



Manual Tecnico

11/11/2023

—

Carlos Alexis Ovalle de Leon
202031064

Información Técnica

Lenguajes

Los lenguajes utilizados han sido, para el frontend, fue el framework de angular, junto algunas librerías como bootstrap y angular material y para el Backend he usado javascript con node js y para el manejo de la base de datos use mongoDB.

Versiones y Entornos

Use Mongo la versión alpine, angular 16.2.1, Node 18, typescript 5.1.6, además de generar contenedores para el manejo del frontend, backend y base de datos a través de Docker.

Información de Desarrollo de Entorno

Como desarrollo, lo realice en un sistema operativo EndeavourOS Linux, desarrollado en el editor de texto Visual Studio Code, a su vez como controlador de versiones utilice git.

Información Complementaria

Node.js: Es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono, con E/S de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.

Angular: Es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

MongoDB: Es un sistema de base de datos NoSQL, orientado a documentos y de código abierto.

En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

Docker:

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.¹ Docker utiliza características de aislamiento de recursos del kernel Linux, tales como cgroups y espacios de nombres (namespaces) para permitir que "contenedores" independientes se ejecuten dentro de una sola instancia de Linux, evitando la sobrecarga de iniciar y mantener máquinas virtuales.

Diagramas:



