Software Quality Assurance (SQA) Plan

for

Cafe Bunny

SQA_PRO_002_V1.2

Version 1.2 approved

Prepared by Team Code Nation

Nanyang Technological University,

School of Computer Science & Engineering

26/03/2021

Submitted to:

Dr. Shen Zhiqi, Lab Supervisor

VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	Chew Zhi Kang	20/02/2021	Bachhas Nikita	20/02/2021	Section 8-14 1st Draft
1.1	Bachhas Nikita	20/02/2021	Chew Zhi Kang	20/02/2021	Section 1-7, 9 1st Draft
1.2	Ong Yi Shen	26/03/2021	Chew Zhi Kang	26/03/2021	Standardization

TABLE OF 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 *Cafe Bunny* 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 to identify any errors, omissions, inconsistencies, and alternatives, enhancements or improvements that can be made at any stage of development.

The *Cafe Bunny* project is aimed at cafe hoppers who like visiting different cafes on a frequent basis. It aims to allow users to record their journey of travelling and exploring different cafes. Each cafe visit will allow users to collect rewards points and they will even have the choice of designing their own bunny avatar. To create the prototype of the application, we will be using Flutter and Firebase database software. Both of these software have been created by Google and are free. Flutter (using Android Studio IDE) will be used to develop and preview the code of the application in Dart programming language while the Firebase software will be used to store and sync data between the users and the application in real time.

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
- Risk Management Plan
- Design Report on Software Maintainability
- Configuration Management Plan
- Change Management Plan
- Test Plan
- Test Case and Requirements Test Coverage Report

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 a 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:

- Account Creation
- Logging into the application
- Viewing of Google Maps on Homepage
- Viewing of Profile Page
- Avatar Selection
- Display of Cafe Information
- Changing of Colour of Icon on Map once Successfully Hopped
- Review of Cafe
- Photos of Cafe
- Viewing of Forum page
- Claiming of Rewards

3.2.2 Process Assessments

The following process assessments will be conducted by SQ personnel:

- System Requirement Analysis
- Prototyping Process
- Project Planning Process - Schedule Estimation Process and Resource Allocation
- Project Monitoring Process
- Risk Management Process
- Change Management Process
- Test Planning
- Testing 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

- Project Proposal
- Use Case Model
- System Requirement Specifications
- Project Plan
- Risk Management Plan
- Design Report on Software Maintainability
- Configuration Management Plan
- Change Management Plan
- Release Plan
- Test Plan
- Test Case and Requirement Test Coverage Report
- CMMI Level 2 Definition

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 four most important qualities for the application are Functionality, Usability, Efficiency and Maintainability.

The application has to be functional by providing the functions to meet the stated and implied needs when the software is used under specified conditions. Based on the main requirements, the *Cafe Bunny* application is considered functional when it allows the user to create an account and log into the application, shows the user a map of the cafes they have visited and not visited in different coloured icons, shows users their profile page, allows users to select an avatar, shows users the information page about cafes and allows them to post photos and reviews of the cafe and also allows them to earn rewards for every cafe they hop.

The application has high usability when it has the capability to be understood, learned, used and be attractive to the user under specified conditions. The *Cafe Bunny* application will have high usability when it is aesthetically pleasing and users enjoy the gamified functions of the application. It also has to be easily understood by users to allow them to start using the application promptly, instead of being deterred by a steep learning curve.

The application also has to be efficient, where the software is capable of providing an appropriate performance, relative to a number of resources used, under stated conditions. Since the application will be a host to a large amount of data, which includes data from the Google maps API as well as personal data of the users, the application still has to function at a reasonable speed, without causing any lags or shutting down entirely. It will also be considered efficient when reviews and photos posted by users are seen in the forum immediately and users can claim their rewards once fulfilling all the requirements.

The application is considered maintainable when it has the capability to be modified (includes corrections, improvements or adaptations of the software to changes in the environment and in the requirements and functional specification). Due to the competitive nature in today's digital world as well as the diverse types of applications available, the application created has to be made in a way where future developers can easily modify any components of the project, allowing it to adapt to the current environment. It should also be easy to remove bugs and fix technical errors without having the need to change the entire code in the future.

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:

- Length of Code
- Fan in/Fan out
- Fog Index
- Cyclomatic Complexity
- Depth of Inheritance Tree
- Number of Overriding Operations

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 - *review of resource allocation and schedule estimation*
- Requirements Analysis Review
- Use Case Model Review
- Software Design Review
- Test Plan Review
- Test Cases Review
- Prototype Acceptance Review
- Change Management Review
- Risk Management Review
- Configuration Management Review
- Release 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 personnel will review different test cases to ensure that they will provide accurate results and are directly testing the requirements of the application. 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. They will carry out both black box and white box testing to ensure every component of the application is tested. SQ personnel will review post-test execution related artifacts including test reports, test results, problem reports, updated requirements verification matrices, etc. SQ personnel will do so before every release of the application to ensure that the updates, corrections or improvements done are working accurately and do not clash with the functions present previously.

8. Problem Reporting and Corrective Action

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

QA Engineer will do testing according to the test plans. The tester will record the test done, name, and date conducted into an excel spreadsheet. If the test fails, record down the test in a separate excel sheet. SQ personnel will pass the list of failed tests to the QA manager, which will inform front-end and back-end developers to fix the bugs.

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
- GitHub
- Flutter
- Android Studio
- Firebase Database
- NTU Media Wiki
- SVN Server

10. Media Control

SQ will document deliverables in one of the following Microsoft software applications: Word, Excel, or PowerPoint. Deliverables will be in soft copy, except for completed checklists from process and product assessments. See Section 11 for more 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 team uses SVN to collaborate with teammates by having a version control system to merge changes made by other developers on the same file. The team uses MediaWiki to upload all documentation.

11. Record Collection, Maintenance, and Retention

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

The table below names the record types 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:

1. Audits and Reviews (Assessments)
2. Risk Management
3. Software Assurance
4. Configuration Management
5. Software Engineering
6. ISO 9001, ISO 9000-3
7. CMMI
8. Verification and Validation

13. 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.

SQ personnel will also identify organisational problems that may have adverse effects on project outcomes. This can be done using an adequate staffing plan by choosing team members with skill sets that are a good match for the project.

SQ personnel will also conduct frequent software risk monitoring and examples of monitoring includes publishing project status reports and include risk management issues and revising risk plans according to any major changes in project schedule.

14. SQA Plan Change Procedure and History

SQ personnel are responsible for the maintenance of this plan. SQ will update this plan throughout the life cycle of the project to reflect any changes in support levels and SQ activities. SQ personnel shall send proposed changes to the Quality Assurance Manager (QAM), along with supportive material justifying the proposed change.