Software Quality Assurance Plan Version 1.0

Team-01 : DEngima

October 2017

Contents

1 Overview

The Software Quality Assurance Plan defines measures in terms of practices, tools and conduct in order to deliver a high quality product. It provides a structured guideline to not only assure timely delivery of the software that meets specified requirements within project resources, but also to ensure that the delivered work product is of good build. The rest of the document is organized as follows:

- **Management**: A description of each major element of the organization and a description of the SQA tasks and their relationships
- **Documentation**: Identification of the documents related to development, verification, validation, use and maintenance of the software.
- **Quality Assurance Requirements**: This section defines the quality assurance review, reporting, and auditing procedures used to ensure that software deliverables are developed in accordance with this plan and the project's requirements.

1.1 Intended Audience

This document is meant for the software development team, to use it as a guide while developing the software which is quality assured. Through this document even client can get a rough idea of the activities that are performed to ensure the quality of the product that will be developed at the end of the SDLC[1].

2 Introduction

2.1 Purpose

The intended purpose of the Software Quality Assurance Plan is to set forth the process, methods, standards, and procedures that will be used to build a high quality working product that meets all the specified requirements in the *SRS* for the *Online Pharmacy* project. Any practices or conducts that appear to be in conflict this document are notified to the development so that they may adapt to the ones specified here. This will not only ensure quality but will also help bringing uniformity through out the project development life cycle.

2.2 Scope

The scope of this document is to outline all procedures, techniques and tools to be used for quality assurance of this project.
This plan assures the following:

- Identify all the responsibilities of the project developing team in terms of quality.
- Makes sure to have a brief idea on quality assurance work procedures.
- To develop a quality approach to quality management.
- To assure all of the SDLC is properly and timely documented.
- Any changes to the procedures or documents after reviews are to be recorded in a log and should be reflected respectively.

[1]Software Development Life Cycle

3 Quality management

3.1 Organization

The Team Head will however conduct inspections, reviews, and walk-through on a regular basis. In addition to this, members will be assigned to review the documents of each phase before moving ahead. All the documentation is shared on a Cloud database with everyone free to access the documents for reference at any point of time. After reviewing, an analysis of the current phase is done to determine any problems. Changes and suggestions will be used in places where quality decisions need to out-weigh development schedule decisions.

3.2 Tasks and Responsibilities

The responsibilities of the development are as follows:

- Develop the design plan and test plan for testing the system.
- Planning, coordinating, testing and assessing all aspects of quality issues.
- Implement and test the application and deliver the application along with the necessary User Manuals, Installation guides, Self Test guides, etc.
- Review the work performed by each other and provide feedback, change suggestions and advice.

4 Documentation

Documentation is an important and integral part of both Software Development Cycle and Quality Assurance Management. All the work that is done in each and every phase is documented and preserved. This works very well with any member to get on track with the rest of the team by simply referring to the documentation. For large scale projects, documentation is the ultimate reference guide to sort out any confusion or conflict that might creep in during development.
In addition to this document, the essential documentation to assess quality in development, verification, validation, use and maintenance of the software are as follows:

- Feasibility Report
- Project Proposal
- Project Plan.
- System Requirement Specifications.
- User Manual

- Design Document
- Configuration Management Plan
- Quality Assurance Plan
- Risk Management Plan
- System Test Plan
- Gantt Chart
- Cost Estimation Document
- Test Report
- Deployment Plan

4.1 Document Conventions

All the Documentation is to be done using LATEX typesetting language. All documentation is done in IEEE Article 12pt format to ensure uniformity, clarity and editability of the documents. Multiple members review the document and the versions of the documents are maintained so that changes in the documents can be tracked and whenever required it will be easy to review all the changes done.

4.2 Coding Conventions

Coding conventions are defined so that the code is readable, easy to debug and maintainable.The project mainly follows the PEP 8 style coding convention for Python.

5 Software Quality Assurance Requirements

5.1 Quality Assurance Procedure

Every document has a set of authors responsible for making it and a set of reviewers, none of whom is part of the authors for critically reviewing the document for it's ambiguity, understand-ability, correctness and degree of completeness. The Authors are responsible to make sure that they write the documents such that the purpose and scope are clearly mentioned and the document does impart according to it's name and meet the expected requirements.

The responsibility of the reviewers is to ensure that the document is well-suited for the target audience, fulfills expectation and provide edits, additions and feedback to the authors. All the suggestions of the reviewing team are to be incorporated into the document and passed on for reviewing again until the document is green flagged by the review team. The reviewing team handles the

responsibility to maintain the quality of the documentation.
The above procedure is not only for documents it is meant for all the different phases like coding, testing, etc.. In case of coding, the testing team should verify the errors and mistakes in the application and the developers team should make to verify them all. The most important task of the testing team is to find ways in which the application can fails in delivering the result.

5.2 Review

Every document that is made during the software development plan is discussed with the team members, reviewed and then that is documented, too. Th assigned members to review are responsible to make sure that the document is in accordance with the document conventions and is easy to understand and interpret unambiguously.In case of changes the modified document should be uploaded within the deadline given and that should be reviewed too.
All codes will also be peer reviewed and tested for readability, modifiability, quality and performance. Necessary steps will be taken to adhere to prescribed coding conventions and standards. This is intended for ease of debugging, ease of access to other members.
The code developed during the coding phase should pass all the testing phase and then, the software is deployed. If there are any bugs detected after the product is launched then the testing should verify those bugs, the developing team should make sure that there are no similar kind of bugs in the next update.

5.3 Audits

Documents and codes are audited periodically to ensure that they are following the prescribed quality assurance methods. they determine if there are any places where the quality is undermined and what changes in practices will help raising the bar of quality. Every assigned member is required to report back on any changes, developments and updates in his/ her work every week and he/she should make sure their work fits well with the quality assurance standards and meet all the requirements.

5.4 Verification and Validation

Verification

The process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.
The following activities are meant to be done for the verification process:
1. Evaluate SRS requirements and relationships once again.
2. Evaluate all the documents and check whether the purpose of that document is verified.
3.

Validation

The process of evaluating software during or at the end of the development process to determine whether the software solution produced is compatible with the user's needs. Thus following activities are carried out during validation:

Each document produced must be accepted by each and every member of the group.
The code developed for this web solution should pass all the test cases.
The criteria for acceptance is:

- How does the product works under stress conditions and at boundary conditions.
- How accurate and adequate is the user documentation?
- All the requirements are fulfilled.

5.5 Testing

There is a possibility of human mistakes and error associated with every piece of software. Software Testing aims at discovering and rectifying all these errors in the code in such a way that they meet the requirements specified. Testing is done for all phases in software development life cycle. For our project, we run testing side by side with development. Before moving on to the next stage, current stage is tested, validated and verified. Testing checks that the outputs to correct inputs are in accordance with the specified requirements, and also how the product reacts to incorrect inputs. This entire process helps in identifying the potential risks and issues regarding the product that might occur and to

nd appropriate ways in which they can be dealt. In case of any kind of change in any phase due to the faults occurred will be documented with appropriate changes conveyed by the respective team members, with the assurance of high levels of quality and considering the ability and efficiency of the team in dealing with particular change.

6 Making of the software solution

Our approach for making this software solution includes steps like:

- The first and important step is to make a basic layout of the application
- We made sure that all the requirements mentioned in the SRS are met.
- The primary features of this application mentioned in SRS are all real time working
- We made sure that the primary features are all usable for the users.

- We will add all the other functionalities (secondary features) and make sure that each of these functionalities are usable when they are added to the software.
- Once a publishable version is ready, we will make sure that the users will start enjoying while working with our application.
- We will make sure that all the other functionalities which are added later on are usable and pleasurable for the users.