# Software Quality Assurance (SQA) Plan
# By Team VotaFun

**Date: September 2023**

# Signature Page

**Prepared by: Ryan Teo Cher Kean** **Date: 14 Sep 2023**

**Reviewed by 1: Ng Yue Jie Alphaeus** **Date: 18 Sep 2023**

**Reviewed by 2: Roy Lau Run-Xuan** **Date: 20 Sep 2023**

**Approved by : Abdul Siddiq Bin Mohd Yussaini** **Date: 20 Sep 2023**

# Document Change Record

| Revision | Description of Change | Approved by | Date |
|---|---|---|---|
| **1.0** | **Preliminary Version 1** | **Ng Yue Jie Alphaeus** | **18/9/23** |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# Contents

# 1. Purpose and Scope

## 1.1. Purpose

The purpose of this Software Quality Assurance (SQA) Plan is to establish the goals, processes, and responsibilities required to implement effective quality assurance functions for the VotaFun project.

The Software Quality Assurance Plan provides the framework necessary to ensure a consistent approach to software quality assurance throughout the project life cycle. It defines the approach that will be used by the QAM and Software Quality (SQ) personnel to monitor and assess software development processes and products to provide objective insight into the maturity and quality of the software. The systematic monitoring of products, processes, and services will be evaluated to ensure they meet requirements and comply with policies, standards, and procedures, as well as applicable Institute of Electrical and Electronic Engineers (IEEE) and ISO standards.

## 1.2. Scope

The purpose of SQA is to ensure that the software developed does not deviate from the original intended product. SQA is also concerned with identifying any errors, omissions, inconsistencies, and alternatives, enhancements or improvements that can be made at any stage of development.

- Query component - This feature queries ChatGPT to get questions and multiple options that a group of players can vote for (e.g. four different cuisine options). After multiple rounds of voting, ChatGPT will suggest a specific final activity (e.g. McDonald's at Jurong Point).
- Voting component - This feature allows players to vote for an option they want to proceed with, given the question and options from ChatGPT.
- Start page - Users can host (create) or join a room on the start page.
- Lobby page - The host can invite more participants or kick players on this page. The host may also change parameters for queries later (budget, type of activity, region in Singapore). Players may choose to leave the room. The Host and players can also edit their username.
- Room page - The host and players will vote for the option they prefer on this page. ChatGPT will generate questions and the options for each question for the host and players to vote for. ChatGPT will generate the questions and options based on the voting results of the previous questions.
- Summary page - On this page, a summary of the options recommended by ChatGPT and votes will be displayed. If the Host or players are not satisfied with the activities recommended, there is also an option to start another round.

# 2. Reference Documents

- IEEE STD 730-2002, IEEE Standard for Software Quality Assurance Plans (http://standards.ieee.org/reading/ieee/std_public/description/se/730-2002_desc.html)
- ISO IEC 90003:2004 Software Standard (http://praxiom.com/iso-90003.htm)
- ISO/IEC 25010 Quality Model (https://iso25000.com/index.php/en/iso-25000-standards/iso-25010)
- Project Plan
- System Requirement Specifications

# 3. Management

This section describes the management organisational structure, its roles and responsibilities, and the software quality tasks to be performed.

## *3.1. Management Organisation*

The implementation of the quality assurance system is the responsibility of the Quality Assurance Manager (QAM).

### 3.1.1. Project Management

The Project Manager will be responsible for approving:

- The system requirement specification document
- The overall time scale for the project
- The choice of system development life cycle
- The choice of software development tools and techniques utilised
- The selection of project teams
- The training of project teams

### 3.1.2. Assurance Management

The QAM provides Project Management with visibility into the processes being used by the software development teams and the quality of the products being built. The QAM maintains a level of independence from the project and the software developers.

In support of software quality assurance activities, the QAM has assigned and secured Software Quality personnel from the pool of available SQ trainees to coordinate and conduct the SQ activities for the project and report back results and issues.

## *3.2. Tasks*

This section summarises the tasks (product and process assessments) to be performed during the development of software. These tasks are selected based on the developer's Project Plan and planned deliverables, and identified reviews.

### 3.2.1. Product Assessments

The following product assessments will be conducted by SQ personnel:

- Project Proposal
- System Requirements Specification (SRS)
- Project Plan
- Risk Management Plan
- Initial Prototype
- Test Plan
- Configuration Management Plan
- Change Management Plan
- Release plan
- Room system
- Query system
- Voting system
- Summary Page

**Project Proposal**: SQ personnel will peruse the project proposal to ensure it is well-defined and addresses all client concerns. The proposal should outline a solution that meets all functional and non-functional client requirements. The proposal will also clearly define deliverables, project management approach, and estimated costs to complete the project.

**System Requirements Specification (SRS)**: The SRS assessment involves verifying that it adheres to the Requirements Management Process outlined by the project team. SQ personnel will confirm that the SRS clearly defines how the system should behave, outlines support for crucial business processes, identifies any assumptions, and establishes important performance parameters for the system.

**Project Plan**: SQ personnel will review the project plan to ensure it provides a comprehensive roadmap for the execution of the project. This review includes looking at the project objectives, timelines, resource allocation, and task dependencies to ensure a well-structured project management approach.

**Risk Management Plan**: SQ personnel will assess the risk management plan to confirm that it identifies potential project risks, defines risk mitigation strategies, and outlines contingency plans. SQ personnel will ensure that it is robust to minimise any project risks and uncertainties during the entire project.

**Initial Prototype**: SQ personnel will evaluate the initial prototype to verify that it reflects the intended system functionality and design. They will check if it aligns with project requirements and if the initial prototype provides a tangible representation of the final product.

**Test Plan**: The test plan assessment will involve reviewing the plan for thoroughness in defining test cases, test environments, and testing methodologies. SQ personnel will ensure that it covers all aspects of quality assurance.

**Configuration Management Plan**: SQ personnel will review the configuration management plan to confirm it outlines a systematic approach for managing project configuration changes. The plan will include version control, documentation, and change tracking.

**Change Management Plan**: SQ personnel will assess the change management plan to ensure it defines a structured process for handling modifications to the project scope, requirements, and deliverables. SQ personnel will confirm that it minimises disruption and maintains project integrity.

**Release Plan**: SQ personnel will scrutinise the release plan to ensure it outlines a clear strategy for product deployment. The release plan should include versioning, rollout schedules, and any necessary user support.

**Room System, Query System, Voting System**: For these specific system assessments, SQ personnel will verify that each system meets its requirements, functions as intended, and meets the performance standards outlined in the SRS.

**Summary Page**: Lastly, the SQ personnel will evaluate the summary page for its completeness and accuracy in summarising key project details, findings, and recommendations. It should provide a concise overview for stakeholders.

SQ personnel will perform a thorough assessment and adhere to established quality standards to ensure that the execution of the project will be successful.

### 3.2.2. Process Assessments

The following process assessments will be conducted by SQ personnel:

- Requirement management process
- Change management process
- Project management
- Risk management
- Test management
- Release management

**Requirement Management Process**: SQ personnel will evaluate the requirement management process to ensure the project requirements are well-defined and adhered to during the project. This process should encompass the identification, documentation, validation, and traceability of requirements, ensuring they align with project objectives and client needs.

**Change Management Process**: SQ personnel will evaluate the change management process to verify that it provides a structured approach for handling changes with the project scope, requirements, and deliverables. SQ personnel will confirm that it includes clear procedures for change request submission, evaluation, approval, and implementation while minimising disruptions to the project.

**Project Management**: SQ personnel will review the project management process to assess its effectiveness in planning, executing, monitoring, and controlling the project. They will evaluate the project plan's alignment with objectives, resource allocation, task management, and adherence to timelines and budgets.

**Risk Management**: SQ personnel will examine the risk management process to ensure it identifies, analyses, and mitigates or reduces projects risks. SQ personnel will confirm that risk assessment, mitigation strategies, and contingency plans are well-documented and that there are ongoing risk monitoring.

**Test Management**: SQ personnel will assess the test management process to verify that it defines clear test objectives, test cases, test environments, and testing methodologies. They will ensure that the testing process includes comprehensive testing to validate the system's functionality and quality.

**Release Management**: SQ personnel will evaluate the release management process to ensure it outlines a systematic approach for planning, scheduling, and deploying product releases. SQ personnel will confirm that it includes version control, rollout schedules, and necessary user support.

SQ personnel will perform process assessments and ensure the project team follows established processes. If necessary, SQ personnel will recommend improvements to enhance the overall quality and efficiency of the project.

## *3.3. Roles and Responsibilities*

This section describes the roles and responsibilities for each assurance person assigned to the Project.

### 3.3.1. QAM

Responsibilities include, but are not limited to:

- Secure and manage SQ personnel resource levels
- Ensure that SQ personnel have office space and the appropriate tools to conduct SQ activities
- Provide general guidance and direction to the SQ personnel responsible for conducting software quality activities and assessments
- Assist SQ personnel in the resolution of any issues/concerns and/or risks identified as a result of software quality activities
- Escalate any issues/concerns/risks to project management

### 3.3.2. Software Quality Personnel

Responsibilities include, but are not limited to:

- Develop and maintain the project software quality assurance plan
- Generate and maintain a schedule of software quality assurance activities
- Conduct process and product assessments, as described within this plan
- Identify/report findings, observations, and risks from all software assurance related activities to the QAM

# 4. Documents

## 4.1. Purpose

This section identifies the minimum documentation governing the requirements, development, verification, validation, and maintenance of software that falls within the scope of this software quality plan. Each document below shall be assessed (reviewed) by SQ personnel.

## 4.2. Minimum Document Requirements

- System Requirement Specifications
- Project plan
- Design documentation
- User documentation
- Source code
- Quality Assurance reports
- Testing reports
- Risk Assessment and Mitigation Plan
- Deployment and release plan

# 5. Standards, Practices, Conventions and Metrics

## 5.1. Purpose

This section highlights the standards, practices, quality requirements, and metrics to be applied to ensure a successful software quality program.

## 5.2. Software Quality Programme

These practices and conventions are tools used to ensure a consistent approach to software quality for all programs/projects.

Referencing the ISO/IEC 25010 Quality Model, the four most important qualities for VotaFun are functionality, usability, reliability and portability.

Firstly, the web application must be functional and can perform all features defined in the project plan without errors. All functionalities and features of VotaFun must meet all the requirements the team has defined in the SRS.

Secondly, VotaFun must be usable by our target users (groups trying to decide on an activity to do in Singapore). The user interface of VotaFun must be functional, easy to use and understand with minimal guidance or tutorial. All these allow users to use VotaFun in a manner that allows for ease of activity planning.

Thirdly, VotaFun must be reliable and able to perform its tasks under any circumstances. The web application must function seamlessly on devices without significant response time delays and maintain an excellent uptime. The backend server must be able to handle all application programming interface (API) calls from the frontend. In the event of failure, the application must recover within a reasonable timeframe to minimise disruptions.

Finally, VotaFun must be portable since our application will be used by users on their devices, likely with different operating systems and web browsers. User experience must be smooth and enjoyable, and this user experience must be consistent across web browsers.

### 5.2.1. Standard Metrics

The following standard metrics are the minimum planned metrics that will be collected, reported, and maintained in the area of software quality assurance:

- Defect Density (no. of defects/line of code) - Usability, Reliability
- Source Lines of Code (SLOC) - Reliability
- Build Stability - Reliability
- Cross-Browser Compatibility - Portability
- Platform independence - Portability
- Release Readiness - Functionality

# 6. Software Reviews

## *6.1. Purpose*

This section identifies the number and type of system/subsystem reviews and engineering peer reviews that will be supported by the SQ Personnel. The project milestone chart, and the SQ Personnel resource levels determine the reviews that are supported.

## *6.2. Minimum Software Reviews*

For each review, SQ will assess the review products to assure that review packages are being developed according to the specified criteria, the review content is complete, accurate, and of sufficient detail, and Requests for Action are captured, reviewed, and tracked to closure. In addition, SQ will assess the processes used to conduct the reviews to determine if appropriate personnel are in attendance, correct information is presented, entry and exit criteria are met, and appropriate documents are identified for update.

The following software reviews will be assessed by SQ:

- Project Plan Review
- Requirements Analysis Review
- Software Design Review
- Test Plan Review
- Acceptance Review
- Release Plan Review
- Risk Assessment and Mitigation Plan Review

# 7. Test

SQ personnel will assure that the test management processes and products are being implemented per Test Plan. This includes all types of testing of software system

components as described in the test plan, specifically during integration testing (verification) and acceptance testing (validation). SQ personnel will monitor testing efforts to assure that test schedules are adhered to and maintained to reflect an accurate progression of the testing activities. SQ will assure that tests are conducted using approved test procedures and appropriate test tools, and that test anomalies are identified, documented, addressed, and tracked to closure. In addition, SQ will assure that assumptions, constraints, and test results are accurately recorded to substantiate the requirements verification/validation status. SQ personnel will review post-test execution related artefacts including test reports, test results, problem reports, updated requirements verification matrices, etc.

## 7.1. Unit Testing

SQ personnel will track all unit test activity, monitor test coverage and evaluate the effectiveness of test cases. They will also ensure that testing follows the best practices in the industry and will track all issues raised from unit tests until completion. Unit tests cover the frontend (game, start and lobby page) and the backend (database interface, socket system and the backend APIs) components.

## 7.2. Integration Testing

SQ personnel will be responsible for tracking the integration testing of VotaFun. They will verify and oversee the integration testing process to ensure that all components of VotaFun will work seamlessly together. SQ personnel will raise any issues arising from integration testing and ensure problems are addressed. Integration testing ensures that modules will work together to achieve their intended outcome. For example, the game page must be able to call the backend API, or the backend APIs must integrate with the database interface to store results.

## 7.3. System Testing

SQ personnel will inspect the system testing process, ensuring that the plan will properly assess VotaFun functionality and performance as stated in the requirements. All individual components are linked and tested to ensure that functionality and quality requirements are met. Similarly, SQ personnel will track any issues that arise from system testing.

## 7.4. User Acceptance Test

SQ personnel will oversee the user acceptance test (UAT) process and ensure that the UAT follows industry standards. UAT allows the client to validate that the requirements of the software are met and that there is no missing functionality. A Beta application will be set up for clients to test. SQ personnel will monitor any issues and problems arising from UAT.

# 8. Problem Reporting and Corrective Action

SQ personnel generate, track, and trend assessment findings and observations in a centralised Reporting and Corrective Action System on Google Drive.

## *8.1. Problem Identification*

### 8.1.1. Issue Discovery

8.1.1.1. Problems and issues are identified through various means, including user feedback, automated testing, manual testing, and system performance monitoring. Team members can report an issue by documenting it in the GitHub issue tracking system, including a description of the problem, steps to reproduce it, and relevant context.

### 8.1.2. Issue Categorization

8.1.2.1. Upon identification, issues are categorised based on their severity and impact on the software and user experience. Categories include Critical, Major, Minor, and Enhancement.

## *8.2. Problem Documentation*

### 8.2.1. Issue Logging

8.2.1.1. All identified problems are logged into the issue tracking system on GitHub or an equivalent tool. Each issue entry will include a unique identifier, description, steps to reproduce the issue, priority, and assignee.

### 8.2.2. Prioritisation

8.2.2.1. The severity and impact of each issue are assessed to determine its priority. Priority levels help the developer team to decide what issues they should be addressed first.

## *8.3. Problem Resolution*

### 8.3.1. Issue Assignment

8.3.1.1. Team members will be assigned to issues based on their expertise and workload. The assignee takes ownership of the problem and its resolution.

### 8.3.2. Root Cause Analysis

8.3.2.1. The assigned team member conducts a thorough analysis to identify the root cause of the problem. Understanding the underlying issue is crucial for effective resolution.

### 8.3.3. Corrective Action

8.3.3.1. The assigned team member will work on implementing the necessary corrective action once they identify the root cause. The solution should follow the project's coding standards and industry best practices.

### 8.3.4. Verification

8.3.4.1. After implementing the corrective action, the assigned team member performs testing to ensure that the raised issue is resolved.

Verification may include running relevant test cases or replicating the issue's initial scenario.

### *8.4. Problem Closure*

#### 8.4.1. Verification of Resolution

8.4.1.1. The assigned team member verifies that the issue is resolved by retesting it and confirming that the problem no longer exists.

#### 8.4.2. Documentation

8.4.2.1. The team member updates the issue tracking system to reflect the resolution, including details of the corrective action.

#### 8.4.3. Communication

8.4.3.1. Relevant stakeholders, including the project manager, quality assurance team, and affected end-users (if necessary), are notified of the problem's resolution.

#### 8.4.4. Monitoring

8.4.4.1. Post-resolution, the issue remains in the tracking system for a specified period to ensure that the issue does not reoccur and for documentation purposes.

### *8.5. Preventive Action*

#### 8.5.1. Analysis

8.5.1.1. If the problem is significant or recurring, the team conducts a deeper analysis to identify preventive measures. The goal is to prevent similar issues from arising in the future.

#### 8.5.2. Implementation

8.5.2.1. Preventive measures are implemented, which may include process improvements, code reviews, or enhanced testing procedures.

#### 8.5.3. Monitoring and Feedback

8.5.3.1. The effectiveness of preventive actions is monitored over time, and feedback from team members is collected to make any necessary adjustments.

By following this "Problem Reporting and Corrective Action" process, the project team ensures that identified issues are promptly addressed and resolved, and, when necessary, preventive measures are put in place to enhance the overall quality of the software.

## 9. Tools, Techniques and Methodologies

SQ personnel will require access to the following:

### *9.1. Software Quality Tools*

- Microsoft Office tools (i.e., Word, Excel, and PowerPoint)
- Github project management tools
- Discord server (For communication within the project team)

- Google Drive (For collaborative work on documentation)

# 10. Media Control

SQ deliverables will be documented in one of the following Microsoft software applications: Word, Excel, or PowerPoint. Deliverables will be in soft copy, with the exception of completed checklists from process and product assessments. See Section 12 for additional details on the collection and retention of key records. Software Quality personnel will request space on the project's secured server for SQ records. This server is password protected and backed up nightly.

The project team will use the following services for media control:

1. MediaWiki
2. GitHub
3. Google Drive

MediaWiki is a free and open-source application developed for creating and managing wikis. Wikis are collaborative websites that allow multiple users to edit and contribute content.

GitHub is a web-based platform that streamlines the version control process and assists with collaborative software development. It offers features and tools to help software development teams to store, track and manage their software projects across multiple devices.

Google Drive is a cloud-based file storage and synchronisation service developed by Google. It is a member of the Google Workspace ecosystem and offers users a range of features like storing, organising, and sharing files and documents. Google Drive also has a built-in office suite Google Sheets, which offers real-time collaboration of files.

# 11. Supplier Control

Not applicable for this project.

# 12. Record Collection, Maintenance, and Retention

SQ personnel will maintain records that document assessments performed on the project. Maintaining these records will provide objective evidence and traceability of assessments performed throughout the project's life cycle. There are two types of records that will be maintained: Hardcopy and Electronic. SQ personnel will maintain electronic or hard copies of all assessment reports and findings. SQ Project folders will contain hardcopies of the assessment work products such as completed checklists, supporting objective evidence, and notes.

The table below identifies the record types that will be collected, as well as the Record Custodian and Retention period

| Record Title | Record Custodian | Record Retention |
|---|---|---|
| SQA Assessments | SQ Personnel | One Year |

| | | |
|---|---|---|
| SQA Checklists | SQ Personnel | One Year |
| Deliverable Defects | SQ Personnel | One Year |

# 13. Training

SQ personnel have fundamental knowledge in the following areas through prior experience, training, or certification in methodologies, processes, and standards:

- Audits and Reviews (Assessments)
- Risk Management
- Software Assurance
- Configuration Management
- Software Engineering
- ISO 9001, ISO 9000-3
- CMMI
- Verification and Validation

# 14. Risk Management

SQ personnel will assess the project's risk management process and participate in weekly risk management meetings and report any software risks to the QAM and the project manager.

# 15. SQA Plan Change Procedure and History

SQ personnel are responsible for the maintenance of this plan. It is expected that this plan will be updated throughout the life cycle to reflect any changes in support levels and SQ activities. Proposed changes shall be submitted to the Quality Assurance Manager (QAM), along with supportive material justifying the proposed change.