

April 1, 2013

Integrated Campus

Software Quality Assurance Plan

Group 2

INTEGRATED CAMPUS | SEN-Winter 2013

Revision History:

Version	Primary Author(s)	Description of Version	Reviewed By	Date Completed
v_1.0	Ishita Agarwal, Pinky.J.Meena	This is SQAP version V_1.0	Anubhav Maity	01/04/2013

Contents

1. Overview -----	3
1.1 Purpose -----	3
1.2 Scope -----	3
1.3 References -----	3
1.4 Acronyms -----	3
2. Management -----	4
2.1 Organization -----	4
2.2 Tasks -----	4
2.3 Responsibilities -----	4
3. Documentation -----	4
3.1 Purpose -----	4
3.2 Required documents -----	4
4. Standards, practices, Conventions and metrics ---	6
4.1 Documentation Conventions -----	6
4.2 Coding Standards -----	6
4.3 Review procedure -----	6
4.4 Meeting record -----	6
4.5 Quality Checkpoints -----	7
5. Reviews and audits -----	7
5.1 Purpose -----	7
5.2 Project plan Review -----	7
5.3 Feasibility report review -----	7
5.4 System requirement review -----	7
5.5 System design review -----	7
5.6 Coding review -----	7
5.7 Corrective actions -----	8
6. Risk management -----	8
7. Test -----	8

1. INTRODUCTION

1.1. Purpose

The purpose of Software Quality Assurance Plan (SQAP) is to define the techniques, procedures, and methodologies that will be used to ensure the timely delivery of the software that meets the specified requirements within project resources.

SQA encompasses a quality management approach, effective software engineering technology (methods and tools), formal technical reviews that are applied throughout the software process, a multi tiered testing strategy, control of software documentation and the changes made to it, a procedure to ensure compliance with software development standards (when applicable), and measurement and reporting mechanisms.

1.2. Scope

The following plan will help to assure the following:

- Software development, evaluation and acceptance standards are developed, documented and followed.
- The results of software quality reviews and audits will be given to appropriate management.
- Test results adhere to acceptance standards.

1.3. References

SRS

Design document

IEEE Std 730-1998 for Software Quality Assurance Plan

Project Plan

1.4. Acronyms

Following are the acronyms used in this document:

SRS- System Requirement Specification

SQA-Software Quality Assurance

2. MANAGEMENT

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

2.1.Organization

Our teams of 10 members have been distributed different roles in the course of the project. Our team structure is Egoless so every member contributes fully and ideas to improve quality are contributed by all at any time of the project.

2.2.Tasks

The scheduling of SQA tasks is driven by the software development schedule. Therefore, an SQA task is performed in relationship to what software development activities are taking place. One or more SQA tasks can be performed concurrently until a task is completed

It is planned that the products at the end of each phase in the lifecycle of the software would be reviewed by the corresponding reviewer, and the changes suggested would be incorporated by the author of the document. In case of the code segment which is being tested, the changes would be incorporated by the author of that code segment, based on the suggestions of tester.

2.3.Responsibilities

The Team Leader will plan/direct/perform testing that meets functional configuration, Customer Environments, and boundary/error condition, testing goals, ensuring functionality, system scalability, reliability and performance goals are met.

- Planning, estimating, scheduling, deliverables and tracks dependencies, with the goal of eliminating obstacles to timely and successful product delivery.
- Determining requirements and making recommendations for test environment hardware and software needs.
- Developing quality assurance standards.

3.Documentation

3.1.Purpose

Documentation has to be maintained from the beginning so that this helps when the team proceeds with the project development and enters different phases. Any changes that need to be made can be easily done if everything is documented properly. Documentation helps in proper monitoring and quality control over the development of the software.

3.2.Required Documents

Following documents are integral part of any project and must be made so as to ensure that every Software Engineering philosophy is followed and that quality of the product is ensured. All the

documentation was done by the prescribed IEEE standards and/or from the Rational Suite Documentation Standard.

Software Requirements Specification

The Software Requirements Specification is the official statement of what the system developers should implement and it helps to provide an organized way to collect all requirements surrounding a software intensive project at the feature level into one document. The SRS includes user requirements for a system (functional and non-functional) in a language, end users understand without much technical knowledge. It also contains the system requirements (as expanded version of the user requirements) that can be used by software developers as a starting point for the system design. The SRS documentation takes care of coherence and dependability in the project.

It acts as an agreement between developers and client. The document also serves as a guide for the design phase, and will assure the client that the product will contain everything that he wants and is feasible. If the client is fully satisfied after reviewing this document, only then shall the design phase start.

To ensure that the software is a quality product, SRS should be thoroughly revised and reviewed. It must be ensured that all features of the product that are mentioned in SRS are incorporated and met with desired quality in the final product.

Project Plan

Project planning includes development of the overall project structure, the activities and the work plan/timeline that will form the basis of the project management processes employed throughout the project lifecycle. Purpose of this document is to give a complete yet concise description of purpose, goal, different criteria, phases and timings of each phase, monitoring and guidance, risk, installation, cost of the project.

Software Design Specification

The prime purpose of the Software Design Specification (SDS) is to find the best possible design within the limitations imposed by the requirements and the physical and social environment in which the system will operate.

Software configuration Management Plan

The purpose of this plan is to describe the criteria and direction necessary to perform the activities of software configuration management. It defines the steps and activities that describe how Configuration and Change Control Management is performed in the development of a software product.

The basic reason of software configuration management is to reduce the rework required for the changes and make the software consistent in order to meet the cost, quality and schedule objectives.

4. Standards, Practices and Conventions

4.1. Document Conventions

The standard IEEE Conventions for writing documents have been followed. Standard font size and style has been used throughout the document. Document has been indented properly for easier and systematic reading.

Sr. No.	Conventions	Metrics	Inference
1.	[Heading 1]	Font Size: 14 Font Type: Times New Roman Font Style: Bold	Heading
2.	[Heading 2]	Font Size: 12 Font Type: Times New Roman Font Style: Bold	Sub-Heading
3.	[List Paragraph]	Font Size: 12 Font Type: Times New Roman Font Style: Regular	Body

4.2. Coding Standards

Have been listed in a separate document named Coding Standards.

4.3. Review procedure

Each and every document in our documentation is reviewed by the reviewer and if required then changes are recommended and the same are incorporated by the author. Review log is maintained for each and every document. The changes are made according to configuration management plan.

4.4. Meeting Record

The group meets two times in week to check the progress of project in addition of official labs for regular monitoring. A meeting record is maintained for all the people present and absent in the meetings.

4.5. Quality Check Points

There would be quality checkpoints at end of each phase. These checkpoints will guarantee that deliverables are of acceptable quality. It is planned that at the end of each phase of software lifecycle, the documents would be reviewed by reviewers, and required changes would be incorporated by respective author of that document.

It also planned that a review list (report) would be made, which would contain the details of every review along with the names of the members, so that the origin of a change can be traced back.

5. Reviews and Audits

5.1. Purpose

This section shall

- Define the technical and managerial reviews to be conducted
- State how the reviews are to be accomplished
- State what further actions are required and how they are to be implemented and verified.

5.2. Project Plan Review

The project plan is reviewed in order to ensure that the project is on the right path i.e. according to plan and meeting the deadlines.

5.3. Feasibility Report Review

The feasibility report is reviewed in order to confirm that the successful execution as well as completion of the project is feasible and that the planning done earlier in the project is correct and reliable.

5.4. Software Requirements Review

The software requirement is reviewed to ensure that the requirements specified by the client in the functional requirements have been taken care of.

5.5. System design Review

The Technical Design review is held to evaluate the technical adequacy of the software design and the acceptability of the design to satisfy the functional requirements.

5.6. Coding Review

The coding review is done at the end of the project to assess the features implemented in the software and recommends appropriate action to be taken. It is done in two ways:

1. Walkthroughs – This is an informal technique of reviewing and is based on common sense.
2. Inceptions – This is a formal technique of reviewing and checks for the correctness of individual features developed during the course of the project.

5.7. Corrective actions

Problem reporting and actions are very important to achieve an error/bug free system. The ultimate target of this is to establish a process that continuously improves the safety and reliability of the system. When a document is reviewed, its quality and correctness is checked and if any changes or modifications are required, they are made accordingly. Even at later stage of the project development, we might discover additions or modifications needed in a document and thus another version of it is released and the changes or modification needed are penned in the review document.

6. Risk Management

Risk management is an attempt to minimize the chances of failure caused by unplanned events. The aim of risk management is to minimize the impact of risks on the cost, quality and schedule of the project that is undertaken. This document describes the risks that the team may encounter while designing and managing the project and the strategies and actions that the team will be conducting to mitigate these risks. The document also specifies the methods of managing any risk.

Certain activities must be performed to ensure the qualities of the product are:

- Maintenance of documentation standard: All documents must be in proper format and completed on time.
- Inspection and reviews of all the documents: Every document is reviewed by some other team member and changes suggested. These changes are then incorporated according to the configuration management plans.
- Verification and Validation: Verification activity is done at the end of every phase. It checks that where the present phase is complemented by the assumptions made in the previous phase.
- Coding guidelines followed in the whole project coding phase: This makes the software code easy to understand and to maintain.

7. Test

Testing is the process of analyzing a software item to detect the differences between existing and required conditions and to evaluate the features of the software item. The plan identifies the items to be tested, the features to be tested and the type of testing to be performed. The main purpose is to ensure that all the modules are tested according to the specifications described in the SRS and that all the modules of this project are working properly.

The type of testing we use in this project are:

System testing

It is done to assure that the system meets the full requirements, including quality requirements (non-functional requirements). At the end of test phase, the software should fit the metrics for each quality requirement and should satisfy the use case scenarios and maintain the quality of the product. At the end of the project development cycle, the developer/owner of this application should find that the project has met or exceeded all of their expectations as detailed in the requirements.