CO255 - Software Engineering Lab
Assignment 2

Date of submission : 22/01/2018

Project Members

1. Ashwin Joisa
   Roll number : 16CO104
   Mobile number : +91 9482851775
   Email id : ashwinjoisa@gmail.com

2. Praveen Raj
   Roll number : 16CO115
   Mobile number : +91 9481276415
   Email id : hlpr98@gmail.com

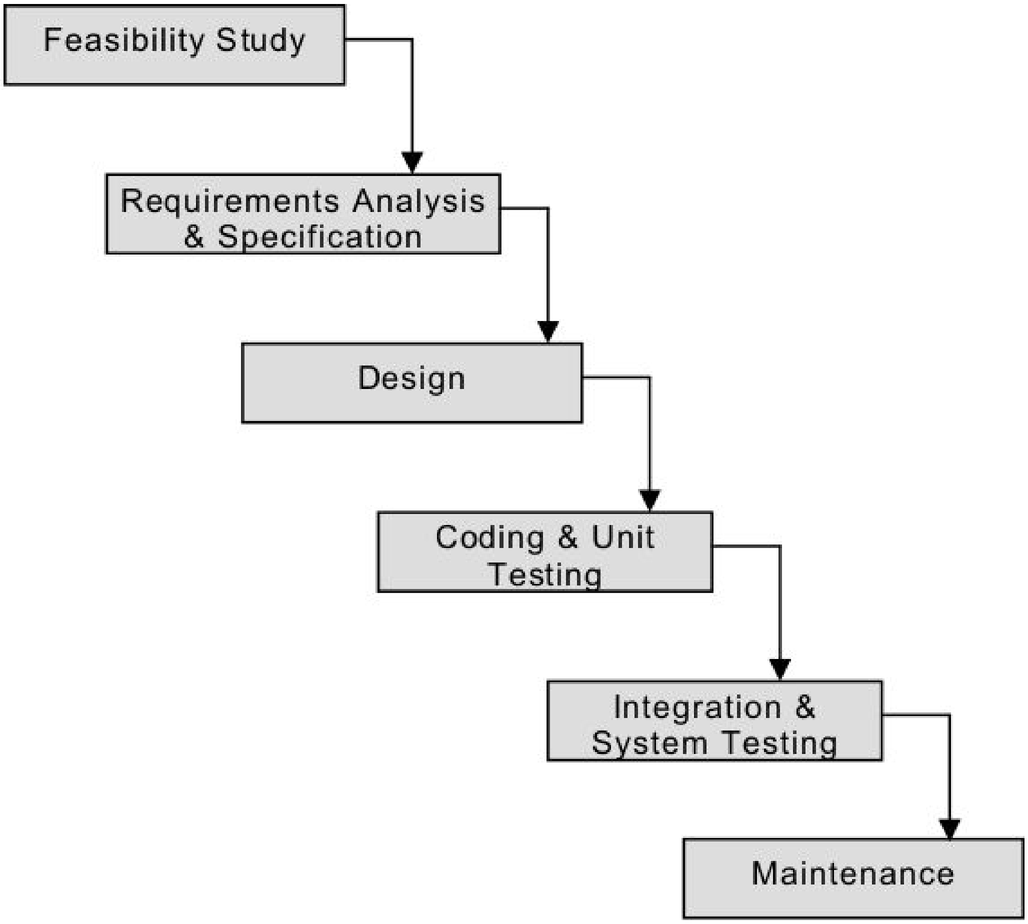
Project : Automated Lab Program Evaluator

Question 1: Identify (and justify) the process model suitable for your chosen problem / project. Explain the steps / stages in the process model that would be suitable for your problem solving with suitable diagrams, if any, in the documentation.

The process model suitable for our chosen project is the **Classic Waterfall Model**. The requirements of this problem are reasonably well understood, and work flows from communication through deployment in a reasonably linear fashion, since this project tries to bring about well defined adaptations or enhancements to the existing system.

Stages involved in the classic waterfall model:

1. Feasibility Study
2. Requirements Analysis and Specification
3. Design
4. Coding and Unit Testing
5. Integration and System Testing
6. Maintenance


Feasibility Study
Requirements Analysis & Specification
Design
Coding & Unit Testing
Integration & System Testing
Maintenance

Feasibility Study

The main aim of feasibility study is to determine whether it would be financially and technically feasible to develop the product.

- At first we try to have a rough understanding of what is required to be done by visiting the client side. We study different input data to the system and output data to be produced by the system. We study what kind of processing is needed to be done on these data and they look at the various constraints on the behavior of the system.
- After we have an overall understanding of the problem we investigate the different solutions that are possible. Then we examine each of the solutions in terms of what kind of resources required, what would be the cost of development and what would be the development time for each solution.
- Based on this analysis we pick the best solution and determine whether the solution is feasible financially and technically.

Requirements Analysis and Specification

The aim of the requirements analysis and specification phase is to understand the exact requirements of the customer and to document them properly. This stage consists of two distinct activities, namely:

- Requirements gathering and analysis, and
- Requirements specification

The goal of the requirements gathering activity is to collect all relevant information from the customer regarding the product to be developed. This is done to clearly understand the customer requirements so that incompleteness and inconsistencies are removed. After all ambiguities, inconsistencies, and incompleteness have been resolved and all the requirements properly understood, the requirements specification activity starts. During this activity, the user requirements are systematically organized into a Software Requirements Specification (SRS) document. The customer requirements identified during the requirements gathering and analysis activity are organized into a SRS document. The important components of this document are functional requirements, the nonfunctional requirements, and the goals of implementation.

Design

The goal of the design phase is to transform the requirements specified in the SRS document into a structure that is suitable for implementation in some programming language. Two distinctly different approaches are available: the traditional design approach and the object-oriented design approach. We will be following the **Object-oriented design approach**. In this technique, various objects that occur in the problem domain and the solution domain are first identified, and the different relationships that exist among these objects are identified. The object structure is further refined to obtain the detailed design.

Coding and Unit Testing

The purpose of the coding and unit testing phase (sometimes called the implementation phase) of software development is to translate the software design into source code. Each component of the design is implemented as a program module. The end-product of this phase is a set of program modules that have been individually tested. During this phase, each module is unit tested to determine the correct working of all the individual modules. It involves testing each module in isolation as this is the most efficient way to debug the errors identified at this stage.

Integration and System Testing

Integration of different modules is undertaken once they have been coded and unit tested. During the integration and system testing phase, the modules are integrated in a planned manner. During each integration step, the partially integrated system is tested and set of previously planned modules are added to it. Finally, when all the modules have been successfully integrated and tested, system testing is carried out. The goal of system testing is to ensure that the developed system conforms to its requirements laid out in the SRS document.

Maintenance

Maintenance of a typical software product requires much more than the effort necessary to develop the product itself. Maintenance involves performing any one or more of the following three kinds of activities:

- Correcting errors that were not discovered during the product development phase. This is called corrective maintenance.
- Improving the implementation of the system, and enhancing the functionalities of the system according to the customer's requirements. This is called perfective maintenance.
- Porting the software to work in a new environment. For example, porting may be required to get the software to work on a new computer platform or with a new operating system. This is called adaptive maintenance.

Question 2 : Identify a set of tools / CASE tools / software that need to be used to do the follow-up of the process model that you have chosen. Enlist all such tools / set of tools required to complete the SW Engg lab work - select tools only from open-source domain (sample list is provided in the class).

Requirement Management Tools:-

1. rmToo

Design Tools:-

1. LucidCraft Drawing tool.
2. ERD Plus.

Development Tools:-

1. Integrated Development Environments:-
    - C++ IDE, a package in, Atom Text Editor.

2. Database Tools:-
    - MySQL

3. Build Tools:-
    - GNU g++ compiler.
    - GNU gcc compiler.

Testing Tools:-

1. CppUnit

Project Management Tools:-

1. LibrePlan