

# Kevin R Minn

## Assignment 2

### 1. Explain what the agile methods are, the aim of agile methods, and the principles of agile methods.

Agile methods develop the system in increments by using the incremental development process. these methods made the change of the system in the small increments depending on the requirements of the customers. In this, the customers are included in the development process. the agile methods focus on the software of the system instead of designing and documentation. These methods allow the developers to deliver the software quickly to the customer.

The principle of agile methods for the development and employment of software:

1. Incremental delivery.
2. Customer involvement.
3. People do not process.
4. Embrace change.
5. Maintain simplicity.

### 2. What is:

- a) **The test-first development:** Test first development requires there to be a clear relationship between system requirements and code implementation.
- b) **The test-driven development:** software development approach that develop cycles created and these are repeated over and over again. These requirements are turned into test cases and hence resulting in overall development of the software, ultimately results in the passing of test. In this, you cannot add extra software that does not match the requirements of the user.
- c) **A sprint in scrum:** sprints make projects more manageable, allow teams to ship high-quality work faster and more frequently and gives them more flexibility to adapt to change. Sprints are at the very heart of scrum and agile methodologies.
- d) **Scrum:** scrum is most used to manage complex software and product development using iterative and incremental practices.it increase the productivity, it provides better estimates while spending less time creating them.

**3. 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.**

**Extreme programming advantages:**

- Scenarios cope with most common operation. It is easy to identify what type of operation is required in the user stories.
- Customers focus on the scenario card increase the chance that the software produced will actually meet the needs of the users.

**Extreme programming disadvantages:**

- Using scenarios on a card can bring to a function overlooked or omission which can be a time-consuming process to complete the system.
- Two different scenarios can lead to the same function as it will be conflicting each other. Crossing out redundant scenarios can be a cumbersome task.

**4. Discover ambiguities or omissions in the following statement of requirements for part of a ticket-issuing system:**

**An automated ticket machine sells rail tickets. Users select their destination and input a credit card and a personal identification number. The rail ticket is issued, and their credit card account charged. When the user presses the start button, a menu display of potential destinations is activated, along with a message to the user to select a destination and the type of ticket required. Once a destination has been selected, the ticket price is displayed, and customers are asked to input their credit card. Its validity is checked, and the user is then asked to input their personal identifier (PIN). When the credit transaction has been validated, the ticket is issued.**

Ambiguities and omission include:

- Does it select multiple destinations same time?
- Tickets need to book either single person or multiple people on same time to destination?
- Destination selected is wrong then is it possible to cancel or go back?
- Suppose Invalid credit card used what will happen?
- Is it swipe or insert the credit card?
- If entered incorrect personal identification numbers, then what happens?
- Sufficient balance not there in credit cards then what will I give?
- Is it manually entering the numbers?
- Ticket Prices are varying for Adults/child?
- Is it their Rooms are Trains?
- Is it print access is there or not?

**5. Why is it 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.**

Plan-based approach of agile methodology:

The point to show the reason for including documentation and methods for a large project developed using plan-based approach in a distributed development environment are:

- If documentation is not provided, then, it becomes very difficult for the team members to understand the entire project.
- To documentation the identified progress issues obtained during monitoring of scrum program, documentation and methods for large projects development are required.
- The problem of learning, planning, and language can be identified on easy bases.
- The test automation with continuous amalgamation can be followed by the scrum team for large project development.

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

Agile method develops the system in increments by using the incremental development process. these methods made the change of the system in the small increments depending on the requirements for the customer. In this, the customers are included in the development process. the agile methods focus on the software of the system instead of designing and documentation. These methods allow the developers to deliver the software quickly to the customer.

## **7. What is the difference between plan-driven and agile processes?**

In the agile approach the iteration is done as per the activities hence the requirement, as well as designs, are being developed instead of individual basis whereas a plan-driven software process is not mostly the waterfall model but is a plan driven, incremental development as well as delivery is most possible.

### **Agile project planning:**

This is an agile, knowledgeable, collocated as well as collaborative. Has dedicated, knowledgeable, representative as well as empowered which is largely emergent as well as has a rapid change which is designed for a current requirement.

### **Plan-driven attribute:**

This has a plan-oriented, with adequate skills which help in accessing the external knowledge.

This is collaborative, representative as well as empowered by the customers. This process is highly stable and is designed for current requirements.