API's, Microservices, Cloud-Native en Serverless

Duration: 1 Day   Course Code: GKAMCS

Overview:

In this APIs, Microservices, Cloud-native – Docker and Serverless training you will learn the latest trends in software development. If you develop your own software or outsource software development, the question remains: which design pattern to choose? What are the differences, the advantages, and disadvantages?

You will learn the best practices for software design, evolved from Service Oriented Architecture (SOA), from APIs to Microservices. Influenced by the Cloud we now have Cloud-native application deployment / installation methods like containers / Docker / Service Mesh and an immutable (unchangeable) infrastructure (throw away and install again). Finally, you can choose for Serverless, with management of infrastructure and middleware completely in control of the Cloud Service Provider (CSP).

This training is also part of the 5-day Masterclass Digital Transformation (GKMDT)

Target Audience:

Anyone who wants to know about the latest trends in software development

Objectives:

- After completing this course you should be able to:

- Understand the main characteristics of SOA: loosely Coupling, the concept of a contract for your interface, reusability

- Explain the features of a RESTful API

- Identify the challenges with APIs

- Understand the need for a Dynamic Service Discovery

- Explain the features of an API Gateway

- Compare an API with a Microservice

- Identify the challenges with API chains

- Understand the importance of observability and Tracing

- Explain the features of an application container

- Understand the popularity of Docker

- Identify the Ecosystem of Docker

- Explain the features of Kubernetes

- Describe the 12-factor App standard

- Understand *The Microservices hierarchy of needs*

- Understand the advantages and pitfalls of Docker - Kubernetes

- Explain the features of a Service Mesh

- Compare Docker and a Service Mesh solution

- Explain the impact on your organization when using Docker and a Service Mesh

- Understand the advantages and pitfalls of a Service Mesh

- Explain the concept of Serverless

- Understand the different implementations of Serverless: Function as a Service (FaaS) and Backend as a Service (BaaS)

- Identify what the major Cloud Service Providers offer

- Understand why Serverless does NOT mean NoOps

Prerequisites:

-
- Basic IT Knowledge
Development skills are not needed for this training.

Follow-on-Courses:

The following courses are recommended for further study.

■ The latest trends in Cloud security: CASB, SD-WAN and SASE Training

Content:

■ What are the main characteristics of SOA: loosely Coupling, the concept of a contract for your interface, reusability?
■ What are the characteristics of a RESTful API?
■ What are the advantages and disadvantages of an API?
■ Why is a dynamic service discovery important?
■ API Gateways
■ What are differences between an API and a Microservice?
■ The problem with (long) API chains

■ The importance of SRE Observability, Tracing
■ What is an application container?
■ What are the requirements for an application to run in an application container
■ The 12-factor App
■ What is Docker and Kubernetes
■ The *Microservices hierarchy of needs*
■ Differences between Cloud Service Providers (CSP)

■ What is a Service Mesh?
■ Advantages and pitfalls of a Service Mesh
■ What is Serverless?
■ What is the difference between Function as a Service (FaaS) and Backend as a Service (BaaS)
■ What do the CSP's offer: AWS Lambda, Azure Functions etc.
■ NoOps?

Further Information:

For More information, or to book your course, please call us on 0800/84.009

info@globalknowledge.be

www.globalknowledge.com/en-be/