Lab 2: Software Quality Management

1. OBJECTIVE

1) Setup quality assurance for the project development.
2) Formalize quality processes.
3) Design the prototype system to confirm the requirements of the proposed system.

2. LABORATORY

This lab will be conducted in the Software Lab 3 (N4-B1C-14) in SCSE.

3. EQUIPMENT

- PCs (Windows OS)
- Version Control System, e.g., Github

4. INTRODUCTION and SCOPE

In the previous lab, your team has proposed your software development project and should have obtained your lab supervisor's approval on the project. In this lab, the scope is for you to hold in-class *d*iscussions on the key points concerning planning for the quality of the software product you have proposed and document the key points in the meeting minutes. You should also start discussing what prototype you are building for the product including the main features, basic performance considerations and the tools to be used to support development. After the lab, you shall do further study and finalize your quality management strategy in your Quality Plan. You may choose to start building the actual prototype at this time.

You should emphasize a quality development ***process***, in the context of your development project.

As required by the Verification Process Area, peer reviews (review meetings) are to be planned. The QA team will follow up on the required action and ensure that the issues raised are addressed and closed.

All of the processes and workflow for quality management should be formalized and described in the Quality Plan.

Prototype system can help to refine and validate the requirement on the entire product. High level design should be conducted in this lab for the preparing the project plan thereafter.

5. EXPERIMENT

5.1 Hold the team meeting and complete the meeting minutes

In the meeting, the deliverables of this lab are discussed. Tasks and deadlines are identified and the tasks should be assigned. By end of the lab session, the minutes should be shown to the lab supervisor and submitted to the Wiki. In the minutes, the basic description of the software, hardware and other parts of the proposed system should be addressed and a list of points on quality management should be described. In addition, the plan for developing the prototype system should be briefly introduced.

5.2 Complete System Requirement Analysis

Given the requirements of the system, determine your goals, objectives and strategies:-

- What is the main purpose and direction of the application/system?
- What are the critical constraints of the application/system?
- What are the major features of the application/system?

Use the use case model to help you to capture high-level functional requirements and refine them further. A prototype system can help you to estimate the complexity and required technologies of the functionalities and features of your system. You can start your initial design of your prototype system based on the use case model before the implementation in the next lab.

System requirements are to be specified in such a way that they are verifiable / testable. Requirements must also be trace-able from their expression in a system requirement specification to their realization in software requirement specifications, design documentation, source code and test cases.

5.3 Start Building a Prototype

Build a prototype using a development framework (IDE). A prototype system should briefly demonstrate the main features of your system. In addition, it should be a working system including frontend and backend components, if any.

The development of the prototype shall begin with the most distinguishing feature/component of your product. The development work must be in a collaborative way by at least three developers.

Along the way, use the prototype to validate the correctness of the requirements, seeking clarification from the customer, if necessary. While building the prototype, the development team must also validate that the customer requirements can be met in the target environment.

The prototyping process can also be used to address technical risks, such as developers' insufficient understanding of the Development Framework, the tools and the best practices that you are about to adopt in the development, and the target environment constraints.

5.4 Prepare Quality Management

The quality models learned from lecture and quality management on different stage of the development lifecycle should be discussed in the team. The quality plan and practices on different stage of the development lifecycle should be documented.

The review meetings shall be arranged. Review results shall be documented and followed through.

6. DELIVERABLES

The following items should be delivered by the end of the in-class lab session:

1. Meeting minutes of the in-class discussion

Have the following ready (hosted/uploaded/compiled) by the next lab session:

2. Wiki: System requirement specification
3. Wiki: Quality Plan
4. Wiki: Backlog
5. Wiki: Meeting minutes