**Lab 12: DevOps Research Assignment, Team 3.  
  
Group Members: Maruf Rahman, Johnny Zeng, Priyanka Choudhary, Nina Zavrazhina, Reena Tailor, Nyal Phillip, Leuul Asaye, Donna Blyth.**  
  
Q.1: **What is DevOps?**

DevOps (A mixture of “software DEVelopment” and “information technology OPerationS”) is useful for bridging the communication gaps between the developers and the operations team. Where the Developers are interested in evolution and the operations team is mainly looking for stability, DevOps are the Mixture of the two teams philosophy to help these two teams work together.

Q.2: **How do ALM and DevOps differ?**

Modern ALM software today is designed for Agile and DevOps and is built for implementing all manner of Agile Frameworks, including SAFe and extending SAFe for DevOps. DevOps software is specifically for Agile, and typically does not support Hybrid models of development. Also, ALM software solutions that are designed for implementing the Scaled Agile Framework (SAFe). This is because SAFe is designed to accommodate the concept of DevOps, therefore the tools that support SAFe should also support DevOps.

DevOps tools can complement the role that ALM software provides in some circumstances and should only be considered with this particular aim in mind, where additional functional.

When reading the feature description of an ALM software you might well think you are reading about a DevOps vendor and that is because a modern Agile ALM is a DevOps solution.  
  
Most, if not all, DevOps tool features are already included in the modern ALM software.

Q.3: **What is CI?**

Continuous Integration is a practice that developers use to combine developer all working copies to a shared mainline, allowing them to integrate code frequently. They continuously check in and the check-ins are verified by an automated build. This allows teams to detect problems early. This also requires no tool to deploy.

Q.4: **What is CD?**

Continuous Delivery (CD for short) is a method in which a team creates software, making sure that the software created can be accurately released at any time, in short cycles. This aims at building, testing, and releasing software faster and more often. By allowing for more steady updates to applications in production, this approach assists to reduce the cost and time. A simple, effortless and repeatable deployment process is important for continuous delivery.

Q.5: **How is the SDLC in DevOps and ALM similar/different within Agile frameworks?**

The whole application lifecycle from the initial application definition to application maintenance, retirement and everything in between is managed by the ALM. It’s best to think of ALM as a branch, which could include one or even more SDLCs. We could compare this to a file management structure, upon which the ALM as the root folder, and the SDLC as a child folder.

Now the SDLC in DevOps and ALM are a bit different compared to Agile frameworks, one of the main differences is that Automation is more prevalent in DevOps, while the Agile Framework is more of a consistent manual progress, updated by person as progress advances.

Agile software development advocates adaptive planning, evolutionary development, early delivery, and continuous improvement, and it encourages rapid and flexible responses to change. To make this happen, the entire development life cycle should be optimized. As for optimization, wherever it's possible, automation the key.

DevOps automate everything right from the planning phase to the release phase.