

Assignment 1.

- 1) Write agile manifesto and principles in detail.

Ans 1)

- > Individuals and interactions over processes and tools
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation
- Responding to change over following a plan.

* Principles behind the Agile Manifesto

- 1) Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- 2) Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- 3) Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- 4) Business people and developers must work together daily throughout the project.
- 5) Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

- 2
- 1) The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- 2) Working software is the primary measure of the progress.
- 3) Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- 4) Continuous attention to technical excellence and good design enhances agility.
- 5) Simplicity—the art of maximizing the amount of work not done—is essential.
- 6) The best architectures, requirements, and designs emerge from self-organizing teams.
- 7) At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.
- 8) Explain in detail Agile Advantages and disadvantages.

3

* Advantages of Agile

- In Agile methodology the delivery of software is unrelenting.
- The customers are satisfied because after every sprint working feature of the software is delivered to them.
- If the customers has any feedback or any change in the feature then it can be accommodated in the current release of the product.
- In Agile methodology the daily interactions are required between the business people and the developers.
- In this methodology attention is paid to the good design of the product.
- Changes in the requirements are accepted even in the later stages of the development.

* Disadvantages of Agile

- In Agile methodology the documentation is less.
- Sometimes in Agile methodology the requirement is not very clear hence its difficult to predict the

expected results.

- iii) In few of the projects at the starting of the software development life cycle it's difficult to estimate the actual effort required.
- iv) The projects following the Agile methodology may have to face some unknown risks which can effect the development of the project.

3) Discuss and explain responsibilities of product owner in detail.

Ans 3) i) Manage Economics

- The product owner is responsible for ensuring that good economic decisions are continuously being made at the release, sprint, and product backlog levels.

Release-level Economics

Sprint-level Economics

Product-Backlog Economics

2) Participate in Planning:

- The product owner is a key participant in

the portfolio-, product-, release- and sprint-planning activities.

- During portfolio planning, the product owner works with internal stakeholders to position the product correctly in the portfolio backlog and to determine when to start and end product development.
- During product planning, the product owner works with the stakeholders to envision the products.
- During release planning, the product owner works with the team to define the content of the work next release.
- During sprint planning, the product owner works with the development team to define a sprint goal.
- He also provides valuable input that enables the development team to select a set of products backlog items that the team can realistically deliver by the end of the sprint.

3) Groom the Product Backlog

- The product owner oversees the grooming of the product backlog, which includes

6
creating and refining, estimating and prioritizing product backlog items.

- The product owner doesn't personally perform all of the grooming work. For example, he might not write all of the product backlog items; others might contribute them.
- The product owner also doesn't estimate the items but is available for questions and clarification during estimation.
- The product owner is, however, ultimately responsible for making sure that the grooming activities take place in a way that promotes the smooth flow of delivered value.

4) Define Acceptance Criteria and Verify That They Are Met:

- The product owner is responsible for defining the acceptance criteria for each product backlog item.
- These are the conditions under which the product owner would be satisfied that the functional and non-functional requirements have been met.

7
The product owner may also write acceptance tests corresponding to the acceptance criteria, or he could enlist the assistance of subject matter experts (SMEs) or development team members.

5) Collaborate With Team

The product owner must closely collaborate with the development team on a frequent basis.

The product owner is an engaged, committed, everyday role.

Many organizations just starting to adopt Scrum fail to foster adequate product owner engagement with the development team, delaying essentials feedback and substantially reducing the value of that feedback when it does occur.

6) Collaborate with the Stakeholders:

The product owner is the single voice of the entire stakeholder community, internal and external.

Internal stakeholders can include business systems owners, executive management, program management, marketing, and sales.

External stakeholders can include customers, users,

partners, regulatory bodies and others.

- The product owner must work closely with the entire stakeholder community to gather input and synthesize a coherent vision to guide product development.

4) What is the use of burn up and burn down charts?

Ans -> Burnup Chart:

→ A burnup chart tracks progress towards a project's completion. In the simplest form of burn up chart, there are two lines on the chart:

A total work line (the project scope line)
A work completed line.

Burn Down chart:

→ It is a graphical representation of work left to do versus time. The outstanding work is often on the vertical axis, with time along the horizontal.

- It is useful for predicting when all of the work will be completed.

8

5) Define: Story point and Velocity.

Ans -> Story point is used for effort estimation for a user story, it's a number without any unit, and sp value represents the comparative complexity from one story to another.

→ Classic example is reading a book or travelling from point A to C. Usually this could be value such as t-shirt size or number of cookies.

* Velocity.

- Velocity is a measure in the Agile methodology.
- Simply put, velocity measures the rate at which team gets work done in a single iteration.
- It deals with what actually was achieved as opposed to what was planned in that iteration.

6) Discuss and explain responsibilities of development team in detail.

→ Here are the responsibilities of development team. include:

- 1) Perform Sprint Execution
- 2) Inspect and Adapt Each day.
- 3) Groom the Product Backlog.
- 4) Plan the Sprint.
- 5) Inspect and Adapt the Product and Process.

9

1) Perform Sprint Execution.

During sprint execution, development team members perform the hands-on, creative work of designing, building, integrating, and testing product backlog items into increments of potentially shippable functionality.

- To do this, they self-organize and collectively decide how to plan, manage, carry out, and communicate work.
- The development team spends a majority of its time performing sprint execution.

2) Inspect and Adapt Each Day.

- Each development team member is expected to participate in each daily scrum, during which the team members collectively inspect progress toward the sprint goal and adapt the plan for the current day's work.
- If some team members do not participate, the team can miss pieces of the big picture and may fail to achieve its sprint goal.

3) Groom the Product Backlog.

- Part of each sprint must be spent preparing for the next.
- A large part of that work focuses on product backlog grooming, which includes creating and refining, estimating, and prioritizing product backlog items.
- The development team should allocate up to 10% of its available capacity every sprint to assist the product owner with these activities.

4) Plan the Sprint.

- At the beginning of each sprint, the development team participates in sprint planning.
- In collaboration with the product owner and with facilitation from the Scrum Master, the development team helps to establish the goal for the next sprint.
- The team then determines which high-priority subset of product backlog items to build to achieve that goal.
- For a two-week sprint, sprint planning typically takes about half a day.
- A four-week sprint might need up to a full day of for sprint planning.

- Notice that planning happens iteratively.
- Rather than focusing on a very large, uncertain, and overly detailed plan at the start of a development effort, the team makes a series of smaller, more certain, and more detailed plans just in time at the beginning of each sprint.

5) Inspect and Adapt the Product and Process

→ At the end of each sprint, the development team participates in the two inspect-and-adapt activities: sprint review and sprint retrospective.

- The sprint review is where the development team, product owner, Scrum Master, stakeholders, sponsors, customers, and interested members of other teams review the just-completed features of the current sprint and discuss how to best move forward.

- The sprint retrospective is where the Scrum team inspects and adapts its Scrum process and technical practices to improve how it uses Scrum to deliver business value.

7) What is incremental and iterative model?

Sessional I.

i) Answer the following.

(i) Scrum development teams main characteristics are _____ and _____.

Ans: - (a) Self organize
(c) Cross functional

(ii) Order list of prioritize requirements what is to be need to complete project is known as _____.

Ans: - (b) Product Backlog.

(iii) Burn down chart is a part of sprint Backlog
State True/False. Justify.

True

(iv) The release to production is done automatically is called Continuous deployment.

(v) _____ has responsibility to work as powerful motivator

Ans: - (a) Product owner.

(vi) _____ and _____ are two inspect and adapt activities in scrum

Ans:- (a) review
(c) retrospective.

Qn 3)

B) Explain responsibilities of Scrum Masters in detail.

Ans:-

1) Coach:-

- The Scrum Master is the agile coach for the Scrum team - both the development team and the product owner.
- Analogous to the coach of a sports team, the Scrum Master observes how the team is using Scrum and does anything possible to help it get to the next level of performance.
- When problems arises that the team can and should be able to solve, the Scrum Master's attitude, like that of any good coach, is "I'm not here to solve your problems for you; instead, I'm here to help you solve your own problems!"
- If the problem is an impediment that the team can't resolve, the Scrum Master takes ownership of getting it resolved.

14

13) Interface sc inheritance

List and constructor

constructor
StringBuf

creates a

15

The Scrum master is often described as a servant leader of the Scrum Master.

- A servant leader:- Would never ask, "So, what are you going to do for me today?" Instead, a servant leader asks, "So, what can I do today to help you and the team be more effective?"

- Process Authority:- The Scrum Master is the scrum team's process authority. In this capacity, the Scrum Master is empowered to ensure that the Scrum team enacts and adheres to the values, principles and practices along with the Scrum team's specific approaches.

2) Interference Shield:-

- The Scrum Master protects the development team from outside interference so that it can remain focused on delivering business value every sprint.

- Interference can come from any number of sources, from managers who want to redirect team members in the middle of a sprint, to issues originating from other teams.

- No matter what the source of the interference, the Scrum Master acts as an interceptor so that the team can focus on delivering value.

car C
-23

3) InterImpediment Remover

- The Scrum Master also takes responsibility for removing impediments that inhibit the team's productivity.
- For example, I observed a Scrum team that was consistently unable to meet its sprint goals.
- The team itself had no control over these servers - that was the responsibility of the VP of Operations.
- Because the team itself could not remove the impediments, the Scrum Master took ownership of improving the server stability by working with the VP of Operations and others who could actually do something about the stability issues.

4) Change Agent:

- The Scrum Master must help change more than faulty servers and similar impediments.
- A good Scrum Master must help change minds as well.
- The Scrum Master helps others understand the

16

need for change, the impacts of Scrum outside of the Scrum team, and the broad-reaching benefits Scrum can help achieve.

- The Scrum master also ensures that effective change is occurring at all levels of the organization, enabling not only short-term success but, more importantly, the long-term benefits from using Scrum.

- In large organizations, the Scrum Masters might band together to become a more effective force for change.

A) Explain collaborative user stories with example. Explain three components of collaborative stories in detail.

Ans: User stories are part of an agile approach that helps shift the focus from writing about requirements to talking about them.

- All agile user stories include a written sentence or two and, more importantly, a series of conversation about the desired functionality.

→ User Stories are short, simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system.

17

- They typically follow a simple template:
- As a <type of user>, I want <some goal> so that <some reason>.
- User stories are often written on index cards or sticky notes, stored in a shoe box, and arranged on walls or tables to facilitate planning and discussion.
- In Agile development, user stories are written to capture requirements from the perspective of developers, testers, and business representatives.
- Example:
As an online Bank customer, I want to access my bank statement for the last 5 years.

* Components of Collaborative User Stories.

- Ron Jeffries proposed the 3C concept to describe the component elements of a user story.
- The three elements are card, conversation and confirmation.
- 1) Card:- The card is the physical media which

- describes a user story.
- It identifies the requirement, its criticality and the acceptance criteria.
- It also includes expected development and test duration.
- This card is used in the product backlog so the description has to be accurate.
- 2) Conversation:- The requirements are communicated and understood, in a number of conversations conducted between the customers, developers and testers.
 - The conversation explains how the software will be used. The conversation can be documented or verbal.
 - The tester perspective adds much to this conversation by asking questions, identifying ambiguity, missing elements such as non-functional aspects and suggesting ways to test the story.
 - Conversation begins during the release-planning phase and continues when the story is scheduled for inclusion in iteration.
- 3) Confirmation:- The acceptance criteria, discussed in the conversation, are used to confirm that the story is done.
 - These acceptance criteria may span more than one user stories. Positive and negative tests should be used to cover the criteria.

- To confirm that the story is done, the defined acceptance criteria should be tested and satisfied.

Q3) OR

- A) Explain comparison of waterfall and agile methodology in detail.

Agile Methodology	Waterfall Methodology
1) It separates the project development process development lifecycle into sprints.	Software development process is divided into distinct phases.
2) It follows an incremental approach	Waterfall methodology is a sequential design process.
3) It is known for its flexibility.	It is structured software development methodology so most times it can be quite rigid.
4) Agile can be considered as a collection of many different projects.	Software development can be completed as one single project.
5) Agile is quite a flexible method which allows changes to be made in the project development.	There is no scope of changing the requirements once the project development starts.

Requirement even if the initial planning has been completed.

- 6) It follows an iterative development approach because of this phases like designing, planning, development, prototyping development, testing etc. and other software development are completed once in the phases may appear more than once.
- 7) Test plan is reviewed after each sprint. The test plan is really discussed during the test phase.
- 8) It is a process in which the requirements are expected to change and evolve. The method is ideal for projects which have definite requirement & changes not expected.
- 9) Testing is performed concurrently. In this methodology, the "Testing" phase comes after the "Build" phase.
- 10) Agile introduces a product mindset where the software project mindset and places product satisfies needs of its end customers and changes on accomplishing the itself as per the customer demand.
- 11) It works exceptionally well with Time & Material or non-fixed funding contracts by getting risk agreement. It may increase stress in fixed price scenarios.

- 22)
- 12) Prefers small but dedicated teams with a high degree of coordination & synchronization. Team coordination / synchronization is very limited.
- 13) Product owner with team prepares requirements just about every day doing a project. Business analyst prepares requirement before the beginning of the project.
- 14) Test team can take part in the requirements change without problems. It is difficult for the test requirements to initiate any change in requirements.
- 15) Description of project details can be altered anytime during the SDLC process. Detail description needs to implement waterfall software development approach.
- 16) The Agile team members are interchangeable, as a result, they work always faster. There is also no project manager need for project manager because the projects are managed by the entire team.

- 23)
- 1) A) Explain characteristics of Agile Methodology.
- 1) Modularity
 - 2) Iterative
 - 3) Time-bound
 - 4) Incremental
 - 5) Convergent
 - 6) People-oriented
 - 7) Collaborative.
- 1) Modularity:
- The Agile process decomposes the complete system into manageable pieces called modules.
 - Modularity plays a role in software development process.
- 2) Iterative:
- The main objective of agile software processes is satisfaction of customers, so it focuses on single requirement with multiple iteration.
- 3) Time-bound:
- As agile process is iterative in nature, it requires the time limits on each module with respective cycle.

4) Incremental

→ As the agile process is iterative in nature, it requires the system to be developed in increments, each increment is independent of others, and at last all increments are integrated into complete system.

5) Convergent.

- All the risks associated with each increment are convergent in agile process by using iterative and incremental approach.

6) People oriented.

- In the agile processes, customer satisfaction is the first priority over the technology and process.

7) Collaborative.

As agile process is modular in nature, it needs a good communication among software development team.

- Diff

24

③ Write user stories and acceptance criteria for customer who can login, register, and add to cart functionality for online buying products.

Modular

The Agile process determines the complete system into co-ordinated pieces called modules. Modularity plays a role in software development process.

Modular

The main objective of agile software process is satisfaction of customer and delivery long requirement with multiple milestones.

Modular

An agile process is done in certain phases. The time looks in each module with a specific cycle.