Table of Contents

1. Software Quality Assurance Plan

This document gives a brief amount of explanation of Software Quality Assurance Plan (SQAP) for transit electrification simulation software project. The methods, standards, practices, and metrics which are used for the quality assurance of the transit electrification simulation software project are included in this SQAP. In this project we must develop a simulation software to full electrification of STM route 211 in which software gives a pre-configured plan and charging schedule according to the user inputs.

1.1. Purpose

This Software Quality Assurance Plan (SQAP) gives information on the software quality assurance activities, procedures, and techniques to ensure that product meets the conformance to specification which are stated by the user of the system.

- This SQAP identifies the tools which is used in developing of the simulation software.
- Microsoft Word is used for the requirements gathering phase to develop software requirement specification (SRS) document.
- Microsoft Paint is used for design phase to obtain a rough sketch of the UI design.
- It identifies the responsibilities of the SQA Planner and Developer.
- After completion of the product SQA reviews and audits will be conducted.
- It gives information on which product and process are going to be reviewed by the SQA member.

1.2. Reference Documents

- IEEE Standard for Software Quality Assurance Plan, IEEE STD-730-2002
- Lecture Slides (6260-W-2-SQA and 6260-W-6-SQAP and Statistical SQA) by Dr. CHUN WANG, Winter - 2020
- Roger S. Pressman, Software Engineering: A Practitioner's Approach, McGraw - Hill, 2004, ISBN - 10: 007301933X
- Software Requirement Specification Document (SRS V1.0 and SRS V1.1).

1.3. Management

1.3.1. Organization

This project is consisting of two individuals for developing the product and documentations. The first individual is responsible for the developing the product and review the usability, reliability, and efficiency. The second individual is responsible for the all the documentation need to be carried out. However, the professor will timely review and audit the product and documents developed. The whole class will act as an organization for reviewing the work products and document deliverables in which the professor will take decisions on development of the project.

1.3.2. Tasks

The tasks of the developer are as follows:

- Developing the simulation software design by transforming the requirements
- Development of the product
- Testing the application
- Deliver the application in the form of product

The tasks of the documenter are as follows:

- Gather the requirements and develop the software specification document (SRS)
- Develop the schedule and plan for the project
- Develop the present the product to the organization after requirement gathering, analysis, design, implementation, and testing phases
- Develop the SQA Plan and Software Design Specification (SDS)

The tasks of the organization are as follows:

- To conduct review and audit of the product and processes
- After conducting review gives the feedback.

1.3.3. Roles and Responsibilities

Team Members	Roles	Responsibilities
Dhaval Sharma	Programmer	Develop the product and testing each module.
Himal Patel	Documentations and SQAP	Develop the documents needed throughout the software development life cycle (SDLC).
Dr. Chun Wang	Organization and Project Manager	Conduct the review and audit including feedback and advice.

1.4. Documentation

1.4.1. Purpose

There are specific documents which are given after completing a phase in software development life cycle (SDLC) through which we can keep the track of activities. The list of the document are as follows:

- Software requirement specification (SRS) used for requirements gathering and analysis is performed on the user requirements.
- During planning phase software development plan (Dev Plan) is created and defines scheduling of the whole project.
- After the planning phase software design specification (SDS) the design document will be developed.
- The product will be developed after the preliminary design of the software has been created and various testing is done on the product for final deployment.
- The product will be delivered including the User Manual describing how the software will be used.

The list of the documents to be reviewed are as follows:

- Software requirement specification (SRS) will be audited by software specification review (SSR).
- Software development plan (Dev Plan) will be reviewed during verification and validation plan.

- The software design specification (SDS) will be reviewed through detailed design review (DDR) and architectural design review (ADR).
- Software quality assurance plan (SQAP) can be reviewed by the Managerial reviews.
- User Manual reviewing can be done in user documentation review (UDR).

1.4.2. Minimum Documentation Requirements

To develop a software following documents will be include:

1.4.2.1. Software Requirements Description (SRD)

- It describes the requirements which are essential like functions, non – functional, design and conceptual model of the system.
- Through this document we can track that the product which are going develop is according to conformance of the requirements.

1.4.2.2. Software Design Description (SDD)

- This document gives an architectural overview how the User Interface (UI) of the system will look.
- It gives the detailed design in which the components of the system design like how the system will communicate with the database, server and other software packages.
- It gives the overall structure of the simulation software.

1.4.2.3. Verification and Validation Plans

- Describes the test plan that will be used to test the product and processes and produce some results.
- Analysis of the user requirements will be done and as a result check the feasibility of the project.
- It evaluates the design of the software and check that conforms to the specification mentioned in software requirement description document.
- The reviews and audits of the documents, products and processes which are taken mentioned in the plan.
- The unit and integration testing will be done for the developed product.

1.4.2.4. Verification results report and validation results report

- The verification activities result which is taken according to the verification plan should be written in the verification results report that the software is developed with conformance of requirements.
- The validation activities result which is taken according to the validation plan should be written in the validation results report that the software is developed by considering development standards.

1.4.2.5. User documentation

- This document contains the brief description that how to operate the software by giving various examples of inputs and outputs.
- It also gives idea about the solution if any unknown error encountered to the user and what the error means.
- It also specifies the requirements to use the software.

1.4.2.6. Software configuration management plan (SCMP)

- The document contains the tasks performed by various member of the team like developing of the software and documentation after every phase of the software development life cycle (SDLC).
- The resources needed like physical space and different software used to development of the product.
- Different versions of software modules are developed and keep track of each module completion that each module will be tested and after combine the modules into a whole software.
- After completion of each module developer will report and check that it is developed as per the standard.
- The format of files in which the product will be delivered.

1.4.3. Other documentation

There are no other documents which are used to develop the software product and software processes.

1.5. Standards, practices, conventions, and metrics

1.5.1. Purpose

It identifies the standards, procedures, and techniques to be used including metrics to be applied. The development of the simulation software should be according to the standards. In this project, document standard, design standards, coding standards and process metrics is applied. The process metrics used in this project is KLOC (Thousands Line of Code) to measure the line of codes. By doing timely review of each standard the conformance of developing the simulation software is assured. However, implementation of error density and defect density is done for these items by which quality is monitored.

1.5.2. Content

In this section the various document standards are defined for development of the product according to below mention standards. The quality assurance member will check through this list that product is developed according to the standards.

- Documentation Standards – All the documents for the project are developed according to the IEEE Standard Guidelines.
- Design Standards – The design standards for the developing the software design description of the software is IEEE STD 1016.
- Coding Standards – The standard used for coding is jQuery JavaScript Style Guide and Python FLASK APP.
- Quality Assurance and Process Metrics Standard – Standard for the Software Quality Metrics Methodology, IEEE STD 1061.

1.6. Software reviews

1.6.1. Purpose

Software reviews are crucial part of any development of the software and it gives the opportunities to find errors in whole software development life cycle at every phase. Senior management team will be responsible for carrying out the reviews and audit.

In above mentioned documentation section 1.4.1, the specific reviews which need to be conducted for completion of the section are as follows:

- Software Requirement Description Review (SRDR)

- Development Plan Review (DPR)
- Detailed Design Review (DDR)
- Formal Technical Review by the whole organization (FTR)

The schedule given for each review which will be conducted is as follows:

Review Documents	**Date**
Software Requirement Specification Review (SRSR)	29/01/2020
Development Plan Review (DPR)	11/02/2020
Detailed Design Review (DDR)	11/03/2020
Product Review	07/04/2020

1.6.2. Minimum Requirements

1.6.2.1. Software Specification Review (SSR)

The requirements in the stated by the user will be written in the software requirement specification (SRS). The review of the document is conducted to assure adequacy of the requirements prior to the original peer review. The review leader and reviewer will be present at the time of the review and will audit the specification document. If they find any inadequacy in the requirements the customer stated will take note of that and ask for revised document.

1.6.2.2. Architecture Design Review (ADR)

The high – level preliminary design of the software in software design specifications documents. The review of the document is conducted to assure adequacy of the preliminary design and find flaws in the design. The overall structure of the software will be reviewed by the review leader and reviewer. The inadequacy in the design according to the software requirement specification results into changing of design.

1.6.2.3. Detailed Design Review (DDR)

The detailed design of the software includes how the system work and what the user will view. The review will give a brief idea of the detailed design is conformed to the user specification and the quality of the design and further improvements in the design. The design detail of the software will be reviewed by review leader and reviewer.

1.6.2.4. Verification and Validation Plan Review

The implementation of Verification and Validation plan reviews will be done during the designing, coding, and testing reviews. The test plan for the unit and integration testing of the whole software will be evaluated to ensure the adequacy of the verification and validation methods and correct according to the software requirements. The verification and validation plan of the software will be reviewed by review leader and reviewer.

1.6.2.5. Functional Audit

In this audit the verification of the whole software takes place in which the functionalities mentioned in the software requirement specification are fulfilled without any errors and defects. It's an useful approach to achieve the user expectations and ready to deliver a high quality product.

1.6.2.6. Physical Audit

The verification of the internal consistency inside the various modules of the system and gives the appropriate pre – configured plan results according to the user inputs in the control panel. However, it is held before the final release of the software and its related documents and also check the consistency between the documents like software requirement specification and software design specification.

1.6.2.7. In – process Audits

It is the verification of the simulation software design with respect to development of the software in which the requirements of the user will is fulfilled. The verification of the developed software will work on the user specified system consists of hardware and software.

1.6.2.8. Managerial Reviews

Senior management will conduct the reviews after each phase of the software development life cycle (SDLC). The management asks for the monthly progress report and weekly checking of the work done by the developers. It establishes understanding of the technical issues and define the project risks.

The results of the quality assurance review and audit will be written inside a report and given to the upper management and head. The members of the quality assurance team prepare the report and given to the project manager for monitoring and evaluations. However, the monthly reviews performed, and list of the activities carried out in previous month are compared to find efficiency of the working team and observe the issues to give feedback.

1.6.2.9. Software Configuration Management Plan Review (SCMPR)

This plan review consists of the intermediate or prototype product which is developed and after completing the final product the deliverables are met to the requirements. This review checks the product delivered is according to the agreement and verify the completeness of the configuration management plan methods.

1.6.2.10. Post – implementation Review

The outcomes of the implementation in which the mistakes found can be used as lessons for the next project.

1.6.3. Other Reviews and Audits

1.6.3.1. Software Developer Plan Review

The details about the scope, objective, project schedule, project resources, risk management plans, project monitoring and controlling which are carried out according to the plan including the document preparation according to the standard.

1.7. Test

During Post Execution various tests were performed as follows:

- **Test Cases:** In the form of testing views and approaches Exhaustive and Selective testing will be implemented. There are other types of testing that can also be implemented such as white box testing and black-box testing.
- **Unit Testing:** This testing will ensure that each performs the designated function.
- **Integration Testing:** A systematic technique can also be conducted to construct the software architecture and conduct tests to uncover the errors associated with interfacing. Bottom-up and top -down are the integration types used in our project.
- **Smoke Testing:** It is also known as “Build Verification Testing”, It comprises a non-exhaustive set of tests that aim at ensuring that the most important functions work. This testing is used to decide if a build is stable enough to proceed with further testing.
- **Validation Testing:** This testing in our project focus on software requirements, user-visible actions and output from the system Validation succeeds when

software performs functions in a manner that can be reasonably expected by the customer (SRS).

- **System Testing:** This testing is conducted on a complete integrated system to evaluate the system's compliance with its specified requirements. By taking its input, all the integrated components that have passed integration testing.
- **Debugging:** The process is also called diagnostic process which identifies and removes errors from computer hardware or software in per project.

There is no separate test plan that exists which is taken into consideration taken in for reference.

1.8. Problem reporting and corrective action

The reporting, tracking, and resolving the problems after the final product is delivered and the user encountered an unknown error in which the update is required. The system will generate a log of all encountered errors and keep on the administrator's server.

The respective maintenance person of the organization will be responsible for checking the log and upload the updated code files to the server.

1.9. Tools, techniques, and methodologies

Following are the identified software tools, techniques, and methods used to support SQA processes:

- Visual Studio Code
- Auditing
- Reviewing
- Python
- JavaScript
- HTML
- CSS
- Virtual Server (FLASK APP)
- Internet Browser

1.10. Media Control

In media control, we are providing a link to access all related documentation and guide to the project to all. However, the guide will be obligated to read only. No write access will be provided.

1.11. Supplier Control

The methods to be employed to assure that the suppliers comply with the requirements of this standard are not applicable. The software is not to be developed under contract, so there are no other procedures for contract review and update which are requires to be described.

1.12. Records collection, maintenance, and retention

Records Collection: Our project does not require any record collection.

Maintenance: The timely maintenance of the system will be carried out to ensure that system runs smoothly and efficiently without generating false outputs.

Retention: The software will be developed in parts as we can say divide and conquer which is the easiest way to approach a high – quality software with minimum errors.

1.13. Training

Following are the identifying training activities necessary to meet the needs of the SQAP:

1) Creation of SQA Management Plane: The SQA Management Plan is created to carry out the foremost activities including laying down a proper plan regarding how the SQA will be carried out in your project.

2) Setting the Checkpoints

3) Apply Software Engineering Techniques

4) Executing Formal Technical Reviews

5) Evaluating Product: This activity confirms that the software product is meeting the requirements which were discovered in the project management plan. It ensures that the standards set for our project are followed in a correct and sequential manner.

6) Process Monitoring: This activity verifies if the correct steps were taken during software development. This is done by matching the taken steps against the documented steps.

7) Performing SQA Audits: The SQA audit inspects the entire actual SDLC process followed by comparing it against the established process.

It also checks whatever reported by the team in the status reports were performed or not. This activity also exposes any non-compliance issues.

8) Maintaining product records and reports: It is vital to share the required SQA information with the stakeholders in the form of reports and records, thus, it is important to keep the necessary documentation related to SQA. For future reference, the test results, audit results, review reports, change requests documentation, etc. should be kept in the form of record.

1.14. Risk management

There are adequate dangers that can be experienced consists of list improvement holes and for instance, absence of adequate information, experience and programming abilities required by the project including absence of assets, instruments, and time.

Insufficient knowledge and experience: The knowledge of the how the STM works and absence of information about Route 211. However, by using various training components we are trying to improve the knowledge on the STM.

Insufficient resources, tools, and time: We can have low – configuration machines to run this application and development time constraint taken into consideration. The low – performance servers and slow internet connection can be a problem.

The software sometimes not produce the reliable result like wrong charging schedule and inadequate pre-configured plan.

To control and monitor this risks Software Quality Assurance Plan (SQAP) is developed.

1.15. Glossary

SQAP- Software Quality Assurance Plan

SRS- Software Requirement Specification

STM - Société de transport de Montréal. The city transport of Montreal which consists of metros and buses.

Route 211 – The buses runs between Terminus Lionel-Groulx station and Terminus MacDonald is on Route 211.

GUI – Graphical User interface (GUI) is an interface through which the user interacts visually.

SDS - System Design Specification
SQA - Software Quality Assurance

ADR – Architectural Design Review

DDR – Detailed Design Review

SDR – Software Description Review

UDR – User Document Review

1.16. SQAP change procedure and history

SQAP Version 1.0 (Date: 01-03-2020)
SQAP Version 1.1 (Date: 10-03-2020): This version of the SQAP contains updated test strategies for the simulation software.