Write short notes on following

Scrum

· Lean Development

· Extreme programming (XP)

· Adaptive Software Development (ASD)

· Feature Driven Development

**NAME : SAAHIL HUSSAIN MUGLOO**

**ROLL NO: 2017UCO1693**

**1.SCRUM**

Scrum is an agile development methodology or framework used primarily for development of software with a goal of achieving the end product within 3-4 weeks time. It is a widely used methodology in which the project progresses via a series of sprints. These sprints are time bound, not exceeding a month. Most commonly they last for 2 weeks duration. A scrum team consists of about 7 (plus or minus 2) members. A meeting is planned at the start of a sprint where in the team members figure out how many items they can commit to and a list of tasks that are to be accomplished during the sprint. This is known as a sprint backlog. Daily scrum or team up meetings are to be done for a period of 15 minutes.

Typical sprint phases include:

* Product backlog: this consists of a list of functionalities with short descriptions derived from description and roadmap of project.
* Sprint backlog: This contains a list of product backlog items for which the development team commits to deliver as part of a particular sprint.
* Sprint planning: This defines the duration and the manner in which the sprint meetings are to be held and also addresses to resource and project planning agendas.
* Sprint: Defined period of time in which the work has to be completed.

The scrum framework is heuristic and is based on continuous learning and adjustment to changing factors. Organisations that exercise this methodology experience:

* Higher productivity.
* Better quality of products.
* Reduced time to market.
* Improved stakeholder satisfaction.
* Better team dynamics.
* Happier employees.

Scrum consists of three roles:

1. Product Owner: Person responsible for product vision and makes decisions on whether or not the products are ready to be shipped.
2. ScrumMaster: Facilitates the scrum process and enforces time bounds.
3. Engineering/development team: This consists of about 7 members responsible for completion of tasks.

While scrum is structured, it is not entirely rigid and there exists some form of flexibility as its execution can be tailored to meet the needs of any organisation. Clear communication, transparency and dedication for continuous improvement are must for achieving success for any team or organisation employing the scrum framework.

**2 LEAN DEVELOPMENT**

Also known as Minimum Viable Product(MVP), Lean Software Development is an agile framework based on optimizing development time and resources, eliminating waste and ultimately delivering only what product needs. The philosophy behind lean methodology is about smart development that is improving virtually everything by eliminating anything that doesn't bring value to the customer. A streamlined approach is followed in this methodology that allows more functionality to be delivered in less time.

LSD is broken down into seven main principles that highlight the working of this development methodology:

1. Eliminating Waste: Anything that contributes no value to the customer is regarded as waste such as partially done work, inventory, extra features, defects etc.
2. Amplifying Learning: Software development is a continuous learning process based on iterations when writing code. Instead of adding more documentation or detailed planning, different ideas could be tried by writing the code and building on it. The iteration process is sped up by usage of short iteration cycles each coupled with refactoring and integration testing.
3. Deferring Commitment: Due to the involvement of uncertainty in the process of software development, decisions should be delayed as much as possible that can lead to better results, achieved through set-based or option-based approach.
4. Delivering Fast: In the current era of technology, survival is much more dependent on the rate at which the end product is delivered, faster the better. Sooner the end-product is delivered with no defects or errors leads to faster feedbacks that can be incorporated in next iterations.
5. Empowering The Team: The lean approach is based on agile principles of building products around motivated individuals and trusting them in getting the job done. People are not treated just as mere resources but are valued on a larger scale.
6. Building Integrity: There should be a balance between flexibility, maintainability, efficiency and responsiveness that can be achieved by understanding the problem domain and solving it at the same time and one of the major ways to work towards integral architecture is refactoring.
7. Optimizing The Whole: Lean thinking has to be understood well by all members of the project, before implementing in a concrete or real life situation. Think bug, act small,fail fast and lean rapidly - these slogans summarize the importance of understanding the field.

**3 EXTREME PROGRAMMING:**

Extreme programming is an agile software development framework that aims at production of higher quality software keeping in view appropriate engineering practices to achieve its goal.

It regards software development project as a system of four control variables which are as follows:

* Cost: This describes the amount of money to be spent keeping in view the available resources.
* Time: This determines when the system is to be released.
* Quality: This describes the overall correctness of the system experienced form user’s (customer's) point of view.
* Scope: This determines the amount of functionality that will be required.

Time is the central variable in XP with all the other values having some form of functional dependency on it.

XP also defines five values which are used as guidelines throughout the development. They are:

* Communication: Good communication is one of the key factors necessary for the success of software development. Customers need to communicate the requirements to the developers while developers need to commute to each other so as to achieve quality end products.
* Simplicity: XP strives for simple systems. This means they should be as simple as possible but at the same time they must not compromise on quality and should work properly.XP also strives for simplicity in methodology.
* Feedback: XP is a feedback-driven process. Through constant feedbacks about previous efforts, teams can identify areas of improvement and revise their practices. Feedback also supports simple design.
* Courage: XP includes courage as well as a certain amount of aggressiveness. Courage is needed because a lot of rules and practices are extreme in a way that they go against tradition or wisdom of software engineering.
* Respect: The members of the team should have mutual respect for each other in order to promote a cordial working environment so as to accomplish goals.

**4 ADAPTIVE SOFTWARE DEVELOPMENT(ASD)**

ASD aims at enabling teams to adapt to requirement changes quickly and effectively and is a direct outgrowth of an earlier agile framework, RAD. It focuses on human collaboration and self organisation which helps to build a complex software and system. JIm HIghsmith and Sam Bayer are credited for the invention of ASD methodology.

This methodology focuses on end users leading to better and intuitive products while encouraging more transparency between developers and clients. However, it also requires extensive user involvement which is difficult to achieve.

PHASES OF ASD

The practices of ASD are driven by a belief in continuous adaptation - a different philosophy and a different life cycle - geared to accepting continuous change as a norm.

In this process the static plan-design is replaced by dynamic Speculate-Collaborate-Learn lifecycle:

* Speculation: In ASD the term plan is replaced by speculate, acknowledging the uncertainty of complex problems and encouraging exploration and experimentation. Iteration with short cycles are also encouraged. “Speculate” gives us more room to explore, to make clear the realisation that we are unsure of.
* Collaboration: The second conceptual component of ASD is collaboration which needs the workers to be motivated. Complex applications are not built, they evolve. Complex applications require that a larger volume of information to be collected, analysed and applied to the problem.
* Learning: Once we admit to ourselves that we are flexible, then learning practices - the “Learn” part of the life cycle becomes vital for success.

This life cycle defined above has six basic characteristics:

1. Mission focused.
2. Feature based.
3. Iterative.
4. Time-boxed.
5. Risk driven.
6. Change tolerant.

**5 FEATURE DRIVEN DEVELOPMENT**

FDD was introduced by Jeff De Luca and Peter Coad in 1997 and is one of the agile software development methodologies that has emerged in the last decade or so as an alternative to the traditional waterfall development model so as to avoid the weaknesses of the waterfall model by doing iterative development.

FDD is an agile, highly adaptive software development process that :

* Supports high iterations that are short.
* Emphasizes on quality on all the steps of production.
* Delivers frequently and tangible working results at all steps.
* Provides accurate and meaningful progress and status information, with minimum overhead and disruption for developers.

There are six primary roles on an FDD project.These are:

1. Project Manager.
2. Chief Architect.
3. Development Manager.
4. Chief Programmer.
5. Class Owner.
6. Domain Expert.

An individual can take on one or more roles.

The FDD life cycle involves the following key steps:

* Developing an overall model.
* Building feature list.
* Planning by feature.
* Designing by feature.
* Building by feature.

This type of development is advantageous as it is highly adaptive and greatly stresses on designing and modeling aspects of the project, however it does not address to issues related to project criticality.