Continuous Delivery and Integration

Course Summary

Description

Continuous Delivery and Integration, more commonly referred to as DevOps, is a current development methodology that is derived from Agile methodologies but is enabled with a number of current technologies like micro-services, cloud computing and visualization.

DevOps is a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality.

DevOps primary model is based on the idea that there should not be separate development and deployment (or operations) phases in a software development project. Software is architected and development processes designed so that enhancements, additions, and changes to the software are continuously being made and deployed. This continuous delivery means allows for very short delivery times and faster responses to changing requirements. Continuous integration means that the software is no longer seen as a single stand-alone fixed release, but is made up of a number of pluggable components which can all be updated independently and then seamlessly integrated into the existing application. To be successful with this model, there is a very strong emphasis on continuous automated testing and quality control throughout the whole development and delivery process.

DevOps requires a high level of process maturity with a strong focus on monitoring and automating all the dimensions of the software development process including continuous testing, software construction, release management, and infrastructure management. DevOps is also made possible by choosing software architectures that allow for continuous integration, testing, and delivery such as building applications as a set of micro-services and using containerization for "run anywhere" components.

This course is a one day overview of the principles, techniques, and tools that make up the DevOps paradigm with an emphasis on how DevOps is currently being practiced, the benefits, and risks of moving to a DevOps model and the human and technical infrastructure that is necessary to be successful with DevOps.

A more technical course with hands-on can also be done but it would probably take about three days.

Topics

- Basic DevOps principles: the unifying of development and operations into a single environment.
- How continuous delivery, integration and testing works – the basic model.
- Phases: Code, Build, Test, Package, Release, Configure, Monitor.
- How DevOps differs from Agile and other methodologies
- The importance of continuous quality metrics – automated testing from requirements through delivery.
- Tools typically used in each DevOps phase: eg. Jenkins, ELK Stack, Docker, Anisble, git, etc.
- Architecturally significant requirements needed for DevOps: Depolyability, Testability, etc.
- Continuous integration architecture: Eg. Micro-services, cloud computing.
- What a DevOps team looks like.
- Organizational and infrastructure issues to address in implementing DevOps.
- Current state and future of DevOps
- Common DevOps pitfalls and established DevOps best practices.
- DevOps variants: SciOps and BizOps.

Continuous Delivery and Integration

Course Summary (cont'd)

Audience

This course is intended for anyone who has an interest in DevOps.

Prerequisites

Before taking this course, basic understanding of software development concepts is useful. There are no prerequisites.

Duration

One day