

Ques1. What is Agile Project Management? How Agile Project

Ans1. Agile Project Management is an iterative approach to planning and guiding project processes that breaks them down into smaller cycles called sprints, or iterations. Agile project is completed in small sections. In Agile software development, for instance, an iteration refers to a single development cycle. Each section or iteration is reviewed and critiqued by the project team, which should include representatives of the project's various stakeholders. Insights gained from the critique of an iteration are used to determine what the next step should be in the project.

Agile teams build rapid feedback, continuous adaptation and QA best practices into their iterations. They adopt practices such as continuous deployment and continuous integration using technology that automates steps to speed up the release and use of products.

Additionally, Agile Project Management calls for teams to continuously evaluate time and cost as they move through their work. They use velocity, burndown and burnup charts to measure their work, instead of Gantt charts and project milestones to track progress.

Agile Project Management does not require the presence

or participation of a project manager. Although a project manager is essential for success under the traditional project delivery methodologies, such as the waterfall model -- where the position manages the budget, personnel, project scope and other key elements -- the project manager's role under APM is distributed among team members.

For instance, the product owner sets project goals, while team members divvy up scheduling, progress reporting and quality tasks. Certain Agile approaches add other layers of management. The Scrum approach, for example, calls for a Scrum Master who helps set priorities and guides the project through to completion.

But project managers can still be used in Agile Project Management. Many organizations still use them for Agile projects -- particularly larger, more complex ones. These organizations generally place project managers in more of a coordinator role, with the product owner taking responsibility for the project's overall completion.

Given the shift in work from project managers to Agile teams, Agile Project Management demands that team members know how to work within the framework. They must be able to collaborate with each other and with

users. They must be able to communicate well to keep projects on track. And they should feel comfortable taking appropriate actions at the right times to keep pace with delivery schedules.

Ques2. Describe different High Tech Communication tools in agile development approach?

Ans2. When agile scrum team members work in different places, you need to set up sophisticated, high-tech communication methods to create a sense of connectedness. When determining which types of high-tech communication tools to support, your most important consideration should be to mitigate the loss of face-to-face discussions. Some tools you can use are

Video conferencing and webcams: These tools can create a sense of being together.

Instant messaging: Although instant messaging doesn't convey nonverbal communication, it is real time, accessible, and easy to use. Several people can also share a session and share files.

Web-based desktop sharing: Especially for the development team, sharing your desktop allows you to highlight issues and updates visually in real time. Seeing the problem is

always better than just talking it out over the phone.

Collaboration websites: These sites allow you to do everything from sharing simple documentation so that everyone has the latest information to using a virtual white board for brainstorming.

Ques3. Describe different Agile Principles for customer

Ans3. Agile approaches focus on customer satisfaction, which makes sense. After all, the customer is the reason for developing the product in the first place.

While all 12 principles support the goal of satisfying customers, principles 1, 2, 3, and 4 stand out for us:

(1) Our highest priority is to satisfy the customer through early and continuous delivery of valuable

(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.

You may define the customer on a project in a number of ways:

In project management terms, the customer is the person or group paying for the project.

In some organizations, the customer may be a client, external to the organization.

In other organizations, the customer may be a project stakeholder or stakeholders in the organization.

The person who ends up using the product is also a

Ques4. What are Agile Principles of quality published by agile

Ans4. The Agile Manifesto is comprised of four foundational values and 12 supporting principles which lead the Agile approach to software development. Each Agile methodology applies the four values in different ways, but all of them rely on them to guide the development and delivery of high-quality, working software.

1. Individuals and Interactions Over Processes and Tools

The first value in the Agile Manifesto is "Individuals and interactions over processes and tools." Valuing people more highly than processes or tools is easy to understand because it is the people who respond to business needs and drive the development process. If the

process or the tools drive development, the team is less responsive to change and less likely to meet customer needs. Communication is an example of the difference between valuing individuals versus process. In the case of individuals, communication is fluid and happens when a need arises. In the case of process, communication is scheduled and requires specific content.

2. Working Software Over Comprehensive Documentation

Historically, enormous amounts of time were spent on documenting the product for development and ultimate delivery. Technical specifications, technical requirements, technical prospectus, interface design documents, test plans, documentation plans, and approvals required for each. The list was extensive and was a cause for the long delays in development. Agile does not eliminate documentation, but it streamlines it in a form that gives the developer what is needed to do the work without getting bogged down in minutiae. Agile documents requirements as user stories, which are sufficient for a software developer to begin the task of building a new function.

The Agile Manifesto values documentation, but it values working software more.

3. Customer Collaboration Over Contract Negotiation

Negotiation is the period when the customer and the product manager work out the details of a delivery, with points along the way where the details may be renegotiated. Collaboration is a different creature entirely. With development models such as Waterfall, customers negotiate the requirements for the product, often in great detail, prior to any work starting. This meant the customer was involved in the process of development before development began and after it was completed, but not during the process. The Agile Manifesto describes a customer who is engaged and collaborates throughout the development process, making. This makes it far easier for development to meet their needs of the customer. Agile methods may include the customer at intervals for periodic demos, but a project could just as easily have an end-user as a daily part of the team and attending all meetings, ensuring the product meets the business needs of the customer.

4. Responding to Change Over Following a Plan

Traditional software development regarded change as an expense, so it was to be avoided. The intention was to develop detailed, elaborate plans, with a defined set of features and with everything, generally, having as high a priority as everything else, and with a large number of many dependencies on delivering in a certain

order so that the team can work on the next piece of the puzzle.

Ques5. Describe how cross-functionality helps in improving the efficiency of agile development?

Ans5. Cross-functional teams are able to easily switch gears and be flexible with requirements because they are aligned on the larger goals of the product and often don't need to go outside of their group to get buy-in.

Agile development also allows for greater automation in building, testing, and deploying. This gives team members the opportunity to spend less time on mundane tasks and more time on bringing their unique expertise to ideation.