**ST PAULS UNIVERSITY**

**LIMURU CAMPUS**

**REGULAR**

**BCS 3106 SOFTWARE ENGINEERING**

**AMOS KILUMA BCISLMR167323**

**STACY MWEREMA BCISLMR142223**

**POWEL OTIENO BCISLMR168723**

**FAITH ONTIERI BCISLMR141423**

**LEC CECILIA NANFULA**

**20/2/2024**

**INTRODUCTION**

In the ever-evolving landscape of technology, the way we exercise our democratic right to vote is ripe for modernization. While traditional paper-based balloting has served its purpose, it presents limitations in accessibility, efficiency, and security. This project aims at developing a program that will be used by the university Council during student campus nominations for the yearly elections. The project also aims at Increasing accessibility which will enable secure voting from anywhere with an internet connection, promoting broader participation it also aims at enhancing efficiency which Streamlines the voting process, reducing wait times and human error. Lastly, the project aims at creating a user interface that designs an intuitive and accessible interface for diverse voters.

**PROJECT PLAN**

The aim of this project is to develop a program that will be used by the university Council during student campus nominations for the yearly elections. In the following project we will be trying to achieve and develop a program that will ensure that each student only casts one vote while maintaining the principle of secret ballot and only and currently registered students are allowed to participate.

While carrying out this task we will look out for certain considerations that will help the project to be a success some of the considerations include tools and technologies such as what software to use and the equipment’s to be used that will be favorable to the project and also the end users. To break down the project plan we will use a Gantt chart management plan to help has break the task into finer details.

**GANTT CHART MANAGEMENT PLAN FOR AN ONLINE VOTING PLATFOM**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **S.NO** | **TASK NAME** | **FEBRUARY** | | | | | | | | | **MARCH** | | | | | |
|  |  | **1** | **4** | **8** | **12** | **16** | **20** | **24** | **28** | **4** | | **8** | **12** | **16** | **18** |
| **1.** | **Project definition/planning** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 1.1 | Agreement |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 1.2 | Project description |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 1.3 | Project objectives |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| **2.** | **Analysis** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 2.1 | Requirements |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 2.2 | Modeling |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| **3.** | **Design** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 3.1 | Design revision |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 3.2 | Software designing |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| **4.** | **Content build** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 4.1 | Creation |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 4.2 | Edition |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 4.3 | Approvals |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 4.4 | Publication |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| **5.** | **User experience** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 5.1 | Testing |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 5.2 | Bug fixes |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 5.3 | Insights |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| **6.** | **presentation** |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 6.1 | Progress indication |  |  |  |  |  |  |  |  |  | |  |  |  |  |
| 6.2 | User training |  |  |  |  |  |  |  |  |  | |  |  |  |  |

**SOFTWARE REQUIREMENTS SPECIFICATION (SRP)**

A Software Requirements Specification (SRS) is a detailed document that lays out the functionality, behavior, and characteristics of a software system that is to be developed. It serves as a blueprint for the development team, helping them understand what needs to be built and how it should work.

1. *Introduction*

* Project Name: Secure Online Voting System
* Client: university institution
* Purpose: This system will enable eligible voters to cast their ballots securely and electronically in elections.
* Target Users: on going session students

1. *System Functionality*

* Verify voter identity and eligibility.
* Maintain a secure and encrypted voter registry.
* Ensure ballot secrecy and prevent vote tampering.
* Record votes anonymously and securely, preventing any association between individual votes and voter identities.
* Prevent any unauthorized modification or deletion of recorded votes.
* Allow only the resisted and the students that are in the ongoing session to cast votes.
* Provide voters with personalized and secure access to their assigned ballots.
* Implement end-to-end encryption for vote verification and auditability.

1. *System Requirements*

* The system shall comply with all resisted students and are in the ongoing session for online voting.
* The system shall ensure the authenticity and integrity of all votes cast.
* The system shall prevent unauthorized access, vote manipulation, and double voting.
* The system shall be resilient to unknown vulnerabilities and cyberattacks.
* Regular security audits and penetration testing shall be conducted.
* The system shall ensure authenticity and integrity of all votes cast.
* The system shall prevent unauthorized access and double voting.
* The system shall be transparent and allow independent verification of results election.
* The system shall be scalable to handle large number of voters during the election period.

**REQUIREMENTS TRACEABILITY MATRIX**

An RTM links requirements across different levels of detail, ensuring they are all addressed during development and testing. Requirement traceability matrix (RTM) has a table that shows how requirements relate to other artifacts in a project, such as test cases, design documents, and code. It is used to track the progress of a project and to ensure that all requirements are met. This is the requirements traceability matrix for the project:

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement description** | **Use case** | **Test case** |
| Functional requirement 1 | University-authenticated voter registration | Voter Registration | Verifies student number and university affiliation |
| Functional requirement 2 | Secure multi-factor authentication | Voter Login, Vote Casting | Password requirement |
| Functional requirement 3 | Student association-specific ballots | Voter Access, Vote Casting | Assigns voter to relevant association ballot(s) based on eligibility |
| Functional requirement 4 | Ranked-choice voting option | Vote Casting | Allows voters to rank candidates in order of preference |
| Functional requirement 5 | Real-time election results (restricted access) | Election Results | Displays aggregated, anonymized results |
| Functional requirement 6 | Individual vote verification | Individual Vote Verification | Provides individual vote details upon request |
| Non functional requirement 1 | 99.9% uptime during voting periods | All use cases | Monitors system uptime and performs regular maintenance |
| Non functional requirement 2 | Two-second response time for user actions | Vote casting | Measures and optimizes system performance for responsiveness |
| Non functional requirement 3 | Data encryption and access control | All use case | Encrypts votes and sensitive data |

**CONCLUSION**

In order for a specific project to work, one needs to perform various tasks for the project to become a success. In the above project, various tasks are managed for the project to be a successful.

*References*

1. *National Institute of Standards and Technology (NIST) Election Testing Laboratory (ETL):* [*https://www.nist.gov/voting-systems*](https://www.nist.gov/voting-systems)
2. *Brennan Center for Justice – Democracy Reform:* [*https://www.brennancenter.org/*](https://www.brennancenter.org/)
3. *Center for Election Research (CER):* [*https://earc.berkeley.edu/*](https://earc.berkeley.edu/)
4. *Election Runner:* [*https://app.electionrunner.com/*](https://app.electionrunner.com/)
5. *Election Buddy:* [*https://electionbuddy.com/*](https://electionbuddy.com/)
6. *“What is an Online Voting System?” by eBallot:* [*https://electionbuddy.com/features/online-voting/*](https://electionbuddy.com/features/online-voting/)