

**Campus Meal Ordering System**

**Software Quality Assurance (SQA) Plan**

**By *Team Foodie***

**Lab Group: TS3**

**Date: September 2020**

**Instructors:**

**Bo An**

**Leong Yin Yoke Junie**

**Qiu Wei**

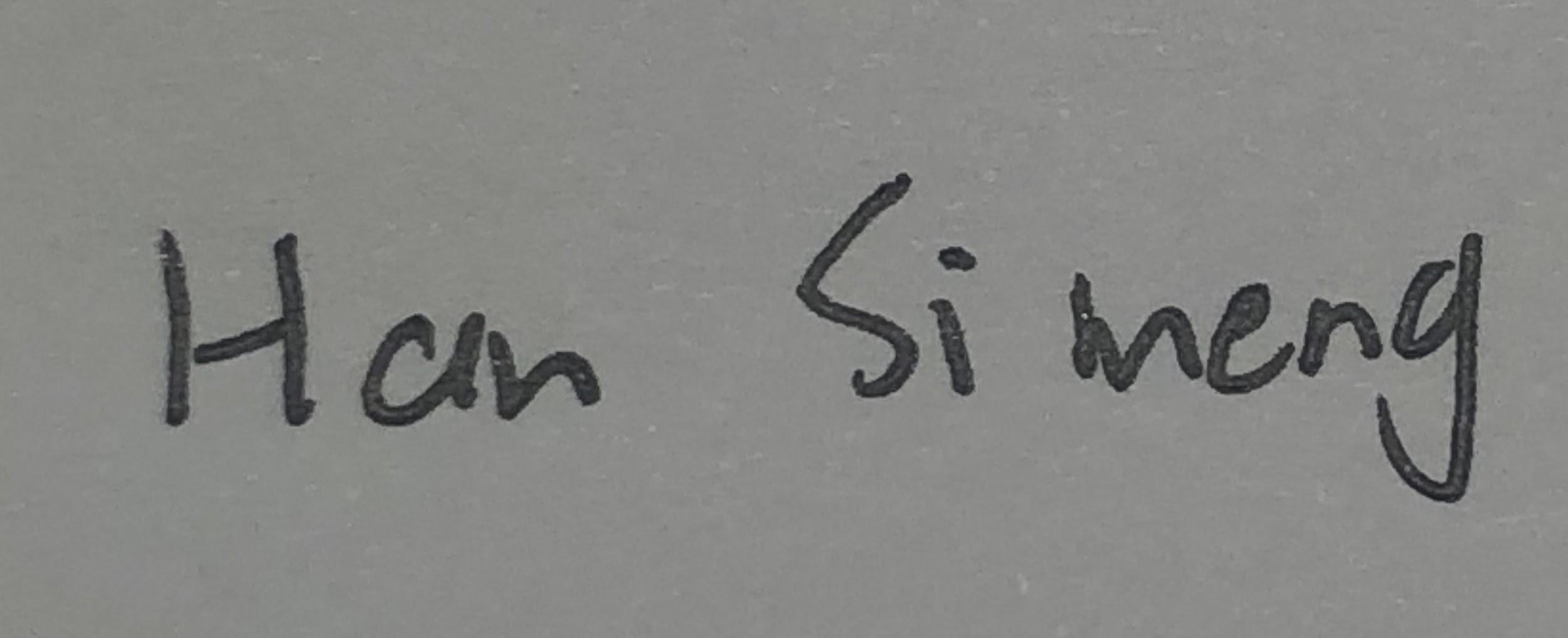
**Wang Rundong**

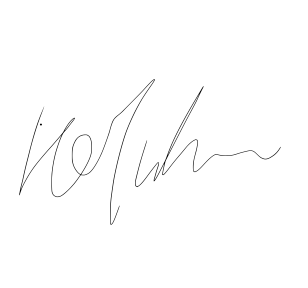
# **SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

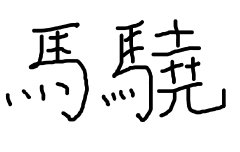
**NANYANG TECHNOLOGICAL UNIVERSITY**

**Signature Page**

**Prepared by: 1. Loh Yi Xuan Renice  
 2. Yeoh Jun Yi Date: 13/9/2020**

**Reviewed by 1:**  **Date: 04/10/2020   
 Han Si Meng**

**Reviewed by 2:  Date: 04/10/2020   
 He Yu Hao**

**Approved by :**  **Date: 04/10/2020   
 Ma Xiao**

**Document Change Record**

| **Revision** | **Revised By** | **Revision Date** | **Description of Change** | **Approved**  **by** |
| --- | --- | --- | --- | --- |
| **0.10** |  |  | **Initial Template** |  |
| **0.15** | **Renice** | **8/9/2020** | **Purpose and Scope** | **Ma Xiao** |
| **0.3** | **Renice** | **12/9/2020** | **Management** | **Ma Xiao** |
| **0.4** | **Renice** | **12/9/2020** | **Documents** | **Ma Xiao** |
| **0.6** | **JunYi** | **12/9/2020** | **Software Quality Programme** | **Ma Xiao** |
| **0.7** | **JunYi** | **12/9/2020** | **Problem Reporting and Corrective Action** | **Ma Xiao** |
| **0.75** | **JunYi** | **12/9/2020** | **Software Quality Tools** | **Ma Xiao** |
| **0.80** | **Renice** | **13/9/2020** | **Media Control** | **Ma Xiao** |
| **1.0** | **Renice** | **13/9/2020** | **Corrective Action** | **Ma Xiao** |
|  |  |  |  |  |

**Contents**

[1.](#_gjdgxs) Purpose and Scope 5

[1.1.](#_30j0zll) Purpose 5

[1.2.](#_1fob9te) Scope 5

[2.](#_3znysh7) Reference Documents 5

[3.](#_tyjcwt) Management 5

[3.1.](#_3dy6vkm) Management Organisation 6

[3.1.1.](#_1t3h5sf) Project Management 6

[3.1.2.](#_4d34og8) Assurance Management 6

[3.2.](#_2s8eyo1) Tasks 6

[3.2.1.](#_17dp8vu) Product Assessments 7

[3.2.2.](#_3rdcrjn) Process Assessments 7

[3.3.](#_26in1rg) Roles and Responsibilities 7

[3.3.1.](#_lnxbz9) QAM 7

[3.3.2.](#_35nkun2) Software Quality Personnel 7

[4.](#_1ksv4uv) Documents 8

[4.1.](#_44sinio) Purpose 8

[4.2.](#_2jxsxqh) Minimum Document Requirements 8

[5.](#_z337ya) Standards, Practices, Conventions and Metrics 8

[5.1.](#_3j2qqm3) Purpose 8

[5.2.](#_1y810tw) Software Quality Programme 8

[5.2.1.](#_4i7ojhp) Standard Metrics 8

[6.](#_2xcytpi) Software Reviews 9

[6.1.](#_1ci93xb) Purpose 9

[6.2.](#_3whwml4) Minimum Software Reviews 9

[7.](#_2bn6wsx) Test 9

[8.](#_qsh70q) Problem Reporting and Corrective Action 10

[9.](#_3as4poj) Tools, Techniques and Methodologies 10

[9.1. Software Quality Tools 10](https://docs.google.com/document/d/1279HbOAYr9I5DBfGJ05_HNpUIC7TuMgt/edit#heading=h.32hioqz)

[10. Media Control 10](https://docs.google.com/document/d/1279HbOAYr9I5DBfGJ05_HNpUIC7TuMgt/edit#heading=h.1hmsyys)

[11. Supplier Control 10](https://docs.google.com/document/d/1279HbOAYr9I5DBfGJ05_HNpUIC7TuMgt/edit#heading=h.41mghml)

[12.](#_147n2zr) Record Collection, Maintenance, and Retention 10

[13.](#_3o7alnk) Training 11

[14.](#_23ckvvd) Risk Management 11

[15.](#_ihv636) SQA Plan Change Procedure and History 11

# **Purpose and Scope**

## **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 Campus Meal Ordering System (CMOS) 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.

## **Scope**

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

CMOS is specifically designed for students and staff on NTU campus provide them many quality-of-life functionalities such as a myriad of food that is more readily accessible while saving them time and money in tandem. As CMOS is meant to increase quality of life for students and staff, our design approach is to ensure that it is available and accessible to everybody. And as such, our team will be employing Flutter such that CMOS will be able to run on both iOS and Android platforms.

# **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

# **Management**

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

## **Management Organisation**

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

### **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

### **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.

## **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.

| **Software Development Life Cycle** | **Task** |
| --- | --- |
| Requirement Specification | * Identify project attributions * Review on client requirements * Review on software requirements |
| Software Design | * Review on software design * Review on system architecture necessary to match client’s requirements |
| Implementation | * Review on test readiness * Review on system to determine if CMOS is aligned with client’s requirements |
| Testing | * Review on test results |
| Maintenance | * Post-mortem meeting |

### **Product Assessments**

The following product assessments will be conducted by SQ personnel:

* Weekly completion and updating of functional requirement checklist
* Weekly completion and updating of non-functional requirement checklist
* Bi-weekly meetings to facilitate CMOS progress status updates with relevant stakeholders
* Constant monitoring and correction to ensure CMOS compliance with ISO 9126 specifications

### **Process Assessments**

The following process assessments will be conducted by SQ personnel:

* Weekly verification of adequate program code commenting
* Weekly verification of design conformity to client requirements
* Weekly monitoring actual progress of project against milestone plan and timeline
* Follow Plan-Do-Check-Act (PDCA) cycle when carrying out changes

## **Roles and Responsibilities**

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

### **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

### **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

# **Documents**

## **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.

## **Minimum Document Requirements**

* + 1. **System Requirement Specifications (SRS)**

The SRS must document all functional requirements, non-non functional requirements and constraints in an atomic, unambiguous, traceable and verifiable manner. User-System interaction will be illustrated in the Use Case Diagram and described in the Use Case description.

* + 1. **Quality Assurance Plan (QAP)**

The Quality Assurance Plan documents the essential guidance, direction and focus of CMOS to provide the client appropriate insight into processes and products being used and developed.

* + 1. **Risk Management Plan**

The Risk Management Plan consists of the process, risk identification and risk analysis as well as how we plan to cope with any risk that the CMOS development team may face.

* + 1. **Verification and Validation Plan (VVP)**

The VVP must list out the necessary procedures and activities used to verify and validate the CMOS project and the methods used in a clear manner.

* + 1. **Verification and Validation Report (VVR)**

The VVR clearly documents the results of the execution of VVP.

* + 1. **Capability Maturity Model Integration(CMMI) process model**

The CMMI process model aims to define the Level 2 of CMMI since various aspects such as goals, models and practices need to be defined and agreed cohesively for the CMMI to work effectively. CMMI process model provides a clear definition of what individuals in a team should follow in order to achieve effective processes throughout the project lifespan.

* + 1. **Change Management Plan**

The Change Management Plan must be used when changes are made to the existing products or processes to ensure effective implementation of the changes, which helps manage the change process and ensures control in budget, schedule, scope, communication and resources.

* + 1. **Software Configuration Management Plan (SCMP)**

The SCMP defines CMOS’s structure and methodology. SCMP must document and inform project stakeholders about the configuration management with the project and what tools will be used and how they will be applied by the project to promote success.

* + 1. **Release Management Plan**

The Release Management Plan must describe the portions of the system functionality implemented within each release stage namely the management, planning, scheduling, and controlling of CMOS development through every stage and environment involved, including testing and deploying software releases.

* + 1. **User Documentation**

The scope of this document encompasses the following areas:

⦁ User Manual

⦁ User Guide

The user documentation must describe the required data & control inputs, input sequences, options, program limitations and minimum software and hardware requirements needed for running the CMOS. All error messages, along with their respective corrective actions for resolving the messages must also be described in the document. Additionally, the steps required for reporting user-identified errors and problems must be outlined in the document. The user documentation must be reviewed by the Lead Developer before it is approved by the QA Manager.

# **Standards, Practices, Conventions and Metrics**

## **Purpose**

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

## **Software Quality Programme**

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

| **Qualities** | **Description** | **Considerations** |
| --- | --- | --- |
| Functionality | CMOS must be able to provide functions which meet both stated and implied needs when it is under specified conditions. | Different levels of testing will be performed to ensure that the software performs according to the specifications in the system specifications |
| Usability | The user interface of CMOS must be easy to using the 5 components as guidance:   1. Learnability 2. Efficiency 3. Memorability 4. Errors 5. Satisfaction | The design of the user interface will be measured throughout the development process - from wireframes to the final deliverable - to ensure maximum usability. |
| Reliability | CMOS must be able to maintain it’s specified level of performance under stated conditions for a period of time | The CMOS development team will practice software reliability techniques.  During the implementation phase:   1. Facilitation of code reviews 2. Software robustness and coverage testing   During the testing phase:   1. Software Reliability standard measurements and metric |
| Maintainability | CMOS must be capable of modification. Modification includes corrections, improvements and also adaptations to meet new requirements or new environments | The architecture of CMOS will be designed to have independent client and server components. The independent components allow reliability to be achieved by providing multiple instances of the same component as backups |

### **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:

| **Software Metrics** | **Description** |
| --- | --- |
| Fan-in/Fan-out | Fan-in is a measurement of the number of functions or methods that call another function, or method. Fan-out is the number of functions that are called by other functions. |
| Length of code | This is used in measuring the size of the CMOS program. In most cases, the larger the size of the code of a component, the more complex and error-prone that component is likely to be. Length of code has been shown to be one of the most reliable metrics for predicting error-proneness in components, which will be used to predict the likelihood for errors in the SMRPS program. |
| Cyclomatic Complexity | This is a measurement of the control complexity of the CMOS project. |
| Length of identifiers | It is a measurement of the completeness of the documentation. This helps CMOS development team programmers maintain the program efficiently. |
| Depth of conditional-nesting | This is a measure of the depth of nesting of if-statements in a program. Deeply nested if statements are hard to understand and are potentially error-prone. |
| Fog Index | This is a measure of the average length of words and sentences in documents. The higher the value for the Fog index, the more difficult the document is to understand. |

# 

# **Software Reviews**

## **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.

## **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

# **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.

# 

# **Problem Reporting and Corrective Action**

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

When a failure has been identified, an implementation plan created in accordance with the Corrective Action Procedure (CAP) must be followed to eliminate future occurrences of the failure. Any problems identified will be logged in and compiled into an excel spreadsheet. Detailed perusal and trend assessment will be done as needed. All tickets have to be updated weekly to ensure that all tickets are actively being resolved.

Problems that require corrective action will be either brought up to the QA manager or project manager. Depending on the severity of the issue, it will be discussed during weekly meetings. All discussion and proposed resolutions will be logged in the excel sheet even after resolution of said problem for future reference.

The created implementation plan must be approved by both the QA Manager and Project Manager before it can be enacted. The QA Team will be responsible for adapting the implementation plan in accordance with the specified failure using the CAP. The framework for the plan is outlined below:

1. Review Failure History to find for related failure cases
2. Check if the existing solution can be used
3. Reenact the cause of failure
4. Document the cause of failure
5. Evaluate and identify possible solutions
6. Document the cause of failure and actions taken for future reference
7. Test that the problem is solved by recreating the cause of failure and using the identified solution
8. Request approval from the QA Manager and Project Manager to modify the existing system
9. Document the change log with regards to this current failure case and the identified solution
10. Resolve issue

# **Tools, Techniques and Methodologies**

SQ personnel will require access to the following:

## **Software Quality Tools**

* Microsoft Office tools (i.e., Word, Excel, and PowerPoint)
* Firebase Test Lab
* Flutter
* Visual Paradigm

# **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 and hosted on cloud infrastructure 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.

[Note: If you don’t have space on the project’s server, please indicate where your records are maintained and protected. Also discuss configuration management of all deliverables (e.g., SQ Assessment Reports).]

SQ personnels will utilize cloud storage platforms such as Google Drive to collaborate to produce deliverables for the project while version control and distribution will be done on GitHub.

# **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 |

# 

# **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

# **Risk Management**

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

Every quarter, a risk assessment will be produced and to be provided to all stakeholders and there will be discussion on the necessity of adapting development practices to the risks posed.

# **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.