Emerging Programming Paradigms

CED-12

SHIVAM GARG

2017UCO1606

COE - 2

**Ques :** Write short notes on following

· Scrum

· Lean Development

· Extreme programming (XP)

· Adaptive Software Development (ASD)

· Feature Driven Development

**1 : SCRUM**

Scrum is an agile framework in which it is believed that the team doesn’t know everything at the start of a project and will learn through a continuous process of learning. It is a heuristic framework where the developer’s team addresses complex problems. Their aim is to deliver a product which has good productivity and creativity. It uses an iterative process with short release cycles so that the teams can learn and improve from changing conditions and user requirements.

**Salient Features Of Scrum:**

* The framework supports collaboration and self organization of work.
* The team work closely to develop the project
* Scrum team self managers the activities within the timebox.

**SCRUM Roles:**

* **Scrum Master :** Scrum is initiated by a scrum master, who is responsible for removing obstacles for the team and defining the product goals.
* **Product Owner :** The product owner represents the product's stakeholder, who is responsible for delivering good business results.
* **Development Team :** The development team has 3 to 9 members who carry out all the implementations required to build and develop every sprint goal.

**SCRUM CEREMONIES**

* **Sprint:** A sprint has a time-box of one month or less. After the completion of the previous sprint, a new sprint starts immediately with a new goal.
* **Sprint Planning :** At the beginning of a sprint, the scrum team holds a sprint planning event to have a short description of what they are going to deliver at the end of the sprint, product backlog etc.
* **Daily Scrum :** During the daily scrum, each team member typically answers three questions which are : what did I complete yesterday, what do I plan to complete today and do I see any impediment in the sprint goal.
* **Sprint Review:** In this stage, the unachievable features of the product are checked and then the product is passed onto the Sprint Retrospective stage.
* **Sprint Retrospective:** Quality of the product is analysed in this stage.
* **Product Backlog:** The product is organized according to the priority features.
* **Sprint Backlog:** Product assigned features to sprint and Sprint planning meeting are two parts of Sprint Backlog.

**Advantages of Scrum :**

* Scrum framework is fast and cost efficient.
* It divides the larger product into various smaller sub-products.
* Constant feedback taken by the scrum framework increases the quality of the product.

**Disadvantages of Scrum:**

* Scrum framework doesn’t allow modifications in sprint goals.
* It lacks a described model.
* It is difficult to plan, organize and structure a project that doesn’t have a clear understanding of what the project does and so on.

**2: Lean Software Development**

Lean Software Development is an agile framework based on optimizing development time and resources, eliminating waste, and fulfilling the products requirements. The Lean approach is also known as the Minimum Viable Product strategy as the team first develops a bare-minimum version of the product which can be supplied to the market, takes the feedback from the users, learns from users feedback and then proceeds further based on the inferences that is gained from the feedback.

**Principles to Lean software development**

* **Eliminating Waste :** Remove everything which is not adding value to the customer such as partially done work, extra features, relearning, task switching, waiting, handoffs, defects, management activities
* **Building Quality In :** This principle is to ensure that quality is maintained all along the process
* **Amplifying Knowledge :** Lean teams are encouraged to provide the proper document and retain valuable learning.
* **Delaying Commitment :** The team is encouraged to demonstrate in a responsible manner. They do so by keeping their options open and continuously collecting information. Decisions are not taken without referring to the necessary data.
* **Delivering Fast :** This principle follows this concept : Build a simple solution, bring it to the customers and potential users, modify incrementally based on customer feedback.
* **Respecting people :** Lean development teams are encouraged to have proactive communication, healthy conflicts, empowering team members.
* **Optimizing the whole thing :** Lean principles aim to achieve higher performance by optimizing the team’s workflow, creating a sense of unity among the team and giving a sense of shared responsibility and shared objectives.

**Advantages of Lean Software Development**

* Allows more functionality to be done in less time.
* Eliminates unnecessary activity which reduces the costs.
* Allows continuous feedback to have a better product at the end.

**Disadvantages of Lean Software Development**

* Highly dependent on the team.
* Non scalable
* Depends on a strong documentation

**3 : Extreme programming**

Extreme programming is one of the key software development frameworks of Agile models. It is used to improve the quality of the software and to be more responsive to customer requirements.

**Principles of Extreme programming :**

* **Coding:** In Extreme programming, coding activity includes drawing diagrams that will be used for coding, scripting a web-based system etc
* **Testing:** Testing is considered as the primary factor to develop a fault-free software hence model gives high importance on testing
* **Listening:** The developers carefully listen to the customers requirements and note it down so as to develop good quality software.
* **Designing:** A good design results in the elimination of complex dependencies within a system so in Extreme programming the effective use of suitable design is emphasized.
* **Feedback:** Frequent feedback is taken from customers to understand the exact customer needs and to make the development effective.
* **Simplicity:** The main principle of the XP model is to develop a simple system instead of a larger complex system. The system is made to work efficiently. It focuses on some specific features that are immediately needed in the present time.

**Advantages of Extreme Programming**

* It saves costs and time required for project development.
* Constant feedback is the key of framework
* Simplicity is another advantage of Extreme Programming.

**Disadvantages of Extreme Programming**

* Rather than on design, extreme Programming is focused more on the code.
* Extreme Programming does not measure code quality.

**4 : Adaptive Software Development**

Adaptive Software Development is an agile framework which is based on Rapid Application Development. The objective is to enable the teams to adapt to the changing requirements in an efficient manner and evolve their products with a little bit of planning and continuous learning.

**Adaptive Software Development Phases:**

* **Speculation:** This phase is known as the planning phase. The plan includes information like project requirements, user needs etc to define a goal of each release cycle.
* **Collaboration:** It is a phase which includes collaborative communication and teamwork but it majorly emphasizes on individualism as individual creativity plays a major role in creative thinking. People working together must trust each other to criticize without animosity, assist without resentment, work as hard as possible, possess a skill set, communicate problems to find effective solutions.
* **Learning:** Learning helps the workers to increase their level of understanding over the project.Learning process is conducted in 3 major ways which are :
  + Focus groups
  + Technical feedback
  + Project design and learning

**Advantages of Adaptive Software Development**

* Focused on the end users, which can lead to better products
* Allows on-time and even early delivery
* Encourages closed communication between developers and clients

**Disadvantages of Adaptive Software Development**

* Demands extensive user involvement, which is difficult to facilitate
* Integrates testing into every stage, which increases the cost
* Focus on rapid iterating and continuous feedback can lead to scope creep

**5 : Feature Driven Development**

Feature Driven Development is an agile framework which focuses on organizing software development around making progress on features. Features in the FDD are not referred to as product features rather, they are the user stories in Scrum.

#### **Characteristics of FDD**

* **Short iterative:** To finish the work on time along with efficiency, this lifecycle works in simple and short iterations.
* **Customer focused:** This agile practice is totally based on user feedback. Feedback of each feature given by the potential users and customers is taken into account before pushing the code to the main build.
* **Structured and feature focused:** Ad-Hoc activities in life-cycle includes building the domain model and features list
* **Frequent releases:** Feature-driven development provides continuous releases of features and retains continuous success of the project.

**LifeCycle in Feature Driven Development**

### **Gather Data :** The first step is to gain understanding of the project, target audience and their needs

### **Develop overall model :** The team will develop detailed domain models, which are then used to build an overall model that acts as a rough outline of the system

### **Build feature list :** Uses the information collected in the first step to create a list of the required features.

### **Plan by feature :** Analyze the complexity of each feature, specific roles are defined and plan tasks that are related for team members to accomplish

### **Design by feature :** The feature are determine that will be designed and build

### **Build by feature :** This step implements all the necessary items that will support the design

**Advantages of Feature Driven Development**

* Allows progress tracking of projects.
* Allows scaling of team and projects.
* Allows multiple test to work together

**Disadvantages of Feature Driven Development**

* Does not work efficiently for smaller projects
* Poor documentation
* Highly dependent on lead developers