Jonah Aney 10/25/25 CSD380 Module-1.3 The History of DevOps

In the world of technology, many facets often need to come together to unsure a complex application performs as it should on a daily basis. In an effort to deliver reliable software fast and more effirciently, the DevOps philosophy was born. Its origin comes from three major movements: Lean, Agile, and Continuous Delivery. Each help shape how organizations improve efficiency, collaboration, and automation, forming the DevOps culture and practices we know today.

The Lean Movement

The Lean movement originated in manufacturing and the auto industry. As the world progressed the same philosophy later influenced how teams build software. According to the Lean Enterprise Institute (2024), Lean principles date back to Henry Ford's moving assembly line in 1913 and later evolved through Toyota's Production System, which was led by Kiichiro Toyoda and Taiichi Ohno. Lean focuses on defining value from the customer's perspective, mapping the value stream, creating flow, enabling pull, and pursuing perfection. These principles also emphasize eliminating waste and continuously improving processes (Lean Enterprise Institute, 2024).

When applied to technology, Lean thinking helps reduce bottlenecks between development and deployment. DevOps adopts these ideas to create efficient value streams, smaller batch sizes, and a culture focused on efficiency and continous improvement.

The Agile Manifesto

In 2001, seventeen software practitioners met in Snowbird, Utah, and created the Agile Manifesto. The manifesto followed a set of principles and placed the customer as it's highest priority. It also valued, collaboration, adaptability, and used working software as the primary measure of progress (Agile Alliance, 2001). Agile introduced short iterations, feedback loops, and cross-functional teamwork to make development more flexible and responsive to change.

Agile development improved how teams created software but the process ended on the development side and didn't involve much with deployment. DevOps extends Agile principles into operations by connecting development, testing, and delivery so that changes can move easily amongst teams and departments.

Jonah Aney 10/25/25 CSD380 Module-1.3 The History of DevOps

The Continuous Delivery Movement

According to Octopus Deploy (2021), continuous delivery (CD) is centered on the delivery pipeline with it's principles generally aligned with the Agile Manifesto. While Lean defines flow and Agile supports collaboration, CD provides the technical discipline by focusing on build quality, constantly improving, utilizing automation, and working in small batches.

CD connects Lean efficiency and Agile adaptability with automation which helps minimize time to deploy and time to mitigate production issues. As Microsoft Learn (2024) explains, CD automates building, testing, and deploying software from development to production. It "optimizes process time and eliminates idle time," allowing teams to deliver updates frequently and safely. CD's goal is to keep production fresh with the fastest path from new code to deployment.

Combine the efficiency of a lean workflow, agiles' flexibility, and the automation of continuous delivery to create a process and culture of innovation and discovery. The constant analysis and feedback ensures teams production quality and volume are at an optimium.

Sources:

Agile Alliance. (2001). History: Manifesto for Agile Software Development. AgileManifesto.org. https://agilemanifesto.org/history.html

Agile Alliance. (2001). Principles: Manifesto for Agile Software Development. AgileManifesto.org. https://agilemanifesto.org/principles.html

Lean Enterprise Institute. (2024). A brief history of Lean. https://www.lean.org/explore-lean/a-brief-history-of-lean/

Microsoft Learn. (2024). *What is continuous delivery?* https://learn.microsoft.com/en-us/devops/deliver/what-is-continuous-delivery

Octopus Deploy. (2021, August 26). *Lean, Agile, and Continuous Delivery*. https://octopus.com/blog/lean-agile-continuous-delivery