Devops VS Agile

What is agile?

Agile is a broader methodology that encompasses various approaches, such as Scrum, Kanban, and Extreme Programming (XP). Agile emphasizes iterative and incremental development, flexibility, and customer collaboration. It promotes adaptive planning and cross-functional teams, enabling faster responses to changing requirements and delivering valuable software in shorter cycles.*We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:*

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

That is, while there is value in the items on the right, we value the items on the left more

The manifest is paired with 12 agile principles to help make better decisions.

What are the benefits of agile?

- **Agility** - More quickly respond to market changes or customer feedback.
- **Quality** - A focus on testing and sustained customer involvement means the chances of a product's overall quality being high are greater.

- **Collaboration** - Agile is about people. By placing value on human interactions over processes and “that’s just the way it’s done,” organizations are able to let employees act guided by their experience and a shared set of values rather than being micro-managed or shackled to detailed documentation.
- **Ownership** - The trust required from leadership to have agile teams can create an increased feeling of ownership.
- **Customer satisfaction** - With a focus on finding and fixing problems quickly and a direct line between customers and developers, customers are more likely to be satisfied and come back for more.
- **Predictability** - By doing away with big plans that are set in stone and often detached from reality, organizations can get a better picture of what’s going well and what’s not working right away rather than months down the road when it’s too late to do anything to correct it.
- **Increase productivity** - Regularly used planning tools like Kanban and Scrum help teams keep tabs of work and progress toward goals.

What is DevOps?

With a name that sounds like a covert military team and the kind of goofy capitalization you’ve come to expect from tech terminology, DevOps combines software development and IT operations. Take the "dev" from software development and the "ops" from IT operations and you get this portmanteau, which is the fun-to-say term that describes mashing together of multiple words, like spork, phablet, brunch, jorts, and bromance.

DevOps isn't just a process. It’s a shift in workplace culture. It’s a collaboration between teams. Doubling down on automation and installing all of the right software won’t get you there alone. Like agile, people are the key component.

What are the benefits of DevOps?

DevOps is all about producing higher-quality software faster and saving a lot of time and money. Here’s a more detailed breakdown of the benefits.

- **Speed** - Release updates and new features faster, adapt to the changing market, and become more efficient.
- **Rapid delivery** - Increase deployment frequency and the pace of releases. Respond to customers' needs faster and build a competitive advantage.
- **Reliability** - Automatic testing is built-in. Ensures rollouts are of the highest quality and that you have less downtime because you build for stability and test before deploying.
- **Scale** - Implements automation. With the use of cloud and container technology, you can scale usage up and down and save money while you do so.
- **Collaboration** - Allow teams who used to be apart to work together. Workflows can be combined, inefficiency is reduced, and time is saved.
- **Security** - Allows infrastructure to be created by code, just like software. By doing this (instead of doing it manually), you can define policies to stay compliant no matter how many servers will be deployed.

- **Difference between Agile and DevOps**

Parameter	Agile	DevOps
What is it?	Agile refers to an iterative approach which focuses on collaboration, customer feedback, and small, rapid releases.	DevOps is considered a practice of bringing development and operatio

Parameter	Agile	DevOps
Purpose	Agile helps to manage complex projects.	DevOps central concept is to manage end-to-end engineering processe
Task	Agile process focusses on constant changes.	DevOps focuses on constant testing and delivery.
Implementation	Agile method can be implemented within a range of tactical frameworks like a sprint, safe and scrum.	The primary goal of DevOps is to focus on collaboration, so it doesn't have any commonly accepted framework.
Team skill set	Agile development emphasizes training all team members to have a wide variety of similar and equal skills.	DevOps divides and spreads the skill set between the development and operation teams.
Team size	Small Team is at the core of Agile. As smaller is the team, the fewer people on it, the faster they can move.	Relatively larger team size as it involves all the stack holders.
Duration	Agile development is managed in units of "sprints." This time is much less than a month for each sprint.	DevOps strives for deadlines and benchmarks with major releases. The ideal goal is to deliver code to production DAILY or every few hours.
Feedback	Feedback is given by the customer.	Feedback comes from the internal team.
Target Areas	Software Development	End-to-end business solution and fast delivery.
Shift-Left Principles	Leverage shift-left	Leverage both shifts left and right.
Emphasis	Agile emphasizes on software development methodology for developing software. When the software is developed and released, the agile team will not care what happens to it.	DevOps is all about taking software which is ready for release and deploying it in a reliable and secure manner.
Cross-functional	Any team member should be able to do what's required for the progress of the project. Also, when each team member can perform every job, it increases understanding and bonding between them.	In DevOps, development teams and operational teams are separate. So, communication is quite complex.
Communication	Scrum is most common methods of implementing Agile software development. Daily scrum meeting is carried out.	DevOps communications involve specs and design documents. It's essential for the operational team to fully understand the software release and its hardware/network implications for adequately running the deployment process.
Documentation	Agile method is to give priority to the working system over complete documentation. It is ideal when you're flexible and responsive. However, it can hurt when you're trying to turn things over to another team for deployment.	In the DevOps, process documentation is foremost because it will send the software to the operational team for deployment. Automation minimizes the impact of insufficient documentation . However, in the development of complex software, it's difficult to transfer all the knowledge required.
Automation	Agile doesn't emphasize on automation. Though it helps.	Automation is the primary goal of DevOps. It works on the principle to maximize efficiency when deploying software.
Goal	It addresses the gap between customer need and development & testing teams.	It addresses the gap between development + testing and Ops.
Focus	It focuses on functional and non-function readiness.	It focuses more on operational and business readiness.
Importance	Developing software is inherent to Agile.	Developing, testing and implementation all are equally important.
Speed vs. Risk	Teams using Agile support rapid change, and a robust application structure.	In the DevOps method, the teams must make sure that the changes which are made to the architecture never develop a risk to the entire project.
Quality	Agile produces better applications suites with the desired requirements. It can easily adapt	DevOps, along with automation and early bug removal, contributes to creating better quality. Developers need to follow Coding and Architectural best practices

Parameter	Agile	DevOps
	according to the changes made on time, during the project life.	to maintain quality standards.
Tools used	JIRA, Bugzilla, Kanboard are some popular Agile tools.	Puppet, Chef, TeamCity OpenStack, AWS are popular DevOps tools.
Challenges	The agile method needs teams to be more productive which is difficult to match every time.	DevOps process needs to development, testing and production environments to streamline work.
Advantage	Agile offers shorter development cycle and improved defect detection.	DevOps supports Agile’s release cycle.