

Universidad de Las Américas
Facultad de Ingenierías y Ciencias Agropecuarias
Ingeniería de Software
Informe de laboratorio

1. DATOS DEL ALUMNO:

Nombre: Dylan Clerque

Correo: dylan.clerque@udla.edu.ec

2. TEMA DE LA PRÁCTICA:

Aprender los fundamentos en DevOps, Microservicios y APIs basado en proyectos reales, ingeniería de software continua.

3. OBJETIVO DE LA PRÁCTICA

El objetivo principal de la práctica es aplicar los conceptos aprendidos en el curso sobre DevOps, microservicios y APIs en un entorno de proyecto real. Además, se busca que adquirir experiencia práctica en el diseño, desarrollo y despliegue de microservicios utilizando las mejores prácticas de la industria.

4. OBJETIVOS ESPECÍFICOS

Comprender los principios y beneficios de la arquitectura de microservicios.

Diseñar microservicios que cumplan con los requisitos funcionales y no funcionales del sistema.

Implementar prácticas de integración continua y entrega continua (CI/CD) para automatizar el proceso de desarrollo y despliegue.

Desarrollar APIs RESTful para permitir la comunicación entre los microservicios.

Utilizar herramientas de monitoreo y registro para garantizar el rendimiento y la disponibilidad del sistema.

5. MATERIALES Y MÉTODOS

- Curso virtual de Udemy

6. DESARROLLO DE LA PRÁCTICA Y RESULTADOS

Durante el desarrollo de la práctica, se verá el curso con una duración de 5 horas, para posteriormente poner en práctica todos lo aprendido dentro de este.

Los resultados de este curso son todos los conocimientos que se adquirieron después de completar el mismo, lo cuales serían los siguientes:

Básicamente estos serían algunos de mis apuntes de lo que se vio en el curso.

1. Fundamentos de DevOps:

Conceptos básicos de DevOps y su importancia en el desarrollo de software.

Principios clave de DevOps, como la automatización, la colaboración y la integración continua.

Herramientas populares de DevOps, como Docker, Kubernetes, Jenkins, Git, y Ansible.

Prácticas de DevOps, como la entrega continua, la integración continua y la monitorización.

2. Arquitectura de Microservicios:

Conceptos fundamentales de microservicios y su contraste con la arquitectura monolítica.

Diseño y desarrollo de microservicios independientes y escalables.

Implementación de patrones de comunicación entre microservicios, como REST, gRPC o mensajería asíncrona.

Estrategias de despliegue y gestión de microservicios, incluyendo contenerización con Docker y orquestación con Kubernetes.

3. Diseño y Desarrollo de APIs:

Fundamentos de las APIs y su importancia en la integración de sistemas.

Diseño de APIs RESTful siguiendo principios de buenas prácticas y estándares.

Desarrollo de servicios web utilizando frameworks populares como Flask (Python), Express (Node.js), o Spring Boot (Java).

Documentación de APIs y gestión de versiones utilizando herramientas como Swagger o OpenAPI.

4. Ingeniería de Software Continua:

Conceptos de ingeniería de software continua y su relación con DevOps.

Implementación de pipelines de integración continua y entrega continua (CI/CD) para automatizar el proceso de desarrollo y despliegue.

Prácticas de automatización de pruebas, incluyendo pruebas unitarias, de integración y de aceptación.

Monitorización y retroalimentación continua para mejorar la calidad del software y la eficiencia del equipo.

5. Proyectos Prácticos:

Trabajar en equipos para diseñar, desarrollar y desplegar aplicaciones basadas en microservicios.

Crear APIs para permitir la comunicación entre los microservicios.

Implementar pipelines de CI/CD para automatizar el proceso de entrega de software.

Utilizar herramientas de monitoreo para garantizar el rendimiento y la disponibilidad del sistema.

7. OPINIÓN PERSONAL

Opinión 1:

El módulo de microservicios fue realmente revelador para mí. Antes del curso, estaba más un poco familiarizado con arquitecturas monolíticas, pero aprender sobre la modularidad y la independencia de los microservicios me abrió los ojos a nuevas posibilidades. Me impresionó cómo la fragmentación de las aplicaciones en servicios más pequeños puede facilitar la escalabilidad y el mantenimiento a largo plazo. Creo que esta comprensión cambiará fundamentalmente la forma en que abordo el diseño de software en el futuro

Opinión 2:

La automatización en DevOps fue una de las partes más emocionantes del curso para mí. Ver cómo las herramientas como Jenkins y Ansible pueden simplificar y acelerar el proceso de desarrollo y despliegue fue increíblemente inspirador. Antes, solía perder mucho tiempo en tareas repetitivas y propensas a errores, pero ahora veo cómo puedo automatizarlas y liberar más tiempo para trabajar en problemas más interesantes. Esta parte del curso realmente me abrió los ojos a las posibilidades de la automatización en el desarrollo de software

Opinión 3:

El diseño de APIs fue un área en la que tenía conocimientos pero no había profundizado antes, a pesar de eso, resultó ser fascinante. Aprender sobre los principios de diseño RESTful y cómo crear APIs bien estructuradas y fáciles de usar fue muy gratificante. Me di cuenta de la importancia de pensar cuidadosamente en el diseño de la interfaz de programación, no solo para facilitar la comunicación entre sistemas, sino también para mejorar la experiencia del usuario. Creo que esta habilidad será invaluable en mi carrera como desarrollador de software

Opinión 4:

El aspecto de seguridad en DevOps fue una revelación para mí. Antes de este curso, mi enfoque en DevOps se centraba principalmente en la automatización y la entrega continua, pero no había considerado adecuadamente la importancia de la seguridad en todo el ciclo de vida del desarrollo de software. Aprender sobre prácticas de seguridad como el escaneo de vulnerabilidades, la gestión de identidades y acceso, y la implementación de prácticas seguras de codificación fue fundamental para mí. Me hizo darme cuenta de que la seguridad no es un complemento opcional, sino un componente crítico en el diseño y la implementación de sistemas de software modernos. Esta comprensión me ha motivado a integrar la seguridad de manera proactiva en todos mis proyectos futuros

8. ANEXOS (Ai se requiere)



9. BIBLIOGRAFÍA (Si se requiere)

<https://udla.udemy.com/course/devops-y-microservicios-fundamentos/learn/lecture/31070746#overview>