(P) Chapter 4

An Agile View of Processes

⊘ What is agility?

□ Defination:

Agility is dynamic, contentspecific, aggresively change embracing and growth oriented.

□ Characteristics:

- encourages team structure and attitude that makes communication more facile.
- emphasize rapid delivery of operational software and de-emphasize importance of intermediate work product.
- adopts customer as a part of development team and works to eliminate the "us and them" attitude
- recognizest that planning in an uncertain world has its limits and that project plan must be flexible.

□ Principles:

1. highest priority is to satisfy customer through early and continuous delivery of valuable software.
2. Welcome changing requirements.
3. Delivery working software frequently.
4. Business people and developers must work together.
5. Build projects around motivated individuals.
6. Emphasize face-to-face conversation.

⑦ Working software is the primary measure of progress

⑧ Agile processes promote sustainable development.

⑨ Continuous attention to technical excellence and good design enhances agility.

⑩ Simplicity - the art of maximizing the amount of work not done is essential.

⑪ Best architectures, requirements and design emerge from self-organizing team.

⑫ team tunes and adjusts its behaviour to become more effective.

● What is an agile software process?

An agile software process is characterized in a manner that addresses three key assumptions—

① Difficulty in predicting changes of requirements and customer priorities.

② For many types of software, design and construction are interleaved.

③ Analysis, design, construction and testing are not as predictable as we might like.

□ Characteristics:

- must be adaptable
- → must adapt incrementally
- → Requires customer feedback
- → An effective catalyst for customer feedback is an operational prototype.

Human factor:

→ Process molds to the need of people and team, not the other way around.

- What key traits must exists among the people on an effective software team ?

1. Competence
2. Common ~~factor~~ focus.
3. Collaboration
4. Decision-making ability
5. Fuzzy problem solving ability
6. Mutual trust and respect
7. Self organization.

Agile Process Models:

① Extume programming (XP)
② Adaptive software development. (ASD)
③ Dynamic systems development Methods. (DSDM)
④ Scrum
⑤ Feature Driven Development (FDD)
⑥ Agile Modeling (AM)

Ⓠ Illustrate and discuss the XP programming process model with respect to its framework activities ?

Ⓠ Illustrate XP programming process model and note some of the key ideas and tasks that are associated with each framework activity.

□ Extreme Programming (XP):

XP uses an object oriented approach as its preferred development paradigm and it has four framework activity. They are–

* ① Planning   ② Design
③ Coding   ④ Testing

* Planning:

⇒ Begins with the creation of a set of user stories.

⇒ Each story is written by the customers and is placed on an index card.

⇒ Customer assigns a value to the story
priority

⇒ Agile team assesses each story and assigns a cost.

⇒ Stories are grouped to for a deliverable increments.

⇒ A commitment is made on delivery date.

⇒ After first increment "project velocity" is used to help define subsequent delivery dates for each increments.

□ Design:

⇒ Follows the keep it simple (KIS) principles.

⇒ Encourages the use of Class-Responsibility-Collaborator (CRC) cards.

⇒ For difficult design problems, suggests the creation of "spike solutions" - a design & prototype.

⇒ Encourages refactoring - an iterative refinement of the internal design.

⇒ Design occurs both before and after coding commences.

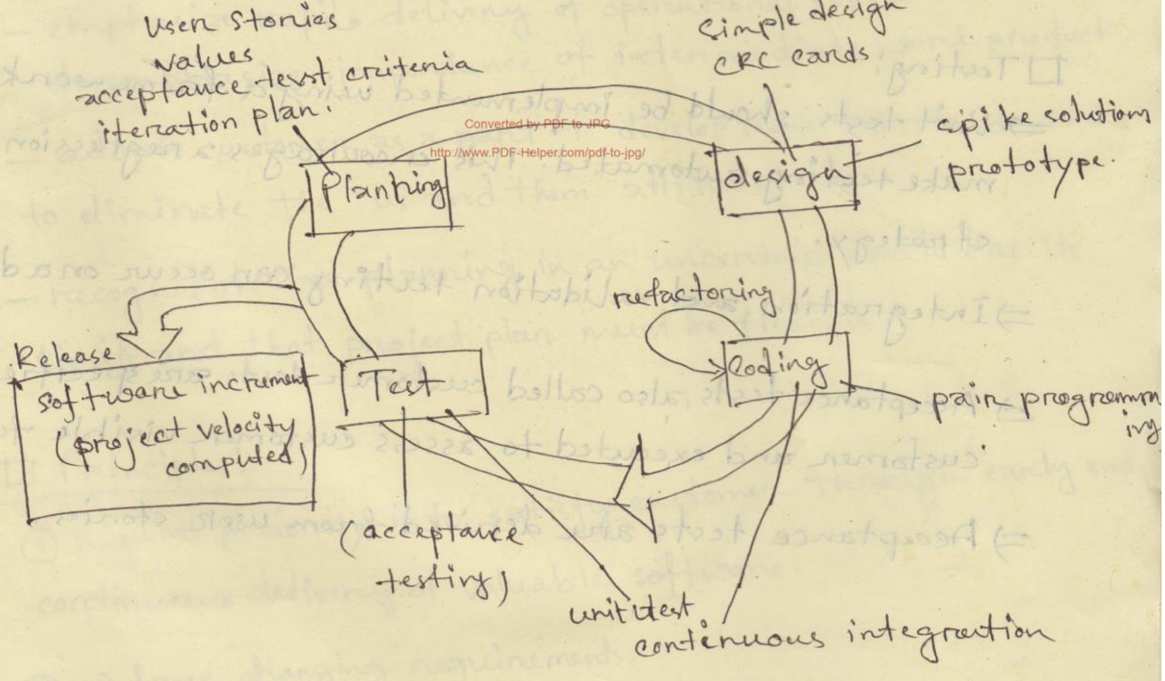

Fig: 4.1. Extreme Programming Process.

□ Coding:

⇒ Recommends construction of a series of unit tests for each of the stories before coding commencies.

⇒ Encourages "pair programming"

→ Mechanism of real-time problem solvingand real time quality assurance.

→ Keeps the developers focused on the problem at hand.

⇒ Needs continuous integration with other portions (stories) of software, which provides a "smoking test" environment.

□ Testing:

⇒ Unit tests should be implemented using a framework to make testing automated. This encourages a regression testing strategy.

⇒ Integration and validation testing can occur on a daily basis.

⇒ Acceptance tests, also called customer tests are specified by the customer and executed to assess customer visible functionality.

⇒ Acceptance tests are derived from user stories..