

SYS5640: System Requirements

Final Exam

Grant Butler

Table of Contents

- 1. Traditional vs. Agile
- 2. Agile Methodology
- 3. Avg. Length of SPRINT
- 4. SCRUM Master
- 5. Name Agile Methodologies
- 6. Scrum. vs XP
- 7. Four Values of Agile
- 8. What is XP?
- 9. Involved vs. Committed
- 10. Scrum Artifacts
- 11. Burndown Chart
- 12. TF Slides at a Sprint Review
- 13. Sprint Backlog vs. Product Backlog
- 14. Team Velocity
- 15. TF Agile in Infrastructure
- 16. SAFe
- 17. Why Agile?

1. Traditional vs. Agile

In a short paragraph describe the main differences between agile and traditional requirements gathering.

The core difference between Agile methodology and traditional requirements gathering is the order in which the steps are done. Traditional often starts with features clearly marked and they are kept throughout development of the system. The cycle is done linearly, and a waterfall of requirements is assumed to be delivered by the end of development. In Agile, requirements gathering is done in quicker iterative cycles that focus on specific requirements done in smaller groups that meet frequently to discuss progress and reassess the scope, validity and verification of features as development progresses.

2. Agile Methodology

Which one of these is an acceptable agile methodology?

- XP
- V Model
- Waterfall model
- Spiral model

Ans: XP

3. Avg. Length of SPRINT

What is the average length of a SPRINT in a SCRUM project environment?

2-4 Weeks

4. SCRUM Master

Describe the role of a Scrum Master.

The Scrum Master is a Project Manager/Team Leader that is there to enforce Scrum methodology and ensure that the team works smoothly by resolving conflict and arguments, in order to keep everyone productive.

5. Name Agile Methodologies

Name 3 agile methodologies used for software development.

Scrum, Extreme Programming (XP), and Kanban are three methodologies used to do Agile development.

6. Scrum. vs XP

What is the main difference between SCRUM and XP?

Scrum breaks a project down into small chunks and focuses on quickly working to finish those problems quickly in a time period called a sprint. Extreme Programming (XP) tries to deliver high quality software, striving to keep core principles of Communication, Simplicity, Feedback, Respect and Courage at the forefront of the development cycle. XP also has small iterations, but unlike Scrum, XP allows for changes in requirements as long as work hasn't started. Scrum is also more open-ended in certain practices like TDD or automated testing, which XP encourages.

7. Four Values of Agile

What are the four values of the agile manifesto?

- Individuals and Interactions Over Processes and Tools
- Working Software Over Comprehensive Documentation
- Customer Collaboration Over Contract Negotiation
- Responding to Change Over Following a Plan

8. What is XP?

In AGILE what does XP stand for?

Extreme Programming

9. Involved vs. Committed

In your own words (short paragraph) explain the differences between involved and committed.

Being committed to a project is ensuring that no matter what the project comes first, and a member who is committed will work to ensure the project comes to fruition even when bumps in the road of development come to make things difficult. An involved member is simply there to contribute somewhat but often will not work to complete any extra hardships that may occur.

10. Scrum Artifacts

Name at least 2 SCRUM artifacts.

- Product Backlog
- Sprint Backlog
- Burndown Charts

11. Burndown Chart

What is the purpose of a burn down chart? And what can you learn from it?

A burndown chart shows what is completed in a sprint, and what is left to complete. Each developer or work item will have one, and are typically updated daily to show how many hours or points are completed daily. The horizontal axis shows the length of the sprint, and the vertical axis shows how many hours of work are remaining to be done. These charts provide direct feedback over the length of a sprint vs. the work in each of them, allowing the team to adjust how much each sprint contains and how much work is happening in them. An ideal burndown trends linearly, indicating that the length of sprint is ideal for the amount of work that needs to get done.

12. TF Slides at a Sprint Review

I should bring a set of slides to a sprint review.

Ans: False

13. Sprint Backlog vs. Product Backlog

What is the difference between sprint backlog and backlog?

A backlog is an ordered list of items that need to deliver specific outcomes. A Product Backlog contains all of these deliverables for the entire project, and influence the prioritization of what tasks are going to be put into the next Sprint being done. The Sprint is then planned out and a Sprint Backlog is created to clearly define the tasks that need to get done in that Sprint. The Sprint Backlog is updated daily and can be updated throughout the duration of the Sprint by any member of the team, while the Product Backlog is prioritized by the owner of the product.

14. Team Velocity

In a short paragraph, explain team velocity.

Team Velocity is used to determine long term scheduling for a project. By finding the mean velocity from the team's last three iterations, a Team Velocity can be determined to see how quickly a project is expected to be finished. The most *pessimistic forecast* is from the worst performing iterations and an *optimistic forecast* is from the best performing ones.

15. TF Agile in Infrastructure

Agile is specifically suited for infrastructure projects.

Ans: False

16. SAFe

In your own words , describe SAFe.

Scaled Agile Framework (SAFe) is used to try to promote lean, agile practices that allow for vertical/horizontal alignment, collaboration and delivery across many agile teams. This attempts to scale Agile to achieve maximum efficiency when delivery products and services to stakeholders.

17. Why Agile?

Why should a project manager learn AGILE?

Agile allows for simple prioritization of tasks in palatable chunks, helping teams tackle seemingly large and daunting projects without having to focus on the big picture constantly. Agile's iterative nature allows for changes to be factored in while a project's original vision is kept intact, and all members of the team benefit from this. The frequent communication of Agile also allows for solidarity between all members of the team, and often brings a new level of productivity out of a development team.