PUTCHARAPON KAEWMAHANIL 6407012663010
Topic: Tools for DevOps

1. What is DevOps, and what are its goals?

Ans DevOps is defined as a combination of processes and tools created to facilitate organizations in delivering services and applications much faster than they can through conventional software development processes. It helps increase customers' confidence in the applications that an organization offers, thereby allowing the company to flourish and achieve its business goals faster.

Goal of DevOps the fast-paced growth of the IT industry and continuous advancements in technology make it critical to set DevOps goals that are experimental and challenging for companies to compete and thrive in the market. Here are the key goals and principles that every successful DevOps program has in common.

2. What are some common tools used in DevOps?

ANS Some of the most popular DevOps automation tools include Puppet, Chef, Ansible, and Salt Stack. Puppet is a configuration management tool that helps you manage your infrastructure as code.

Topic: RESTAPIs

1. What is a RESTful API?

   ANS A RESTful API is an architectural style for an application program interface (API) that uses HTTP requests to access and use data. That data can be used to GET, PUT, POST and DELETE data types, which refers to the reading, updating, creating and deleting of operations concerning resources. An API for a website is code that allows two software programs to communicate with each other. The API spells out the proper way for a developer to write a program requesting services from an operating system or other application.

2. What are the benefits of using RESTful APIs?

ANS scalability Systems using REST APIs can scale efficiently because REST optimizes client-server interactions. Statelessness eliminates server load because the server does not need to keep historical client request data. Some or all well-managed caching reduces some client-server interaction. All these features support scalability without causing communication bottlenecks that degrade performance.

Flexibility RESTful web services support complete client and server separation.

3.What are the HTTP methods used in RESTful API?

ANS PUT, POST, PUT, DELETE, PUTCH

Topic: Microservices

1. What are microservices?

ANS Microservices are an architectural approach to building applications. As an architectural framework, microservices are distributed and loosely coupled, so one team's changes won't break the entire app. The benefit of using microservices is that development teams can rapidly build new components of apps to meet changing business needs.

2. What are the benefits of using microservices?

ANS Faster time-to-market Developers can plug this new "microsurgery" into the architecture without fear of conflicts with other code or of creating service outages that ripple across the website.

3. What are the drawbacks of using microservices?

ANS Cascading failure is when one of the systems fails. May cause all surrounding systems to crash which as well of course It can happen to any system. But good luck with this problem. It has a clear pattern. We must pay attention to this when designing a microservice. Regarding the change in requirement, we may not care about fixing the service. But we will create a new service or maybe create 3 new services. to replace the old service, causing the service to be a lot but looking back at the meaning of

Microservice, we will find that it is small. Therefore, it is more conducive to us to fix it than to create a new one.

Topic: K8s

1. What is Kubernetes, and what are its goals?

ANS Kubernetes, often abbreviated as “K8s”, orchestrates containerized applications to run on a cluster of hosts. The K8s system automates the deployment and management of cloud native applications using on-premises infrastructure or public cloud platforms. It distributes application workloads across a Kubernetes cluster and automates dynamic container networking needs.

GOAL Kubernetes is a powerful open-source container orchestration platform that provides several benefits for organizations looking to deploy and manage containerized applications at a scale. One of the key benefits of Kubernetes is its portability, which allows applications to be easily moved between different environments without the need for modifications.

2. How does Kubernetes handle scaling and load balancing?

ANS Kubernetes offers two convenient abstractions to deploy apps: Services and Deployments. Deployments describe a recipe for what kind and how many copies of your app should run at any given time.

Each app is deployed as a Pod, and an IP address is assigned to it.

Services, on the other hand, are like load balancers.

They are designed to distribute the traffic to a set of Pods.