Final Project

SAAD 1001

Ian Christian

W0480449

Due Date: April 6th, 2023

Contents

[1. Introduction 3](#_Toc131512291)

[2. Strengths, Weaknesses, Opportunities, Threats 4](#_Toc131512292)

[3. Risk Analysis 5](#_Toc131512293)

[4. Software Development Lifecycle & Project Planning 6](#_Toc131512294)

[4.1. Model Overview 6](#_Toc131512295)

[4.2. Project Requirements 7](#_Toc131512296)

[4.3. Product Design 8](#_Toc131512297)

[4.4. Product Development 9](#_Toc131512298)

[4.5. Testing & Revision 10](#_Toc131512299)

[4.6. Timeline 11](#_Toc131512300)

[5. UML Diagrams 13](#_Toc131512301)

[5.1. Structural Diagram 13](#_Toc131512302)

[5.1.1. Class Diagram 13](#_Toc131512303)

[5.2. Behavioural Diagrams 14](#_Toc131512304)

[5.2.1. Use Case Diagram 14](#_Toc131512305)

[5.2.2. Activity Diagram 15](#_Toc131512306)

[5.2.3. Sequence Diagram 16](#_Toc131512307)

[6. Test Case Scenario 17](#_Toc131512308)

[7. Conclusion 18](#_Toc131512309)

# Introduction

This document serves to give a complete overview of the development of a voting software product, as envisioned by ElectSys incorporated. The time scale from planning to release of this software product has been set at 4 months.

ElectSys inc. is a newly created IT start-up company which aims to provide best-in-class software products. While voting software has applications in a variety of industries, ElectSys primarily targets clients in the government/public sector.

The development team at ElectSys consists of two programming teams, each with approximately five members including a manager.

For the purposes of this document, it is assumed that ElectSys inc. is concerned solely with the development of the voter polling software itself. Hardware required to run the software, i.e. sever databases, on-site voting machines, etc. is to be considered external to the development process and can be assumed as already procured.

# Strengths, Weaknesses, Opportunities, Threats

The first task in the planning of a software development project is to justify the professional feasibility of the product. This can be demonstrated by way of a SWOT analysis. Below is a breakdown of the strengths, weaknesses, opportunities, and threats involved in ElectSys’ task of creating a voter polling program, inasmuch as they pertain to the feasibility of the business proposition.

|  |  |  |
| --- | --- | --- |
|  | **Helpful** | **Harmful** |
| **Internal Origin** | **Strengths**   * Small development team can work quickly and react to changes of requirements. * Team works remotely which reduces company operating costs. * ElectSys hires extremely qualified developers, ensuring code quality. | **Weaknesses**   * Small development team means higher level of responsibility for each employee. * Small development team is less able to design robust security measures. * Being a tech startup, ElectSys has limited resources to fund development. |
| **External Origin** | **Opportunities**   * Increasing demand for accurate and flexible electronic-based voting solutions. * Public demand for remote voting process to increase ease of use. | **Threats**   * Large companies already exist in this space, making for difficult competition. * Wide variety of possible requirements based on the given government client. |

# Risk Analysis

It is also valuable to the success of the project to determine some likely risks that may be involved. What follows is a table of several potential risks and their components, followed by risk statements which include mitigation strategies.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Risk Analysis** | | | |
|  | **Threat Actor** | **Asset** | **Effect** | **Method** |
| **Risk #1** | External hackers. | Personal voter data | Data may be manipulated without authorization, leaving it compromised. | Looking for system loopholes and exploiting unknown errors. |
| **Risk #2** | Internal Developers | Project timeline | Project goals are not met in a timely manner, leading to large-scale delays. | Developers work slower than expected or are given unfeasible deadlines. |
| **Risk #3** | Government investors/Clients | Project Budget | Operations may cease or staff are let go from the company. | If investors are unsatisfied with the progress of the product they may reduce or cancel funding. |

**Risk Statements**

**Risk #1:** The integrity of personal voter data is at vulnerable to manipulation by external hackers. If copies of the software are leaked publicly then bad actors may seek to find flaws in the software to exploit. This risk is moderately unlikely but has severe consequences if it occurred. As a mitigation strategy, team members should be instructed to never share the details of their work with outsiders.

**Risk #2:** The progress of the product schedule could be hampered by developers not meeting deadlines. This could occur as result of a number of factors, not least of which being the initial deadlines set for goals. This risk has moderate likelihood of occurring. As a mitigation strategy, team leaders and project managers should keep in regular contact with their subordinates to ensure they are managing workloads well.

**Risk #3:** If the clients or governments responsible for funding ElectSys’ operations are dissatisfied with how the product is developing, they may choose to halt said funding. This risk is not likely to occur but to mitigate that risk there should be regular meetings between the client and project leaders to ensure expectations are made clear and agreed upon as reasonable.

# Software Development Lifecycle & Project Planning

## Model Overview

The development methodology selected for this project will have a significant impact on the final product. Ideally, the chosen methodology should consider both the requirements of the scenario, and the circumstances of the teams involved.

The methodology recommended for ElectSys is to utilize a waterfall model. This is where each major phase of the development occurs sequentially and without reiteration (for example, the requirements are defined at the start of the project and not revisited.) The benefits of this model are that it is relatively straightforward and allows for clearly defined primary goals and milestones. This means that it is well suited for small teams which require minimal management.

The waterfall model is well suited for ElectSys, most importantly, because the requirements as established by the project goals are very clear and, being primarily aimed at government applications, unlikely to change quickly. This means that ElectSys does not have to allocate resources and plan for the possibility of a change in project requirements (whether they be internal or external as requested by a client.)

Because of the focused nature of the Waterfall model, having daily stand-up meetings and team check-ups is not as essential as in an Agile environment. Progress can be gauged well enough by team managers communicating directly with their subordinates. This is feasible because of the relatively small scale of the development teams. What will be useful however, is for all developers involved to attend a kickoff meeting at the beginning of the development phase. This meeting will serve to inform everyone about the overall goals and objectives, and to clarify the process of giving assignments out. During the development process there may also be team specific meetings at the start of each new feature assignment (for example, progressing from voter validation to the ballot casting feature.) These smaller team meetings will allow managers to publicly assign tasks to individual developers. This will keep everyone up to date on which person is working on what aspect of the current feature.

## Project Requirements

The requirements for the voter polling software are defined as follows:

1. The software must allow for the registration of new voters, as administered by a voting official.
2. The software must allow for the registration of voting officials. Only registered officials may have access to the higher functionality of the system.
3. The software must allow for the listing of multiple candidates and their relevant information.
4. The software must generate a unique keycode for each newly registered voter.
5. The software must allow for editing of personal information for voters, officials, and candidates. Only successfully logged-in officials may have permission to make edits.
   1. Candidate’s vote counts, as a security measure, are not permitted to be edited. These values are directly dependent on the candidate choice inputs of voters.
6. Only a user validated by their keycode, matched with their personal information may be permitted to cast a ballot.
7. The software must display a selection of candidates to the validated voter.
8. The software must reliably and accurately receive and data input from voters, indicating their choice of candidate.
9. The software must store said data and track the votes per candidate.
10. The software must provide limited access for voters, and unrestricted access for voting officials.
11. The software must not permit the editing of candidate counts by any individual, including voting officials.
12. Voting officials should have the ability to view candidate details including vote counts.
13. Voting officials have the option to allow the software to declare the winning candidate.
14. The software must have an intuitive user interface, to allow ease-of-use by voters and officials.
15. It is assumed that vote statistics, login details, personal information, and the like will be stored in a protected server accessed via the software.

## Product Design

Given the list of requirements, the design of the product may begin. This includes determining the choice of programming language, data structures and relationships, choice of graphical user interface frameworks, and user experience planning.

It is worth stating that ElectSys plans to create an entirely original voting software product to satisfy the outlined requirements. They will not be adapting or inheriting a previously developed product. Many voting systems around the world are still conducted through entirely physical means (pen and paper). Consequently, there are relatively few examples of software products to draw upon for this purpose. ElectSys aims to develop software that is more consistent, more intuitive, and more secure than old-fashioned manual methods.

The recommended programming language for this project is Java. Java is a robust and well-established language in the Object-Oriented style. OOP style languages are intuitive to work in because they operate on the concepts of blueprints (classes) and realisations of those blueprints (objects or instances). This is like the structure of human language and thought so it is a widely used approach in software design.

Java is also ideal because of its extensive security capabilities. All operations in a Java-based program occur in a virtual machine called a “sandbox”. This allows for restrictions on what sort of interactions may occur between the program contents and external actors.

A number of graphical user interface frameworks exist for Java. The framework recommended for this project is JavaFX because of its feature-rich toolset. It is possible to create highly sophisticated, and aesthetically pleasing applications using JavaFX.

## Product Development

At this stage the development team may begin work on creating the voting software itself. Each phase will focus on a specific feature and will follow a similar format.

1. A working prototype is developed quickly to achieve the feature functionality.
2. The prototype is refined to robustly achieve the feature design requirement.
3. Once fully developed the feature is integrated into the working build of the main software product.

Each phase also serves to further verify the success of the previously completed feature. If it is revealed that the previous feature does not meet its requirements, progress is reversed to address the issue. Progressing to a new phase is not allowed until the present feature is demonstrated to be adequate. In this way, the main software product develops sequentially, guaranteeing the validity of the final outcome.

During the development stage, implementation of software features will be divided between the two teams at ElectSys. The division of responsibilities are as follows.

|  |  |
| --- | --- |
| **Team A** | **Team B** |
| * Adding Voters, Candidates, and Officers | * Editing Voters, Candidates, and Officers |
| * GUI Design | * GUI Implementation |
| * Officer Login Verification | * Voter Keycode Validation |
| * Ballot Casting | * Ballot Counting and Statistics |

To prevent the teams progressing out of sequence with each other, only one new feature will be allowed in development at any given time. When team A, for example, is developing a new component, team B will either be performing unit testing and verification on the most recently completed feature, or they will be implementing revisions discovered by the testing process. The benefit of this workflow is that software validation and new development can always be occurring. This also allows the small development team at ElectSys to simultaneously perform the role of a quality assurance team, thus reducing company resource requirements.

## Testing & Revision

At this stage of the development process it is expected that the primary requirements of the project have been achieved and the voting software is in a state similar to the final product. This is when the focus will change from developing new features and functionality to testing the program as a whole. The ElectSys team will search extensively for bugs and unexpected behaviours, record them, and make code revisions as needed. Because unit testing is conducted throughout the development process it is expected that any bugs which may occur in the final testing phase are related to the integration of multiple components. Once the behaviour of the system is consistent enough that it meets the high standards at ElectSys, the product will be released.

## Timeline

To stand a chance at successfully delivering the finished voting software as a marketable product, it is wise to establish a timeline for the project milestones. Focusing on smaller goals within larger scope of the project will ensure that progress is made consistently and reliably. Below is an approximate timeline for the major milestones identified as essential to the completion of the project.

|  |  |  |
| --- | --- | --- |
| **Week #** | **Feature or Focus** | **Comments** |
| 1-4 | **Focus:** Project Planning and preliminary design | Here the ElectSys team will conduct industry research, assess requirements, and build mock-ups. |
| 5 – 6 | **Feature:** Adding Voters, Candidates, Officers, Generating Secure Keycodes | The focus of this period is on building the general structure of the datatypes. Implementation will be guided by OOP principles |
| 7 | **Feature:** Editing of Personal Information | This feature will allow for more flexible use of the software. It provides the ability to correct human errors in input. |
| 8 – 9 | **Focus:** Ballot casting process, Security Checks | This period will be devoted to building a functional voter input stream. Also addressed will be user identity validation. |
| 10 | **Features:** Statistics Review Panel, Winner Declaration | At this phase the focus will be on implementing various forms of statistical analysis on vote data, as well as program functionality for declaring the winning candidate. |
| 11 – 12 | **Focus:** GUI Refinement | During this period the teams will focus on making the user interface both aesthetically pleasing, and intuitively navigable. |
| 13 – 16 | **Focus:** Integration Testing | For the final phase of development, everyone at ElectSys will focus on testing the software to search for errant behaviour. |

# UML Diagrams

## Structural Diagram

### Class Diagram

## Behavioural Diagrams

### Diagram Description automatically generatedUse Case Diagram

### Activity Diagram

Diagram

Description automatically generated

### Graphical user interface Description automatically generatedSequence Diagram

# Test Case Scenario

A useful tool in defining the behaviour of a program is to include test scenarios. This is a methodical breakdown of a specific aspect or desired functionality of the system. Below is a table of several possible scenarios within the ElectSys voting software.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test #** | **Description** | **Test Process** | **Data** | **Expected Result** |
| 1 | Officer Initial Login | 1. Run program  2. Enter Root officer username  3. Enter Root officer password | Username = root  Password = 123 | Officer will successfully login to the Admin panel |
| 2 | Voter Validation | 1. Officer adds voter  2. Officer gives voter their keycode  3. In Voting Panel, voter enters their keycode correctly  4. Voter enters their name correctly | **Officer:**  New voter = Emily Smith  New keycode (randomly generated) = xi39fo9  **Voter:**  Keycode = xi39fo9  Name = Emily Smith | Voter is successfully validated and proceeds to vote selection page |
| 3 | Voter Validation | 1. Officer adds voter  2. Officer gives voter their keycode  3. In Voting Panel, voter enters the wrong keycode | **Officer:**  New Voter = Mark James  New keycode (randomly generated) = e034k41  **Voter:**  Keycode = 28fgi21 | The voter will be rejected when the provided keycode cannot be found |
| 4 | Winner Declaration | 1. All voters cast ballots successfully  2. Officer reviews candidate vote counts  3. Program selects candidate with highest vote count | **Candidate #1:** 1 vote  **Candidate #2:** 6 votes  **Candidate #3:** 3 votes | Candidate #3 will be declared the winner |
| 5 | Winner Declaration | 1. Officer reviews candidate vote counts  2. Program selects candidate with highest vote count | **Candidate #1:** 0 votes  **Candidate #2:** 0 votes  **Candidate #3:** 0 votes | No winner will be selected because all vote counts are the same |

# Conclusion

While ElectSys inc. may be a relatively small and new technology startup, they aim to achieve lofty goals by way of their highly professional standards, their insight into modern software design, and their commitment to satisfying the needs of their clients.

Having a clearly defined project plan is essential to the success of the venture. ElectSys hopes that with this preliminary document as a guide, they will be led on a path to a technically elegant and commercially viable product.