, the proponents will be sure to document it thoroughly. |
| **Portability**   * Adaptability * Installability * Replaceability | The system will be created as a web-based application that is responsive and can be used on a variety of devices, including PCs, laptops, tablets, and mobile phones as long as they have a web browser. |

*Table 25. ISO 25010 Product Quality Standards*

## Tools and Instruments

To assess the system’s quality and its compliance with the ISO/IEC 25010 software product standards and its quality model with six internal and external quality characteristics, the proponents will conduct the following activities:

## • User Acceptance Test

A user acceptance test will be given to Faculty and COSOA of PUP to evaluate the quality of the system based on the software product standards of ISO/IEC 25010.

## • Test Case

The proponents will prepare test cases which is a step-by-step procedure to test each feature of the system. Each test cases will have a “Passed” or “Failed” mark to evaluate every feature of the system.

## • Code Review

The proponents will review the system’s code to ensure that it follows coding best practices and industry standards, maximizes code efficiency, and is fully functional and free of errors.

## • Document Review

The proponents will ensure that the documentation of the system is clear, concise, and easy to understand. The document will be analyzed to make sure that it is complete and is consistent with the system.

* 1. Evaluation Plan

After the development and testing, the system will be evaluated by the Faculty and other different users.

### 3.6.1 Evaluation Schedule

On or before November the system should be available for testing. The following activities below will be observed during those weeks:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Objective** | **Timeline** | **Evaluation Measure** |
| Testing and Evaluation | The users must evaluate the system based on the ISO 25010 standard | August 2023 | The system will be ready for testing and evaluation. |
| Data Collection | After the evaluation of users, the team must gather the data based on the results | August 2023 | The data collected will be tallied accordingly |
| Data  Interpretation | The team must analyze and interpret the calculated results | August 2023 | The Likert scale will be used to interpret the average results |

*Table 26. Evaluation Schedule*

#### **3.6.2 Evaluation Procedure**

There will be two significant evaluation phases for the system. User Acceptance Testing will be employed in the initial phase, and the ISO 25010 standard will be used to assess its quality. A measurement method will also be used: the evaluation rating scale. The developers' test scripts will be used to evaluate it in the second phase.

#### **3.6.3 Users for the Evaluation (of the system)**

|  |  |
| --- | --- |
| The Commission on Student Organizations and Accreditation (COSOA)'s | The COSOA is the most important evaluator because they will be the primary users of the system. The Faculty are chosen to lead this evaluation. |
| Faculty | The team will randomly select from the faculty members of  CCIS for quality testing |

*Table 27. Users for the Evaluation*

#### **3.6.4 Statistical Treatment**

|  |  |  |
| --- | --- | --- |
| **Scale** | **Range** | **Verbal Interpretation** |
| 5 | 1.0-1.80 | Strongly Agree |
| 4 | 1.81-2.60 | Agree |
| 3 | 2.61-3.40 | Neutral |
| 2 | 3.41-4.20 | Disagree |
| 1 | 4.20-5.00 | Strongly Disagree |

*Table 28. Likert Scale* To compute the mean value:

Σ𝑥

x̅ =

𝑁

WHERE:

x̅ = mean Σ = sum of

X = data points

N = total number of respondents

#### **3.6.5 Survey Questionnaire**

The survey questionnaire below follows the ISO 25010 evaluation standard

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Direction:** Please put a check on the criteria of your choice.  **5 – Strongly Agree, 4** **– Agree,** **3** **– Neutral, 2 – Disagree, 1 – Strongly Disagree** | | | |  |  |
| **Functional Suitability** | **5** | **4** | **3** | **2** | **1** |
| 1. Was it easy for you to find and access the ballot and voting options on iCast? |  |  |  |  |  |
| 2. Were there any features missing or confusing during the voting process? |  |  |  |  |  |
| 3. Did iCast provide a straightforward process for selecting your preferred candidates or choices? |  |  |  |  |  |
| 4. Did iCast accommodate different types of elections, ensuring you could vote according to your preferences? |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reliability** | **5** | **4** | **3** | **2** | **1** |
| 1. Did you experience any technical issues, such as system crashes or unexpected interruptions while using iCast? |  |  |  |  |  |
| 2. Were you confident that your vote was securely and reliably recorded without errors? |  |  |  |  |  |
| 3. Did you receive any notifications or support in case of system failures or disruptions during your voting session? |  |  |  |  |  |
| **Performance Efficiency** | **5** | **4** | **3** | **2** | **1** |
| 1. Did iCast respond quickly to your actions and requests while casting your vote? |  |  |  |  |  |
| 2. Were there any noticeable delays or sluggishness during the voting process? |  |  |  |  |  |
| 3. Did you face any difficulties due to slow loading times or unresponsiveness? |  |  |  |  |  |
| 4. How well did iCast handle the traffic during your voting period? |  |  |  |  |  |
| **Usability** | **5** | **4** | **3** | **2** | **1** |
| 1. Did you find iCast's user interface intuitive and easy to navigate? |  |  |  |  |  |
| 2. Were instructions and guidance provided to help you understand the voting process? |  |  |  |  |  |
| 3 Were there aspects of the user interface that you found confusing or unclear? |  |  |  |  |  |
| **Maintainability** | **5** | **4** | **3** | **2** | **1** |
| 1. Did you receive information about the maintenance schedule or planned downtimes for iCast? |  |  |  |  |  |
| 2 Were any issues or defects you encountered addressed promptly by the system administrators? |  |  |  |  |  |
| 3. Did you notice any updates or changes to iCast during your voting experience? |  |  |  |  |  |
| 4. Were you satisfied with the overall maintenance and support provided? |  |  |  |  |  |
| **Portability** | **5** | **4** | **3** | **2** | **1** |
| 1. Were you able to access iCast in your preferred language? |  |  |  |  |  |
| 2. Were there any limitations in terms of language support or device compatibility? |  |  |  |  |  |
| 3. Did iCast adapt to different devices and screen sizes to accommodate your voting preferences? |  |  |  |  |  |
| **Security** | **5** | **4** | **3** | **2** | **1** |
| 1. Did you feel that your vote was secure and confidential when using iCast? |  |  |  |  |  |
| 2. Were there any concerns about the security of your personal information or voting choices? |  |  |  |  |  |
| 3. Did iCast provide clear information about the security measures in place to protect your data? |  |  |  |  |  |
| 4. Were you confident that iCast protected against unauthorized access and tampering? |  |  |  |  |  |
| **Compatibility** | **5** | **4** | **3** | **2** | **1** |
| 1. Were you able to access and use iCast on your preferred device (e.g., computer, smartphone, tablet)? |  |  |  |  |  |
| 2. Did iCast work seamlessly with the web browser or operating system you were using? |  |  |  |  |  |
| 3. Did you encounter any issues related to device compatibility or software compatibility? |  |  |  |  |  |
| Comments and Suggestions | | | | |  |
| Signature over Printed Name Date | | | | |  |

# REFERENCES

Admin. (2022, September 5). *Utilization of UVOTE Student Electronic Voting System as voter education Initiative in Rinconada Secondary Schools - DAP*. DAP. https://coe-psp.dap.edu.ph/compendium-innovation/utilization-of-uvote-student-electronic-voting-system-as-voter-education-initiative-in-rinconada-secondary-schools/

Boado, C. J., Bonifacio, E. J., & Dagocdocan, K. (2019). *A Comparative Study Of Manual Voting System And E-votingsystem For Supreme Student Government Council Election Ofpaso De Blas National High School. Scribd. https://www.scribd.com/document/475537711/ A Comparative Study Of Manual Voting System And E-votingsystem For Supreme Student Government Council Election Ofpaso De Blas National High School*

Cahyo, H. N., & Suryawinata, M. (2022). Responsive Web-Based E-Voting information System. *Indonesian Journal of Innovation Studies*, *20*. https://doi.org/10.21070/ijins.v20i.739

Cañeda, M. P. A. (2019). *WEB-BASED ONLINE ELECTION SYSTEM FOR COLLEGE STUDENT ORGANIZATION USING BLOCKCHAIN TECHNOLOGY*. https://ejournals.ph/article.php?id=14079

Dewi, P. F., Badruzzaman, A., Misbahuddin, M. H., & Febriawan, M. R. (2022). Implementation of design thinking for web based E-Voting Student Organization in Nahdlatul Ulama University of Yogyakarta. *Telematika: Jurnal Informatika Telekomunikasi Komputasi Elektronika Dan Industri*, *19*(3), 385. <https://doi.org/10.31315/telematika.v19i3.7962>

Govindaraj, R., Kumaresan, P., & Sree Harshitha, K. (2020, February 1). *Online Voting System using Cloud*. IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/9077751?fbclid=IwAR2RBtYPUKNXKz9kvsscodcTCcb2HECxhFac3CRbb9anqQ6mFtH0g34ECWY>

Ilakkiyavani, I., & Amsaveni, P. (2018, April 8). *A survey on web based application of secure online voting system*. International Journal of Trend in Scientific Research and Development. <https://www.ijtsrd.com/engineering/computer-engineering/11483/a-survey-on-web-based-application-of-secure-online-voting-system/shanthi-s>

Ilakkiyavani, R., & Shanthi, S. (2018). Web Based Application For Secure Online Voting System. *| IJCSET(www.ijcset.net)*, *Volume 8*(Issue 2), 23–27. https://ijcset.net/docs/Volumes/volume8issue2/ijcset2018080205.pdf

Institute of Bachelors in Information Technology Studies (IBITS). (2021). iVOTE++ Voting Application and Survey System. *Polytechnic University of the Philippines*.

Lai, W.-J., Hsieh, Y.-C., Hsueh, C.-W., & Wu, J.-L. (2018, August 1). *DATE: a decentralized, anonymous, and transparent e-voting system*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/8605994

Matos, Z., Jr. (2023). The Implementation of an Electronic Voting System for Supreme Student Council in the State College. *International Journal of Multidisciplinary Research and Publications*, *Volume 5*(Issue 8), 62–66.

Obeidallah, R., & Alzamel, R. (2019). An online e-voting prototype for student’s union elections in Jordanian universities. *Electronic Government, an International Journal*, *13*(2), 101. https://doi.org/10.1504/eg.2017.086035

Pereyras, J. (2019). *A computerized touchscreen student voting system for the universities and colleges*. https://papers.ssrn.com/sol3/papers.cfm?abstract\_id=3580542

Shankar, A., Pandiaraja, P., Sumathi, K., Stephan, T., & Sharma, P. (2020). Privacy preserving E-voting cloud system based on ID based encryption. *Peer-to-peer Networking and Applications*, *14*(4), 2399–2409. https://doi.org/10.1007/s12083-020-00977-4

Ushashree, Gudimani, K. S. G., & Patil, P. (2021, October 1). *Authenticated web based voting system*. IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/9587712