

Assignment

1. What are various Agile principal followed during software development?

Ans: • Highest priority is to satisfy customer through early & continuous delivery of valuable software.

• Working software should be delivered from couple of weeks, couple of months in shorter time or cycle.

• Business people & developer must work together daily through the projects.

• Build the project around the motivated developer give them the liberty or the environment so that the job can be completed in time.

• Working software is the primary measure of progress.

• Communication b/w developer team is talk to face to

• Importance is given to technical excellence and good design

• The design should be simple.

• Try to reduce the work done which is not essential.

• The best architecture required & design

comes out /emergence from self-organising team. Agile process must promote sustainable development at regular intervals. The developer team focuses on becoming effective.

2. What are the human factors responsible for agile development? What are the key trades that must be present among the people in the agile team?

Ans : The key trades are competence, common focus, collaboration, decision making ability, fuzzy problem solving ability, mutual trust & respect & self-organising.

(i) Competence - It means talent & specific skills possessed by the members of agile team or the skills that must be through all the people of agile team.

(ii) Collaboration - It should be b/w software customers, stakeholders which are necessary for the software building.

(iii) Common focus - All the members of agile team should focus on ^{one} goal that is to develop quality oriented software as per requirement of customer.

(iv) Decision Making Ability - All the members of agile team must be allowed to freedom that are required for effective software development covering project & technical issues.

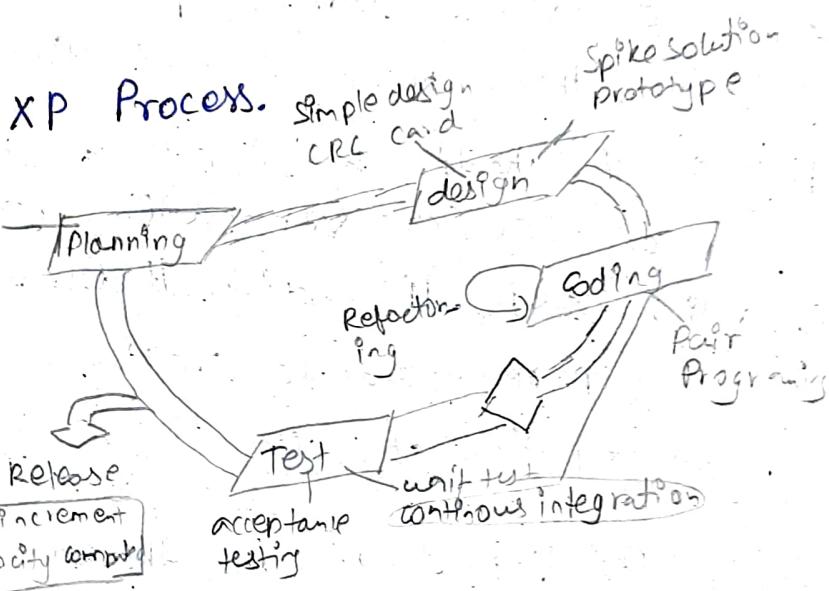
(V) Mutual trust & Respect - This must exist among all the members of agile team that are necessary for building effective software.

(VI) Self Organization - Agile team organizing itself, in order to complete the work according to the local environment. Team organizes itself so that the work schedule is maintained & software is developed & delivered in time.

Q Describe XP Process.

Ans

User stories
values acceptance
test criteria
iteration plan



The key XP activities are

- Planning
- Design
- Coding
- Test
- Release

> Planning - This includes business context for software like cost, Major features & functionality as per the requirement given by the customer. Each feature is called as story & if it is

assigned a value / priority indicating some values.

so, the XP team orders the stories that needs to be developed in following ways:

- All the stories should be implemented immediately.
- The story with more highest should be implemented first.

- Depending upon the date schedule the story needs to be implemented as the development work proceeds. Customer can add stories & change the values of existing stories, split stories or even eliminate them according to the time schedule.
- ▷ Design: It should be simple for a story encouraging the use of CRC Cards (Collaboration responsibility column, collaboration) cards. If a difficult design (concerned, xp recommends the creation of operational prototype or portion of design, also called Spike solution.
- ▷ Coding: The team doesn't do the code directly but first it develops a series of unit test, data used test & the story which has been implemented. Once the code is completed it can be retested.
- ▷ Testing: The unit test that are created always encourage a regression testing strategy. Whenever the change is initiated by the customer it should not effect the existing software product code which have already been developed.

▷ Releases - It is the delivery software to the customer as & when the stories are implemented in no form or a software increment.

Q. Difference between ~~stakeholders~~ and viewpoints, functional & non-functional requirement.

Ans:

Functional Requirements

- A functional requirement defines a system or its component.
- It specifies "what should the software do?"
- Functional requirement is specified by user.
- It is mandatory.
- It is captured in use case.
- Defined at a component level.
- Helps you to verify the functionality of the software.
- Its testing like system, integration, End to End, API testing, etc are done.
- Usually easy to define

Ex- System shutdown in case of a cyber attack

Non-functional Requirements

- A non-functional requirement defines the quality attribute of a software system.
- It places constraints on "how should the software system fulfill the functional requirements?"
- Not this is specified by technical people e.g - Architect, Technical leader & software developers..
- It is not mandatory.
- It is captured as a quality attribute.
- Applied to a system as a whole.
- Helps you to verify the performance of the software.
- Its testing like performance, stress, usability, security testing, etc are done.
- Usually more difficult to define.
Ex- The processing of each request should be done within 10 seconds.

Q. What are the key trades / difference b/w Stakeholders views
points?

Ans:- Stakeholders are the people any one who directly
benefits directly and indirectly from the system.

Ex:- Business operation manager, product manager,
marketing people, Internal & external customers
product & software engineer, Support Engineer.

→ Each stakeholders have different views of
system which will contribute to requirement

Assignment

1. Explain Spiral Model & prototype model with advantages & disadvantages.

Ans: Spiral Model:

It is a risk-driven software development process model. It is a combination of waterfall model & iterative model. Spiral model helps to adopt software development elements of multiple process models for the software project-based on unique risk patterns ensuring efficient development process.

Advantages:

- Additional functionality or changes can be done at a later stage.
- Cost estimation becomes easy as the prototype building is done in small fragments.
- Continuous or repeated development helps in risk management.
- Development is fast & features are added in a systematic way in spiral development.
- There is always a space for customer feedback.

Disadvantages:

- Risk of not meeting the schedule or budget.
- Spiral development works best for large projects only also demands risk assessment expertise.

For a smooth operation spiral model protocol needs to be followed strictly.

- Documentation is more as it has intermediate phases.

- Spiral software development is not advisable for smaller project, it might cost them a lot.

Prototype Model: It is used when the customers do not know the exact requirements before hand.

Advantages:

- This model is flexible in design.
- It is easy to detect errors.
- We can find missing functionality easily.
- It is ideal for online system.
- It helps developers & users both understand the system better.
- It can actively involve users in the development phase.

Disadvantages:

- This model is costly.
- There may be too much variation in requirement.
- There is certainty in determining the numbers of iterations.

- o There may be incomplete or inadequate problem analysis.
- o There may be increase in the complexity of the system.

2. Describe a short note on reverse engineering.

Ans: It is a process of recovering the design requirement specifications & functions of a product from an analysis of its code. It builds a program database & generates information from this.

The purpose of this process is to facilitate the maintenance work by improving the understandability of a system & to produce the necessary documents for a legacy system.

Goals:

- o Cope with complexity
- o Recover lost information
- o Detect side effects
- o Synthesise higher abstraction
- o Facilitate reuse

Steps of R.E

- i. Collection Information: This step focus on collecting all possible information about the software
- ii. Examining the information: The information collected in step 1 is studied so as to get familiar with the system.

3. Extracting the structures: This step concerns with identification of program structure in the form of structure chart where each node corresponds to some routine.
4. Recording Functionality: During this step processing details of each module of the structure, charts are recorded using structural language like decision table, etc.
5. Recording data flow: From the information extracted in step-3 & step-4, set of data flow diagrams are derived to show the flow of data ~~money~~ among the processes.
6. Recording Control Flows: High level control structure of the software is recorded.
7. Review extracted design: Design document extracted is reviewed several times to ensure consistency & correctness. It also ensures that the design represents the program.
8. Generate documentation: Finally, in this step, the complete documentation includes SRS, design document, history, overview, etc. are recorded for future use.
3. Write a note on concurrent engineering & unified process model along with advantages & disadvantages.

Ans: Concurrent Models in Software: Most of the successful software out there involves a series of phases of development, such as requirements gathering and prototyping, that are put together to develop the software. These phases are discrete & often performed concurrently. Often there is an intertwining between the phases, which makes it inevitable to return to the earlier phases to make some changes according to the results obtained in the later phases.

Advantages:

- It's relatively simple to understand.
- It's a sequential model. Each phase is followed by the next in sequential order. In any phase, if you want to return back some earlier phase then you would have to go through the entire process again sequentially.
- Decrease design & development time taken make products which match their customers needs, in less time & at a reduced cost.
- Focuses on accurate documentation, hence can be considered as complete methodology.

Disadvantages:

- Complex & disorganized development process.
- Reusability is impossible to the projects which incorporates new technology.
- High expense can be involved in heavy documentation.

- Issues may arise at the testing phase due to ^{too} many integrations.
- Except expert team members.

Unified Process Model: Unified process (UP) is an architecture centric, use case driven, iterative & incremental development process. UP is also referred to as the unified software development process.

The unified process is an attempt to draw on the best features & characteristics of traditional software process models, but characterize them in a way that implements many of the best principles of agile software development.

Phases

- i. Inception Phase
- ii. Elaboration Phase
- iii. Construction Phase
- iv. Transition Phase

i. Inception Phase:

- ▷ Vision document
- ▷ Initial use case diagram
- ▷ Initial risk assessment
- ▷ Project Plan.

ii. Elaboration Phase

- ▷ Use case Model
- ▷ Analysis Model
- ▷ Software Architecture description
- ▷ Preliminary design model.
- ▷ Preliminary model.

(III) Construction Phase

A Design Model

- ▷ System Components
- ▷ Test plan & procedure
- ▷ Test cases
- ▷ Manual.

(IV) Transition phase

▷ Delivered Software Increment

▷ Beta test results

▷ General use feedback.

Advantages:

o It covers the complete software development life cycle.

- The best practice for software development are supported by unified process model.
- It ~~also~~ encourages designing in UML.

Disadvantages:

- Could be complex to implement
- Heavy weight process
- Experts are needed for it.
- Risks cannot be determined

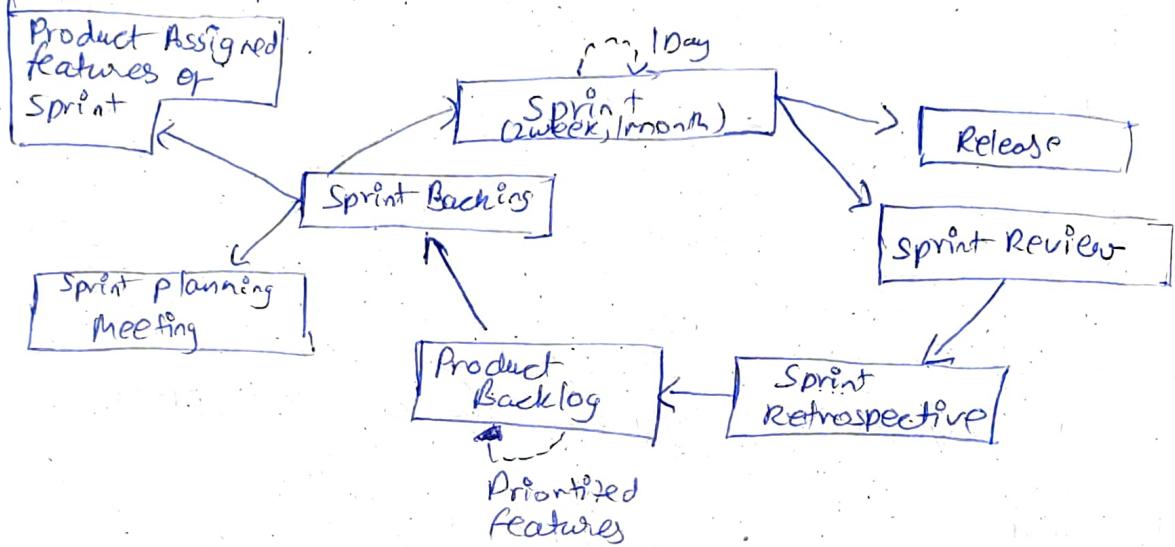
Q. Write a note on scrum digital feature development.

Ans: Scrum is the type of Agile framework. It is a framework within which people can address complex adaptive problems while productivity & creativity of delivering product is at highest possible values. Scrum uses Iterative process.

Silent features of Scrum:

- It is light-weighted framework
- Scrum emphasizes self-organization.
- It is simple to understand.

• This framework help the team to work together.



Advantages:

- It is fast moving & money efficient.
- Customer satisfaction is very important.
- It is adaptive in nature because it have short sprint.

Disadvantages:

- It do not allow changes into their sprint.
- It can be difficult for the scrum to plan, structure & organize a project that lacks a clear definition.
- The daily scrum meetings & frequent review required substantial ~~resources~~ resources.

so Dynamic Software driven features derived development, Adaptive Software development,

Ans:

Dynamic Software Development: It is an Agile method that focuses on the full project lifecycle, DSDM XP framework was created in 1994, after project manager using Random Application Development (RAD). So added more Governance & discipline to this new iterative way of working.

Its success is due to the philosophy "that any project must be aligned to clearly defined strategic goals and focus upon early delivery of real benefits to the business." Supporting this philosophy with the eight principles allows teams to maintain focus & achieve project goals.

Eight principles:

- Focus on no business need
- Deliver on time.
- Collaborate
- Never compromise quality
- Build incrementally from the firm foundation.
- Develop ~~iterately~~ iteratively.
- Communicate continuously & clearly
- Demonstrate ~~continuity~~ control.

Adaptive Software development: It is a method to build complex software & system. ASD focuses on human collaboration & self-organization. ASD's life cycle incorporates three phases namely:

1. Speculation
2. Collaboration.
3. Learning.

1. Speculation - During this phase project is initiated and planning is conducted. The plan uses initiation information like project requirements, user needs, customer mission statement, etc.

2. Collaboration: People working together must trust each others to

- ▷ Criticize without animosity
- ▷ Assist without resentment
- ▷ Possession of skill set
- ▷ Communicate problems to find effective solution.

3. Learning:

- ▷ Focus groups
- ▷ Technical reviews
- ▷ Project postmortem,

Feature Driven Development: It is an agile framework that, as its name suggests, organizes software development around making progress on features. Features in the FDD context, through are not necessarily In the commonly understood sense.

It has five step development process

- Develop an overall model
- Build a features list.
- Plan by feature.
- Design by feature
- Build by feature