Akshay Chaturvedi

2017UCO1690

COE 3

**EPP Assessment Assignment**

**Q). Write short notes on the following:**

**(i) Scrum**

(i) Scrum is a framework which can be utilised by an organisation to increase teamwork and enhance the learning experience of the different users. When referring to Scrum, one generally thinks about software development. However, its principles can be applied to all types of situations involving teamwork. Scrum is an agile project management framework that describes the roles and responsibilities of teammates in order to structure and manage the work related to a project.

The Scrum framework is very simple and easy to understand. Instead of implementing a traditional algorithmic approach, Scrum replaces it with a heuristic approach. One of the many reasons for Scum’s success is that it takes into consideration that the team might not know everything about a project initially, and will develop knowledge along the way. It is structured in such a way that aids teams to naturally adapt to challenging problems, while also focussing on short term releases so that teams can learn and improve.

There are three essential roles defined for the success of a Scrum project. The roles are, product owner, Scrum master, development team. The product owner’s main responsibility is to ensure that the development team adds the most possible value to the business. An effective product owner bridges the gap between the business and the development team by clearly communicating the features that need to be implemented to maximize customer satisfaction and revenue. It is of paramount importance that the product owner not only understands the business, but also the customer needs and wants. The Scrum master can be thought of as the project lead. The Scrum master focusses on coaching the development team on practices that make their approach more transparent and sustainable. An effective Scrum master knows every detail about the working of the development team and comes up with ways to increase their efficiency. As Scrum is cross-functional, the development team consists of not only developers but also testers, UI/UX designers, and operation engineers. The development team is assigned a new project which they complete under the Scrum master. An effective dev team updates the Scrum master regularly about the project’s progress, and also analyzes each of the new releases (sprints) to better the next release.

**(ii) Lean Development**

Lean development is the process of applying lean manufacturing principles to software development. Lean manufacturing focuses on optimizing the assembly line to minimize waste, and maximize product quality. The goals of software development align with that of lean manufacturing. Another similarity is that software development follows an iterative process which releases prototypes in a short amount of time and that both manufacturing and software development require the collaboration of specialized workers. There are seven lean development principles namely, waste elimination, quality enhancement, knowledge creation, delayed decision making, fast delivery, respect, and optimization as a whole. Waste elimination refers to the removal of all types of situations that may hinder the progress of a project. Removal of unnecessary code can greatly reduce the resource spent in testing. Unnecessary functionalities should also be removed as they add complexity to the code base while adding no real value. Quality enhancement is the process of gradually increasing the quality of the product without producing extra waste. Often it is seen that when organizations perform quality enhancement, extra waste is generated in the form of unnecessary testing, or excessive logging of defects. A popular method used to tackle this issue is pair programming. Pair programming involves using two developers instead of one for feature delivery. Writing criteria for code can also help, as the developers know exactly what to do, and testing time is also reduced. Knowledge creation and expansion is an important principle of lean development. When a developer works on a new project, he/she needs to study the topic. This in turn, expands the knowledge base of the developer, and this new gained knowledge can then be applied to other projects, saving time and money. Delayed decision making reduces the overall cost of implementing changes. If one delays the release of a project, he/she has more time to test and ensure the product meets the expected quality standards. Fast delivery of the product is desired by every customer or client. Lean development focuses on placing the product in the customers’ hands as soon as possible, and incrementally enhancing it using customer feedback. The respect aspect of lean development is often overlooked, but can be the reason for the downfall of a project if neglected. In the fast paced and result driven environment of software development one might not treat their juniors with respect if they fail to deliver on the deadline. This creates animosity within the teams, which in turn result in the downfall of the project in hand. Optimizing as a whole acts as an antidote to all the vicious cycles caused by suboptimization. The first step is to optimize the value streams. Once the value stream is known and explained thoroughly, the next step is to divide the teams such that they are self sufficient and they don’t need to reference any other team related to work at hand.

**(iii) Extreme Programming (XP)**

Extreme programming is an agile development methodology which focuses on making superior quality products with respect to ever changing dynamic customer requirements. There are 5 steps to XP. The first step is planning. Planning involves the setting up of goals for the entire project. This can be done by first inviting customers to tell their vision about the ‘perfect product’ and then to break down and prioritize the vision in the release plan. After planning, the designing phase begins. Firstly the main features of the project are decided upon. Then, a simple structure is created of the least complexity to accurately model the minimum set requirements. The third step is coding and refactoring. The developers first implement the design using the most simple programming practices. Then refactoring is performed. Refactoring code is the process of rearranging the code in the optimal way without changing the external behaviour. Testing is done side by side while developing the product., instead of after. Testing the code on the test cases defined in the planning stage allows the developers to analyze the shortcomings of the code and come up with solutions. The final step of XP involves listening to customer feedback. Customer feedback is really important as it helps the developers to analyze how their product is doing in the outside world, while also incrementally implementing customer suggestions.

**(iv) Adaptive Software Development (ASD)**

Adaptive software development (ASD) is a method to develop complex software with an emphasis on teamwork and self-organization. There are three phases in the ASD life cycle namely, speculation, collaboration, and learning. The speculation phase is the planning phase in the ASD life cycle. In this phase a project plan is built on the basis of the user and project requirements, customer vision etc. The project plan is a document which shows each step that needs to be taken in order to reach the finished product. Based on the project plan, a release plan is also created which is maintained to show the expected release date for various releases. In the collaboration phase, professionals of different specializations interact with each other to come up with creative ideas and help execute the project plan. A strong emphasis on individuality is also placed to motivate individuals to come up with creative ideas. The last phase of the ASD life cycle is learning. A developer might not be able to work with a new technology, but after gaining experience with it, the new technology can be exploited by the developer to develop new projects. ASD’s focus on collaboration, individuality, and creativity make projects much more likely to succeed.

**(v) Feature Driven Development**

Feature driven development is an agile framework which focuses on the development of particular features separately and then combining them to create a whole project.

There are five steps to feature driven development:

* Develop an overall model
* Build a features list
* Plan feature
* Design feature
* Build features

There are several advantages and disadvantages of using the feature driven development framework. Some advantages are:

* Simple steps make it easier to execute and allow for more rapid development.
* Larger teams can be split into smaller teams for each feature.

Some disadvantages are:

* Is not suitable for smaller projects or teams as dividing a smaller team into even smaller teams will mean many teams overlapping in the development of different features.
* Can lead to confusion as there is very less documentation.
* Highly dependent on lead developers and project managers.

·