Software Quality Assurance (SQA) Plan

By Team Kim's Convenience

Date: September 2021

Signature Page

Prepared by: Hussain Kheriwala
Jeremy Book Kay Yip
Parthasarthi Taneja

Date: 6 September 2021

Reviewed by 1: Hussain Kheriwala	**Date: 10 September 2021**
Reviewed by 2: Parthasarthi Taneja	**Date: 14 September 2021**
Reviewed by 3: Wong Ying	**Date: 14 September 2021**
Reviewed by 4: Desmond Yap	**Date: 14 September 2021**
Reviewed by 5: Jeremy Book Kay Yip	**Date: 14 September 2021**
Reviewed by 6: Ta Anh Duc	**Date: 14 September 2021**

Approved by : Wong Ying **Date: 14 September 2021**

Document Change Record

Revision	Description of Change	Approved by	Date
1.0	Initial Template + Content	Jeremy	04/09/21
1.1	Added purpose, reference documents and management sections	Jeremy	05/09/21
1.2	Added standards, software reviews and test sections	Hussain	06/09/21
1.3	Added rest of the sections and completed the document	Jeremy	06/09/21
1.4	Revision of document	Wong Ying	07/09/21
2.0	Update document based on revision	Desmond	08/09/21
2.1	Final revision	Duc	14/09/21

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 Kim's Convenience Modern Cinema 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.

Kim's Convenience Modern Cinema (KCMC) is a movie booking web application which provides personalised movie recommendations with the help of deep learning to create a recommendation system. KCMC aims to help local cinema businesses to quickly recover from the impact of the pandemic by introducing a recommender system, promoting targeted advertisement.

PyTorch is used to train a model which recommends movies based on the user's booking history. SQLite database will be used to store movies, shows and users data. Django has been selected for our KCMC web application.It provides support for user authentication and authorisation, integration with databases and delivering scalable, secure and versatile webpage. KCMC adopts the Model-View-Controller (MVC) architectural pattern which separates the application into three main logical components.

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)
- Project Plan
- System Requirement Specifications
- Project Proposal and Timeline
- Team Backlog
- Risk Management
- Test Plan
- CMMI Level 2
- Design Report on Software Maintainability
- Configuration Management Plan
- Change Management Plan
- Release Plan

3. Management

This section describes the management organizational 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 summarizes 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.

The software will be developed by the front-end development team and the back-end development team. The QA team will ensure quality control for the development of the software according to the ISO 9126 Quality Model.

3.2.1. Product Assessments

The following product assessments will be conducted by SQ personnel:

- KCMC Recommending Model
- KCMC Web Application

3.2.2. Process Assessments

The following process assessments will be conducted by SQ personnel:

- Requirements management process
- Change management process
- Configuration management process
- Testing process
- Release process
- Server administration process
- Technical support process

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

Documents to be reviewed:

- System Requirement Specifications
- Project Proposal and Timeline
- Team Backlog
- Risk Management
- Test Plan
- CMMI Level 2
- Design Report on Software Maintainability
- Configuration Management Plan
- Change Management Plan
- 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.

The primary target audience of this project are elderly users who may not be well-versed with modern technology. Correspondingly, the key qualities we will be focusing on emphasise improving the convenience effectiveness of our application:

Usability requires that the user execute the relevant application use cases effectively, such as being able to easily browse and book movies with minimal confusion and external assistance. Font, text size and colours will be chosen to maximise readability, i.e. bigger fonts and easily distinguishable colours and shades to ensure all the texts are clear and readable.

Consistency is crucial, as having as it helps them feel more in control of their tasks and minimises confusion. The navigation of common features such as browsing movies, booking tickets and login/logout buttons should be kept consistent throughout the software. The movie booking process should also be kept consistent and straightforward to avoid any confusion.

Ease of Understanding is another key factor as we must take into account the diverse age of users of this system. As such, who are unable to quickly understand how to use the system may lose patience and give up on using it entirely.

The user interface will have a minimalistic, simple design and with minimal buttons per screen, thereby reducing the number of unexpected interactions and complicated features and operations.

Error messages, if any, should provide visual and useful information on how to resolve the issue easily. It should be designed in a way not to alarm the user or cause panic when they encounter an issue.

Functionality is also emphasised to ensure that the features introduced and implemented are of high quality and will help to improve the overall user experience of the cinema-goers

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 quality assurance:

- Number of user registrations per week
- Number of movie bookings per week
- Average time taken for a user to book a movie
- Application crash rate / Error rate
- Number of bugs per test
- Requirements coverage

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

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 artifacts including test reports, test results, problem reports, updated requirements verification matrices, etc.

8. Problem Reporting and Corrective Action

SQ personnel generate, track, and trend assessment findings and observations in a centralized Reporting and Corrective Action System.

Asana, an application designed to help teams organize, track, and manage work will be used to communicate data and track the status of the communication. Team Members assigned with a task can indicate completion upon completing their assigned task for the QAM and project manager to view. Furthermore, QAM or project managers can create follow up tasks if there are any corrective actions to be taken.

GitHub has an Issue tab to allow its development team members to report/create an issue which then a QAM can perform required corrective actions such as assigning members to fix the issue within a certain timeline.

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)
- Asana
- GitHub
- Wiki

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.

Documentations are saved in the project team's Google Drive folder. The folder is only shared with team members in the project, each of whom needs to enter their Google account credentials. Furthermore, Google Drive has version control features for every documentation.

11. 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

12. 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

13. Risk Management

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

SQ personnel will perform a monthly 5 step iterative process as follow:

1. Identify potential threats and their probability occurrence by using various tools and risk management techniques as well as the risk related knowledge from previous projects and other teams' experience.
2. Analyze the impact of a particular risk. Having this in mind, a proper course of action and business decisions can be formulated to mitigate the risks.
3. Prioritize urgency of multiple risks to better allocate resources.
4. Respond to individual risk based on their occurrence, impact and priority to decide the course of action for risk mitigation.
5. Monitor the plan throughout the development process to decide if necessary alterations/modifications are required.

14. 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.