GeeksforGeeks
A computer science portal for geeks

Custom Search

Suggest a Topic

Login

Write an Article

Software Engineering | Agile Software Development

Agile is a time-bound, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver all at once.

Why Agile?

Technology in this current era is progressing faster than ever, enforcing the global software companies to work in a fast-paced changing environment. Because these businesses are operating in an ever-changing environment, it is impossible to gather a complete and exhaustive set of software requirements. Without these requirements, it becomes practically hard for any conventional software model to work.

The conventional software models such as Waterfall Model that depends on completely specifying the requirements, designing, and testing the system are not geared towards rapid software development. As a consequence, a conventional software development model fails to deliver the required product.

This is where the agile software development comes to the rescue. It was specially designed to curate the needs of the rapidly changing environment by embracing the idea of incremental development and develop the actual final product.

Let's now read about the on which the Agile has laid its foundation:

Principles:

1. Highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. It welcomes changing requirements, even late in development.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shortest timescale.
4. Build projects around motivated individuals. Give them the environment and the support they need, and trust them to get the job done.
5. Working software is the primary measure of progress.

6. Simplicity the art of maximizing the amount of work not done is essential.

7. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Development in Agile: Let's see a brief overview of how development occurs in Agile philosophy.

- In Agile development, Design and Implementation are considered to be the central activities in the software process.
- Design and Implementation phase also incorporate other activities such as requirements elicitation and testing into it.
- In an agile approach, iteration occurs across activities. Therefore, the requirements and the design are developed together, rather than separately.
- The allocation of requirements and the design planning and development as executed in a series of increments. In contrast with the conventional model, where requirements gathering needs to be completed in order to proceed to the design and development phase, it gives Agile development an extra level of flexibility.
- An agile process focuses more on code development rather than documentation.

Example: Let's go through an example to understand clearly about how agile actually works.
A Software company named **ABC** wants to make a new web browser for the latest release of its operating system. The deadline for the task is 10 months. The company's head assigned two teams named **Team A** and **Team B** for this task. In order to motivate the teams, the company head says that the first team to develop the browser would be given a salary hike and a one week full sponsored travel plan. With the dreams of their wild travel fantasies, the two teams set out on the journey of the web browser. The team A decided to play by the book and decided to choose the Waterfall model for the development. Team B after a heavy discussion decided to take a leap of faith and choose Agile as their development model.

The Development plan of the Team A is as follows:

- Requirement analysis and Gathering – 1.5 Months
- Design of System – 2 Months
- Coding phase – 4 Months
- System Integration and Testing – 2 Months
- User Acceptance Testing – 5 Weeks

The Development plan for the Team B is as follows:

- Since this was an Agile, the project was broken up into several iterations.
- The iterations are all of the same time duration.
- At the end of each iteration, a working product with a new feature has to be delivered.
- Instead of Spending 1.5 months on requirements gathering, They will decide the core features that are required in the product and decide which of these features can be developed in the first iteration.
- Any remaining features that cannot be delivered in the first iteration will be delivered in the next subsequent iteration, based in the priority
- At the end of the first iterations, the team will deliver a working software with the core basic features.

Both the team have put their best efforts to get the product to a complete stage. But then out of blue due to the rapidly changing environment, the company's head come up with an entirely new set of features and want to be implemented as quickly as possible and wanted to push out a working model in 2 days. Team A was now in a fix, they were still in their design phase and did not yet started coding and they had no working model to display. And moreover, it was practically impossible for them to implement new features since waterfall model there is not reverting back to the old phase once you proceed to the next stage, that means they would have to start from the square one again. That would incur them heavy cost and a lot of overtime. Team B was ahead of Team A in a lot of aspects, all thanks to Agile Development. They also had the working product with most of the core requirement since the first increment. And it was a piece of cake for them to add the new requirements. All they had to do is schedule these requirements for the next increment and then implement them.

Advantages:

- Deployment of software is quicker and thus helps in increasing the trust of the customer.
- Can better adapt to rapidly changing requirements and respond faster.
- Helps in getting immediate feedback which can be used to improve the software in the next increment.
- People – Not Process. People and interactions are given a higher priority rather than process and tools.
- Continuous attention to technical excellence and good design.

Disadvantages:

- In case of large software projects, it is difficult to assess the effort required at the initial stages of the software development life cycle.
- The Agile Development is more code focused and produces less documentation.
- Agile development is heavily depended on the inputs of the customer. If the customer has ambiguity in his vision of the final outcome, it is highly likely for the project to get off track.
- Face to Face communication is harder in large-scale organizations.
- Only senior programmers are capable of taking the kind of decisions required during the development process. Hence it's a difficult situation for new programmers to adapt to the environment.

Agile is a framework which defines how the software development needs to be carried on. Agile is not a single method, it represents the various collection of methods and practices that follow the value statements provided in the manifesto. Agile methods and practices do not promise to solve every problem present in the software industry (No Software model ever can). But they sure help to establish a culture and environment where solutions emerge.

Naveen_Naidu
Check out this Author's contributed articles.

If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please Improve this article if you find anything incorrect by clicking on the "Improve Article" button below.

Improved By : Naveen_Naidu

Article Tags : Software Engineering

👍

Be the First to upvote.

2.5

Based on **2** vote(s)

☐ To-do ☐ Done

(Feedback) (Add Notes) (Improve Article)

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

[Load Comments]   [Share this post!]

A computer science portal for geeks

710-B, Advant Navis Business Park,
Sector-142, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

About Us
Careers
Privacy Policy
Contact Us

LEARN

Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

PRACTICE

Company-wise
Topic-wise
Contests
Subjective Questions

CONTRIBUTE

Write an Article
Write Interview Experience
Internships
Videos