Tarea de Arranque Grupo 03

Francisco Jose Correa Rozo, Juan Camilo Muñoz Castelblanco, Javier Alejandro Gómez Muñoz

Análisis de información sobre Big Data - MINE

Universidad de los Andes, Bogotá, Colombia

{fj.correa10, jc.munozc12, ja.gomez1003[}@uniandes.edu.co](mailto:%7d@uniandes.edu.co)

Fecha de presentación: febrero 13 de 2023

Tabla de contenido

[1 Introducción 1](#_Toc127182639)

[2 Arquitectura Planteada 1](#_Toc127182642)

1. [Desarrollo de la tarea: 2](#_Toc127182643)

# Introducción

# Este trabajo presenta una oportunidad única para explorar cómo la tecnología puede ayudarnos a solucionar los desafíos de la gestión de datos masivos. Al utilizar contenedores de Docker, Flask y servicios REST con JSON, podemos ver cómo estas tecnologías pueden trabajar juntas para crear soluciones eficientes y escalables para la gestión de grandes volúmenes de datos.

# Adicionalmente, esta tarea brindó la oportunidad de desarrollar y aplicar habilidades valiosas en tecnologías de vanguardia, adicional de refrescar conocimientos ya adquiridos. La demanda de profesionales capacitados en el manejo de Big Data sigue creciendo, y este trabajo puede ser un primer paso para adquirir las habilidades necesarias para convertirse en un experto en este campo.

# Arquitectura Planteada

En este trabajo, se ha desarrollado una arquitectura robusta y escalable para la gestión de datos masivos que utiliza los siguientes componentes: contenedores de Docker, Flask, servicios REST con JSON y apoyados en un servidor Linux para la capa de presentación.

Los contenedores de Docker se utilizaron para crear y desplegar de forma rápida y segura la aplicación web desarrollada con Flask bajo el puerto 80. Esto permitió una mejor gestión y control de las dependencias y recursos de la aplicación.

Flask, por su parte, fue utilizado para desarrollar la aplicación web y proporcionar una interfaz amigable y accesible para los usuarios. La arquitectura de Flask permite una fácil integración con otros componentes de la aplicación, provee un patrón de desarrollo MVC (Modelo Vista Controlador), lo que hizo posible una gestión eficiente de la presentación de los datos.

Por último, se utilizaron servicios REST con JSON para integrar la capa de datos y permitir una comunicación flexible y eficiente entre los componentes de la aplicación. Con esto se separa la aplicación web en el servidor presentación, del componente de data y procesamiento que posiblemente se pueda ir a una infraestructura distribuida que en futuros ejercicios sea utilizada.

# Desarrollo de la tarea:

Durante la ejecución de la tarea identificamos varias etapas que se fueron ejecutando a lo largo del tiempo propuesto, la primera, comprensión de lo requerido en el enunciado de la tarea; la segunda: preparación del servidor de presentación destinado para estos fines pedagógicos, Docker, Python (librerias), paths, entorno virtual entre otros; la tercera, desarrollo de la solución de los puntos de Python – Unix; la cuarta, construcción del proyecto Web con Flask, se desarrolla Framework modular que permite contener las futuras actividades que se proponen; el quinto, construcción de los servicios Web para integrar la presentación a los servicios de datos; por último, las pruebas y revisión del funcionamiento global de la solución.