CSC 7400
Assignment1: SW Processes
Student: Lina Mi

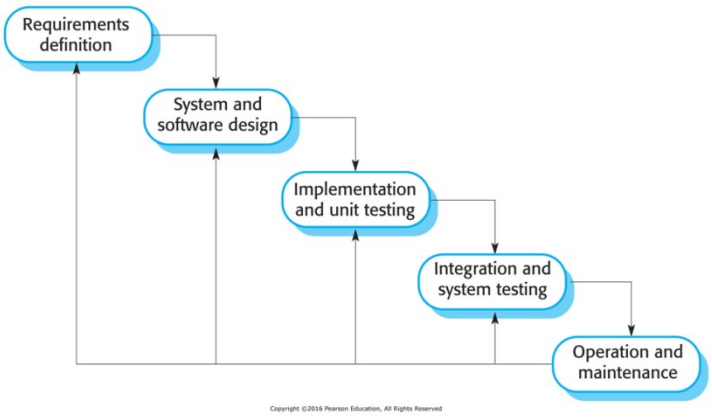
1. What is the most important difference between generic software product development and custom software development? What might this mean in practice for users of generic software products?

   Answer: generic software development is producing stand-alone systems which are developed by development organization and sold on open market to any customer who is able to purchase them, such as Microsoft Office; Custom software development focus on developing the software systems that is for particular customers, for example the control system for food processing industry.

   The most important difference between generic software development and custom software development is that for generic software the development organization, not customers, decides the performance and functionalities of the software; for custom software development, however, the customer has the authority to control the software specification.

   The essential difference between generic software development and custom software development indicates that for the users of generic software product, they have no authority to control over the software specification and then cannot control the development of the product. The developer decide what performance and functionalities the software would provide. This could result that the generic software could not be able to practically meet some special requirements of particular users.

2. Consider the waterfall model as shown in the following figure. Why it is called a waterfall model?



   Answer:

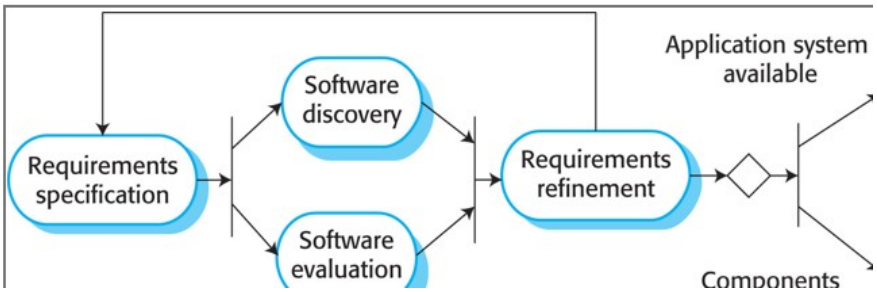
   According to the above figure of waterfall model, it can be seen that this model takes fundamental process activities of specification, development, validation and evolution and represents them as separate phases, such as requirements specification, software design, implementation, testing, and the processes are implemented one by one, namely after

requirement definition finished, the process moves to next stage of system and software design, this process continues until reach the final step of operation and maintenance. just like a cascade falls down from top of hill, flowing from one step to another.

3. What is an incremental software development process?

   Answer: In an incremental software development process, the activities of specification, development and validation are interleaves together, and a series of versions of software are developed with each version adding functionality to previous version (increments).

4. Consider the integration and configuration process model as shown in the following figure. Explain why it is essential to repeat the requirements engineering activity in the process.



   Answer:

   In the above given figure it is essential to repeat the requirements engineering activity in the process because after the initial requirements specification are finished, the development phases move to next stage of software discovery and software evaluation. During this stage, some of requirements are found to be inappropriate and new specifications and new requirements are developed, and that lead to the demand of modifying original requirements specification, so it is indispensible to repeat the activity of requirement engineering to make modifications on the requirements using information about the reusable components and applications that have been discovered. Requirements are also modified to reflect the available components.

5. Why it is important to make a distinction between developing the user requirements and developing system requirements in the requirements engineering process?

   Answer:

   The user requirements are abstract statements of the system requirements for the end-user user of the system and customers. User requirements are the final goal of the software system, specifying the functionalities and performances the system should provide to users. User requirements should be generally expressed in native language and mainly used for end users. System requirements, however, are developed based on user requirement, they are expressed in much more detail than the user requirements and

mainly used by system developers. Making distinction between developing the user requirements and developing system requirements can help system developer focus on developing system with explicit system requirements as guide.

6. Suggest two advantages and two disadvantages of the approach to process maturity that is embodied in the SEI's Capability Maturity Framework.

   two advantages:

   1) Higher Productivity
   2) Higher Cost efficiency

   two disadvantages:

   1) Harder to implementation (may need more time and effort)
   2) It is hard to make improvement on system for small-size organization,

7. Explain how the principles underlying agile methods lead to the accelerated development and deployment of software.

   Answer:

   The principles in agile method are described as following

   a. **Customer Involvement**
      This principle guarantees that customer can be closely involved throughout the development of process, their role is provide and prioritize new system requirements and to evaluate the iterations of the system, thus developer of system can get the feedback from customer quickly and make corresponding modifications alongside the process of development , rather than making change in last stage of development.

   b. **Embrace Change**
      This principle expects system requirement to change, and then these changes are accommodated in the process of system design, so the new changes in requirements can be taken in account in time, thus accelerate the development of system.

   c. **Incremental delivery**

      With this principle, software are developed in increments, with customer specifying the requirement to be included in each increment. So this principle help to get fast deployment of system

   d. Maintain simplicity.
      This principle lead to the simplicity of in both the software being developed and in the development process. So it is much easier for Agile developers to develop system and make change when necessary.

e. **People, not process**
The skills of the development team should be recognized and exploited. Team members should have the freedom of choosing their own ways of working, thus this principle facilitates the development of software system

According to the previous explanation, it can be concluded that all of previous principles in agile software development are helpful to accelerate the development and deployment of software.

8. Extreme programming expresses user requirements as stories, with each story written on a card. Discuss the advantages and disadvantages of this approach to requirements description.

Answer: Advantages of this approach of requirements description:

a. Easier to integrate requirements elicitation with development .
b. Allowing system customer works closely with the development of team and discuss scenarios with other team members.
c. The user cards can be break down into tasks and easier to estimate the effort and resources required for implementing each task.
d. Fast development of software with new useful functionalities.
e. It can accommodate the changes in a fast pace.
f. Easier get user involved in suggesting requirement during initial predevelopment requirements elicitation active.
g. Easier to relate to the stories.

. Disadvantages of this approach:

a. Incompleteness. Hard to judge if the user stories are enough to cover all of essential requirement.
b. Not good reliability. Difficult to tell if each story give an true picture of an activity.

9. Why it is necessary to introduce some methods and documentation from plan-based approaches when scaling agile methods to larger projects that are developed by distributed development teams.

Answer:
when scaling agile methods to larger projects that are developed by distributed development teams, it is necessary to introduce some methods and documentation from plan-based approaches because no single model is appropriate for all large-scale agile products since the type of product the customer requirements and the people available are all different. The other reason is the principle underlying agile method are sometime difficult to realize in practice.

10. It has been suggested that one of the problems of having a user closely involved with a software development team is that they ‘go native’. That is, they adopt the outlook of the

development team and lose sight of the needs of their user colleagues. Suggest three ways how you might avoid this problem and discuss the advantages and disadvantages of each approach.

Answer:

a. Having more users involved in the development team: the advantages of this method is that we could get different opinions on the system development and may provide many useful prospective for development team. Disadvantages of this method are that some time extra users are not available or too expensive to have, the other disadvantage is that there may be confliction among the opinions of multiple users, this confliction may lead confusion for development team.
b. Periodically Changing the user: Advantage of this user is that introducing new customer may provide new perspective for system development. Disadvantage is that it will take time, sometimes even long time, to familiarize the new user with current situation of system development and extend the development cycle of system.
c. Having other users to assessing the suggestions of current user from time to time: Advantage of this approach is that other users may "pull out" the current user from deep involvement in the development team by providing fresh prospective. Disadvantage of this method is that different evaluators may have different conclusions and sometimes conflicting opinions may resulted. Furthermore frequent evaluation may result in the extension of development cycle of system.