

## Daily Challenge:

### 1. Explain the following agile methodologies Scrum, Kanban, Extreme Programming.

**Scrum** is an agile process framework for managing and implementing agile methodology also complex knowledge work, with an initial emphasis on software development. It is designed for teams who break their work into goals that can be completed within timeboxed iterations, called sprints, then track progress and re-plan in daily stand-up meetings, called daily scrums.

**Kanban** is an agile methodology that is not necessarily iterative. Processes like Scrum have short iterations which mimic a project lifecycle on a small scale, having a distinct beginning and end for each iteration. Kanban allows the software be developed in one large development cycle.

The principle behind Kanban that allows it to be incremental and Agile, is limited throughout. With no iterations a Kanban project has no defined start or end points for individual work items; each can start and end independently from one another, and work items have no pre-determined duration for that matter. Instead, each phase of the lifecycle is recognized as having a limited capacity for work at any one time. A small work item is created from the prioritized and unstated requirements list and then begins the development process, usually with some requirements elaboration. A work item is not allowed to move on to the next phase until some capacity opens up ahead. By controlling the number of tasks active at any one time, developers still approach the overall project incrementally which gives the opportunity for Agile principles to be applied.

**Extreme Programming** is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.

### 2. Who are the members of an agile team and what are their roles?

**Development Team:** These are the people building the product. It consists of dedicated professionals who can develop, test, and deploy a Story, Feature, or component. The Dev Team typically includes software developers and testers, engineers, and other dedicated specialists required to complete a vertical slice of functionality.

## Roles

- Collaborating with the Product Owner to create and refine user stories and acceptance criteria.
- Participating in PI Planning and creating Iteration plans and Team PI Objectives.
- Developing and committing to Team PI Objectives and Iteration goals.
- Working with the Product Owner to confirm that the code and acceptance tests reflect the desired functionality; writing the code.
- Conducting research, design, prototyping and other exploration activities.
- Creating unit tests and automated acceptance tests.
- Using design and coding best practices to build high-quality components and solutions.
- Checking new code into the shared source code repository Pairing to write code and automated acceptance test cases.
- Executing acceptance tests and maintaining the test cases in a shared repository.
- Continuous improvement of the team's process

**Scrum Master:** Scrum Masters are servant leaders and coaches for an Agile Team. They help educate the team in Scrum, Extreme Programming (XP), Kanban, and SAFe, ensuring that the agreed Agile process is being followed. They also help remove impediments and foster an environment for high-performing team dynamics, continuous flow, and relentless improvement.

The Scrum Master role is a unique Agile team member who spends much of her time helping other team members communicate, coordinate, and cooperate; generally, this person assists the team in meeting their delivery goals.

## Roles

- Exhibits Lean-Agile leadership
- Supports the team rules.
- Facilitates the team's progress toward team goals
- Leads team efforts in relentless improvement
- Facilitates all team meetings, including (where applicable) the Daily Stand-up, Iteration Planning, Iteration Review, and Iteration Retrospective.
- Supports the Product Owner – The Scrum Master helps the Product Owner in their efforts to manage the backlog and guide the team while facilitating a healthy team dynamic with respect to priorities and scope.
- Eliminates impediments – Many blocking issues will be beyond the team's authority or may require support from other teams. The Scrum Master actively addresses these issues so that the team can remain focused on achieving the objectives of the Iteration.
- Protects and communicates – Communicates with management and outside stakeholders; helps protect the team from uncontrolled expansion of work.
- Coordinates with other teams – The Scrum Master is typically the representative in the Scrum of Scrums (SoS) meeting, and they pass information from that meeting back to the

team (see Program Increment for more details). They often coordinate with the System Team, User Experience, Architecture, and Shared Services. It is important to note, however, that the responsibility for inter-team coordination cannot be delegated entirely to the Scrum Master; every team member shares responsibility in that regard.

- Facilitates preparation and readiness for ART events – Assists the team in preparation for ART activities, including PI Planning, System Demos, and the Inspect and Adapt.
- Supports estimating – Guides the team in establishing normalized estimates and helps the team understand how to estimate Features and Capabilities.

**Product Owner:** The Product Owner is a member of the Agile Team responsible for defining Stories and prioritizing the Team Backlog to streamline the execution of program priorities while maintaining the conceptual and technical integrity of the Features or components for the team.

The product owner is the member of the Agile team who serves as the Customer proxy responsible for working with Product Management and other stakeholders including other product owners to define and prioritize stories in the team backlog. This allows the Solution to effectively address program priorities (features and Enablers) while maintaining technical integrity. Ideally, the product owner is collocated with the rest of the team, where they typically share management, incentives, and culture. But the product owner also attends most relevant Product Management meetings about planning and Program Backlog/Vision refinement.

## Roles

- Maintaining the team backlog – With input from System Architect/Engineering and other stakeholders, the product owner has the primary responsibility for building, editing, and maintaining the team backlog. Consisting mostly of user stories, it also includes defects and enablers. Backlog items are prioritized based on user value, time, and other team dependencies determined in the PI planning meeting and refined during the PI.
- Iteration Planning – The product owner reviews and reprioritizes the backlog as part of the prep work for Iteration Planning, including coordination of dependencies with other product owners. During the iteration planning meeting, the product owner is the primary source for story detail and priorities and has the responsibility of accepting the final iteration plan.
- Just-in-time story elaboration – Most backlog items are elaborated into user stories for implementation. This may happen before the iteration, during iteration planning, or during the iteration. While any team member can write stories and acceptance criteria, the product owner has the primary responsibility for maintaining the flow. It's usually good to have approximately two iterations' worth of stories ready in the team backlog at all times. More would create a queue, while less might inhibit flow.
- Accepting stories – The product owner is the only team member who can accept stories as done. This includes validation that the story meets acceptance criteria and has the appropriate, persistent acceptance tests, and that it otherwise complies its Definition of

Done (DoD). In so doing, the product owner also assures a level of quality, focusing primarily on fitness for use.

- Participate in team demo and retrospective – As the person responsible for requirements, product owners have an essential role in the team demo, reviewing and accepting stories. They also participate in the Iteration Retrospective, where the teams gather to improve their processes and are active in the Agile Release Train's (ART's) Inspect and Adapt (I&A) workshop.

### **3. List 5 Project Management Tools**

- Slack
- Teamweek
- Toggl
- Trello
- CoSchedule

### **4. List the Phases of Software Development Life Cycle**

- Planning
- Analysis and Requirements
- Design
- Development
- Integration and Testing
- Implementation
- Operations and Maintenance