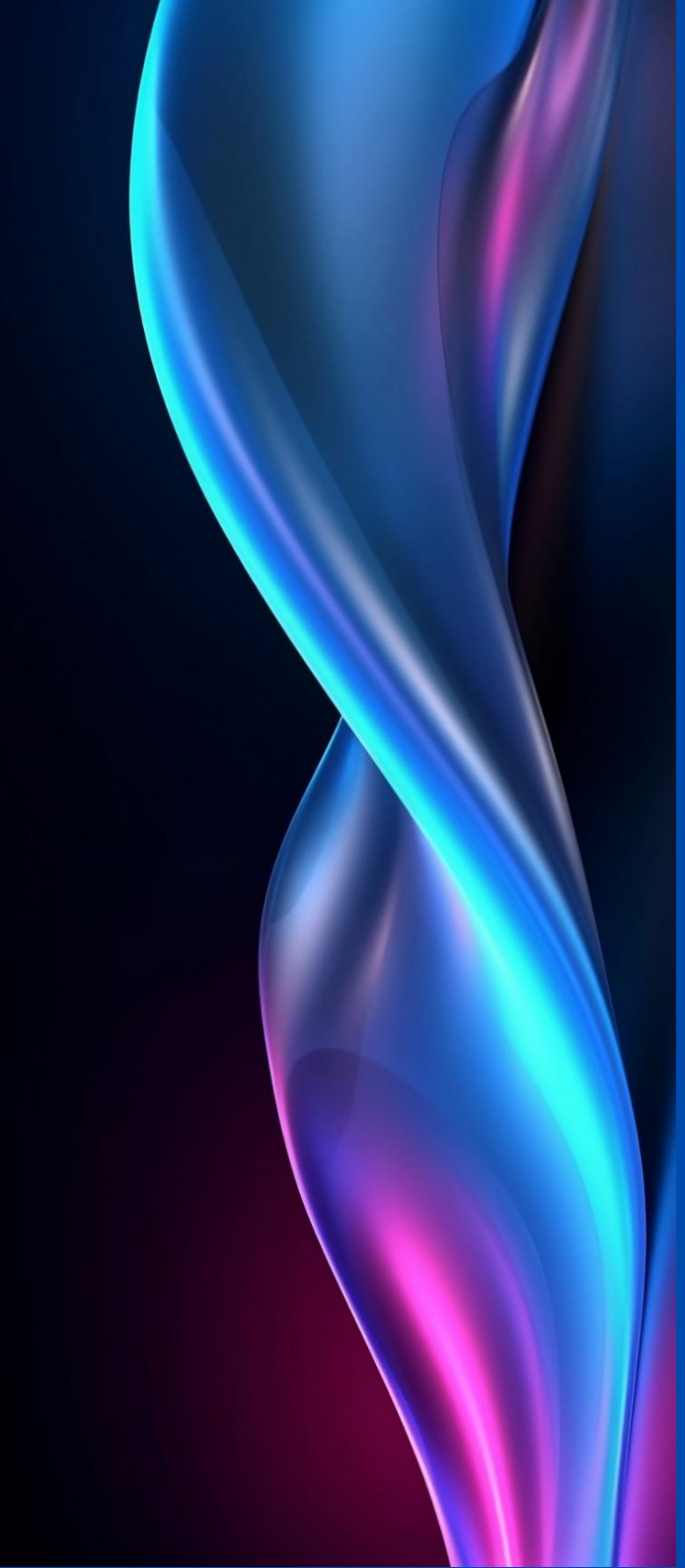

Technology + Knowledge = Innovation

Via Copenaghen, 10 Roma

www.vivasoft.it

info@vivasoft.it

Vivasoft

Consulting & Training

Microsoft

Vivasoft is a leading company in the technology sector, specializing in offering innovative solutions based on Microsoft. With years of experience in the market, we are proud to be certified Microsoft partners, committed to supporting the digital transformation of businesses. We provide a comprehensive range of Microsoft products and highly qualified training courses, designed to help companies optimize their processes, enhance productivity, and acquire advanced skills in the world of technology.

kubernetes

Microsoft

Module 1: Introduction to Kubernetes

What is Kubernetes and why to use it
Definition of Kubernetes and overview of the ecosystem
Problems solved by Kubernetes: container management, scalability, high availability Kubernetes architecture and major components
Master Node and Worker Nodes
Kube-apiserver, kube-scheduler, kube-controller-manager, etcd Kubelet and Kube-Proxy
Difference between Docker and Kubernetes
Kubernetes vs. other orchestration solutions (Mesos, Docker Swarm, etc.) Installation of Kubernetes
How to install Kubernetes on local machine (Minikube, K3s, Docker Desktop) Creating a Kubernetes cluster on cloud (GKE, EKS, AKS)
Installing kubectl, the Kubernetes command line interface Configuring the development environment for Kubernetes
Overview of the Kubernetes interface
Introduction to kubectl commands (e.g., kubectl get, kubectl describe, kubectl logs) Exploration of the main Kubernetes objects: Pods, Deployments, Services, ConfigMaps, Secrets

kubernetes

Microsoft

Modulo 2: Fondamenti di Kubernetes

Concetto di Pod in Kubernetes
Cos'è un Pod e a cosa serve
Differenza tra Pod e container
Creazione e gestione di Pods con kubectl
Gestione dei logs e dello stato dei Pods
Concetto di ReplicaSet e come Kubernetes garantisce l'alta disponibilità dei Pods
Gestione dei Deployments
Creare e gestire un Deployment
Come funzionano le strategie di aggiornamento (Rolling Updates, Blue-Green, Canary Releases)
Esegui il rollback di un Deployment
Visualizzare lo stato dei Deployment e dei Pods associati
Servizi in Kubernetes (Services)
Cos'è un Service e come mappa i Pods
Creare un Service di tipo ClusterIP, NodePort, LoadBalancer, e ExternalName
Differenze tra i vari tipi di Service e quando usarli
Utilizzare un Service per comunicare tra Pods
Namespaces in Kubernetes
Cos'è un Namespace e come organizza le risorse in Kubernetes
Creare e gestire i Namespaces
Utilizzare i Namespaces per il multi-tenancy

Module 3: Volume Management and Persistence

Storage in Kubernetes
Introduction to Volumes and how Kubernetes manages persistent storage
Differences between Volumes, Persistent Volumes (PV), and Persistent Volume Claims (PVC)
Creating and managing Volumes and PVCs
Storage Classes and volume provisioning dynamics
Difference between cloud storage (EBS, GCE, etc.) and local storage
ConfigMaps and Secrets
What is a ConfigMap and how it handles configurations in Kubernetes
What is a Secret and how it handles sensitive data (passwords, API keys)
Creating and using ConfigMaps and Secrets within StatefulSets Pods
Introduction to StatefulSets for stateful applications
Difference between Deployment and StatefulSets
How to manage the lifecycle of stateful applications in Kubernetes

Module 4: Networking and Communication between Pods

Networking in Kubernetes
Kubernetes network architecture: Pod-to-Pod communication, Service-to-Pod communication
The main networking concepts: CNI (Container Network Interface)
Differences between ClusterIP, NodePort and LoadBalancer
How DNS works in Kubernetes and use of domain names for Services Ingress and
Ingress Controllers
What is an Ingress and how it is used to expose applications over HTTP(S)
Configuring an Ingress and choosing an Ingress Controller (NGINX, Traefik,
HAProxy)
Routing and traffic management with Ingress
Network Policies
Introduction to Network Policies (Network Policies)
How to define and enforce access rules between
Pods Network protection and segmentation

Module 5: Managing Jobs and CronJobs

Jobs in Kubernetes
What is a Job and how it handles batch and one-time task execution
Creating and monitoring Jobs in Kubernetes
Manage the completion behavior and re-execution attempts of CronJobs Jobs in Kubernetes
What is a CronJob and how to run scheduled tasks (similar to cron in Linux)
Create and manage CronJobs
Configure scheduling and maximum number of simultaneous executions

Module 6: Security Management in Kubernetes

Authentication and Authorization
Concepts of authentication and authorization in Kubernetes
Creating and managing users and roles (RBAC - Role Based Access Control) Implementing access policies with RBAC and Service Accounts Pod security
Protecting Pods and containers: Pod Security Policies (PSP)
Use securityContext to control Pods' resources and privileges
Configure AppArmor, SELinux and other security measures
Protecting Sensitive Data.
Using Secrets and secure management of sensitive data (such as passwords and private keys)
Manage secure access to persistent volumes

Module 7: Monitoring, Logging and Troubleshooting

Cluster Monitoring.
Overview of monitoring tools for Kubernetes (Prometheus, Grafana, etc.) How to collect metadata and metrics from Kubernetes
Configure monitoring and view metrics in real time Logging in Kubernetes
How to collect and visualize container logs in Kubernetes
Integrating logging tools such as ELK Stack (Elasticsearch, Logstash, Kibana) or Fluentd
Troubleshooting of Pods logs
Debugging and Troubleshooting of Pods
How to analyze the behavior of failed or crashing Pods
Diagnosing and troubleshooting with kubectl logs, kubectl describe, and kubectl exec
Advanced troubleshooting tools in Kubernetes

Module 8: Updates and Resource Management

Cluster Management and Updates
Kubernetes cluster upgrade strategies (rolling upgrades, Blue-Green, Canary)
Managing compatibility between Kubernetes versions
Upgrading major cluster components (kube-apiserver, kubelet, etc.)
Auto-scaling in Kubernetes
Use the Horizontal Pod Autoscaler (HPA) to automatically scale Pods based on metrics (CPU, memory, etc.)
Use the Cluster Autoscaler to scale the number of nodes in the cluster
Dynamic scaling of Deployments and StatefulSets
Cluster resource management
Limiting resource utilization (CPU, memory) for Pods
Configuring required resources and container limits
Monitoring and managing resources in the cluster to optimize efficiency

Module 9: Advanced Practices and Architecture Considerations

Advanced Kubernetes architecture Multi-region and multi-cluster clusters
Managing networking across multiple Kubernetes clusters
Configuring hybrid and multi-cloud clusters with Kubernetes
Kubernetes and DevOps
Integrating Kubernetes with CI/CD (Continuous Integration/Continuous Deployment) pipelines
Managing releases with Helm
Automation of deployment and upgrade operations in production

Module 10: Practical Exercises and Final Project

Hands-on project: Creating and managing an application on Kubernetes
Create and deploy a multi-tier application on Kubernetes, with replication, autoscaling, and data management
Configure networking, load balancing, monitoring, and logging for the app
Resolve availability, security and performance issues