Software Quality Assurance (SQA) Plan

by The Team

Date: 22 September 2019

Signature Page

Prepared by: Li Shanlan Date: 20 Sept 2019

Reviewed by 1: Simon El Nahas Christensen Date 21 Sept 2019

Reviewed by 2: Akshaya Muthu Date: 22 Sept 2019

Approved by : Cao Ngoc Thai Date: 22 Sept 2019

Document Change Record

Revision	Description of Change	Approved by	Date
0.10	Initial Template	Li Shanlan	16.09.18
0.20	Formatting and Finalization	Cao Ngoc Thai	22.09.18

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 **E-tendance** 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 covers the description of the Software Quality Assurance workflow, standards applied and techniques to be used. SQA is also concerned to identify any errors, omissions, inconsistencies, and alternatives, enhancements or improvements that can be made at any stage of development.

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

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 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.

3.2.1 Product Assessments

The following product assessments will be conducted by SQ personnel:

Product Assessments 001	**Requirements of System**
Entry Criteria	After first lab session
Description	Identifying the core functionalities of the E-tendance and special features

Exit Criteria	Functional and non-functional requirements documentation
Procedure	QAM will review the documentation
Notes	Nil

Product Assessments 002	**Code Quality Check**
Entry Criteria	For every merge/pull request to development branch
Description	Code Review
Exit Criteria	Code review passed
Procedure	Do code review for every merging operation/ pull request. QAM approves pull requests.
Notes	Nil

Product Assessments 003	**Initial Prototype**
Entry Criteria	Functional and non-functional requirements documentation
Description	The initial prototype will contain the core functionalities of the E-tendance
Exit Criteria	Review of functionality checks using black-box testing Bug reports
Procedure	QAM will review product requirements that were stipulated in the previous meetings. QAM will then test the prototype for bugs QAM will update the bug report
Notes	Nil

Product Assessments 004	**Subsequent Prototype**
Entry Criteria	After initial prototype
Description	Improvements will be made based on the report of initial prototype. Supplementary functionalities will be added to the subsequent prototype.

Exit Criteria	Review of functionality checks Review of previous bugs Bug reports
Procedure	QAM will review product requirements that were stipulated in the previous meetings. QAM will review the previous bugs. QAM updates bug reports.
Notes	Nil

Product Assessments 005	**Finalized Product**
Entry Criteria	When prototype consists of all functionalities
Description	All functionalities of the E-tendance will be implemented in working conditions
Exit Criteria	Review of functionality checks No bugs found
Procedure	QAM will review all product requirements. QAM will check across bug reports to make sure all bugs are resolved. QAM certifies that the product is free of bugs.
Notes	Nil

3.2.2 Process Assessments

The following process assessments will be conducted by SQ personnel:

Process Assessment 001	**Requirement management process**
Entry Criteria	At the end of every iteration of product
Description	This stage is to ensure that the requirements are implemented and the product does not deviate from the original design.
Exit Criteria	Quality assurance policy

Procedure	Review list of requirements given by contract and project plan. Set schedule for conducting reviews and creating audits of ex-sell prototypes . Every prototype introduced will be checked for the requirements conformity (functional and non-functional requirements). Compile findings into a report to be sent to Project Manager, QAM and lead Developer to go through.
Notes	Nil

Process Assessment 001	**Change management process**
Entry Criteria	Once a change of software requirements/design or team policy is raised by any one of the team members.
Description	This process serves as a guideline for the project team to follow should any change be requested.
Exit Criteria	Change request accepted or rejected
Procedure	Changes of software requirements/design or team policy to be raised during weekly meeting of the team. Changes to be discussed during the meeting. Team members vote for the approval of the changes. Approved changes to be recorded to documentations.
Notes	Nil

Process Assessment 002	**Evaluation of Software Environments**
Entry Criteria	Project Plan
Description	This stage ensures that the libraries or technology used to develop E-tendance are in conformance with the contract

Exit Criteria	Non-conformance report (if any)
Procedure	Before every milestone as noted in the project schedule, conduct a meeting with developer lead to level the type of libraries and packages that may be used for development. Requirements will be explicitly stated and updated should any changes occur. During the development phase, every prototype produced will be scrutinized for any discrepancies in conformance. If there exists and discrepancies in conformance, SQ personnel will highlight to QAM and together with the lead developer, will hold an immediate meeting to discuss and highlight issues with the abovementioned.
Notes	Nil

Process Assessment 003	**Evaluation of Process Measurements for Conformance**
Entry Criteria	Project Plan
Description	This stage ensures that a quantitative way to measure the progress of the development of the project is introduced, as well as reviewed regularly to ensure that the project is heading forth as planned
Exit Criteria	Non-conformance report (if any)
Procedure	Every milestone, QAM will liaise with lead developer to estimate the lines of code needed to achieve that milestone, as well as different components of the milestone. Lead Developer will disseminate information to developers on the team At the end of every month, developers will be required to submit a simple report on the component that they are working or have

	worked on, as well as the lines of code produced (This can be done through version control commits) Should any fail to meet the estimated lines of code or progress, measures will be discussed between lead developer and QAM to improve on current status. These measures will be created as Trello tasks.
Notes	Nil

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

- Software Requirement Specification
- Quality Management
- Software Model Prototype
- Risk Management
- Design report on software maintainability
- Configuration Management Plan
- Release Plan
- Test Plan and Documentation

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.

1. Functionality

Our application E-tendance should provide four functionalities:

a. Automatically recognize students by real-time camera and take attendance for them.

b. Interfaces for students to register their faces with the system

c. Interfaces for professors to enter course information.

d. Generate reports to professors (present students, absent students, late students).

2. Simplicity

Our application must be simple to use. The user interface should be intuitive and the instructions given on screen should be minimal. From the software development's perspective, the application should have a simple design to serve its functionalities.

3. Reliability

The application should have minimum probability of failure-free software operation for a specified period of time in a specified environment.

- Error rate : 1% wrong facial recognition
- Latency: 99% less than 30ms
- Availability: 99% of the time

4. Ease of Understanding

The application should have a clean code base, simple logic, and proper documentation for easy understanding.

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:

- Fan in/Fan out
- Length of code (simplicity)
- Cyclomatic complexity
- Depth of conditional nesting
- Weighted methods per class
- Length of user manual = 0
- Prof is able to use E-tendance without any prior training (simplicity)

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, SQM 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, SQM 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 SQM:

- Project Plan Review
- Requirements Analysis Review
- Software Design Review
- Code 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 an online Reporting and Corrective Action System. On our shared google drive there should be an Excel sheet named `ReportingAndCorrectiveAction.gsheet`.

8.1 Possible Problems which may arise

Document Problems:

- Non-compliance with other project documents
- Incompleteness
- Grammar errors

Code Problems:

- Lack of functionality
- Wrong functionality
- Non-Compliance with coding standards

8.2 Problem Reporting Procedure

When the documentation requires rectification due to an error or being incomplete, the member which discovers the problem will be required to perform the following procedures:

1. Report the situation to the Project Manager and QA Manager
2. The Project Manager provides suggestions to the corresponding team member to complete or change accordingly. The appointed member will then be responsible for solving the problem.
3. The team member will have to notify the Project Manager and QA Manager when the problem is solved. They will have to check that the problem is solved.
4. When the appointed member is unable to solve the problem, they will have to notify the Project Manager and QA Manager. They will then set up a meeting with the group and a decision must be made to solve the problem.

In the event when the system or code contains a function which might be incomplete, does not comply with the requirements or is missing totally, the member who discovers the problem will be required to perform the following procedures:

1. Report the situation to the Project Manager and Lead Developer.
2. The Project Manager provides suggestions to the corresponding team member to complete or change accordingly. The appointed member will then be responsible for solving the problem.

3. The appointed member will have to notify the Project Manager and Lead Developer when the problem is solved. They will have to check that the problem is solved.
4. When the appointed member is unable to solve the problem, they have to notify the Project Manager and Lead Developer. They have to set up a meeting with the group and a decision must be made to deal with the problem.

All changes and revisions are to be updated and notified during our weekly meetings.

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)
- Google Drive, Google docs, Google sheet
- Github
- Code coverage: IDE
- Continuous Integration: jenkins
- Static Code Analysis: WebStorm

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.

Our records are stored and maintained on the following services:

1. Google Drive
2. GitHub

Google Drive is used as a file storage and backup of documents initially created. This service allows users to share and store files within the group easily and allows users to edit documents concurrently. Documentation of the project and the team are stored and shared on Google drive.

GitHub is used for version control across all team members. This service helps to store, share and revise different versions of source code files and documents being uploaded. Programming Code of this application are stored both on Github and on each team members' PC locally.

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.

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.