**PROYECTO PERSONAL**

**Exchange de Criptomonedas – Api Rest**

**Fecha de inicio:** 18 / diciembre / 2024

**Fecha de finalización:**

**Estado del proyecto:** En desarrollo.

**Demostración del proyecto – Video YouTube:**

**Tecnologías utilizadas:** Java 17 LTS, Spring Boot 3, MySQL 8.2 / Flyway, JPA Hibernate, Maven

**Objetivo:** Plasmar los conocimientos adquiridos en un proyecto de la vida real que me permita seguir practicando y aprendiendo sobre programación; demostrar por medio de un proyecto personal mis habilidades como programadora.

**Características del proyecto:** Sistema que permite gestionar la compra y venta de criptomonedas en un Exchange ficticio.

**Autora:** María Raquel Martínez Urrutia

Estudiante de Ing. en Desarrollo de Software en UnADM

**LinkedIn:** www.linkedin.com/in/raqmartinez

**Email:** [meztinea.dev@gmail.com](mailto:meztinea.dev@gmail.com)

# Requerimientos del Sistema

El Exchange permite a los usuarios registrarse en el sistema para tener una cuenta propia en la que pueden realizar compras y ventas de las criptomonedas disponibles.

## Requerimientos funcionales y no funcionales

Con base en las características del sistema se han identificado los requerimientos funcionales y no funcionales que guiarán la realización del proyecto.

**Requerimientos funcionales**

1. Es necesario que el sistema permita crear y gestionar cuentas de usuario.
2. El sistema debe permitir a los usuarios consultar, depositar y retirar criptomonedas.
3. El sistema debe permitir a los usuarios comprar y vender criptomonedas.
4. El sistema debe mostrar a los usuarios el precio de las criptomonedas en tiempo real.
5. Se debe contar con un historial de transacciones para cada usuario.

**Requerimientos no funcionales**

1. Es necesario que el sistema se conecte con una base de datos para almacenar la información de manera persistente y permita consultar los datos en el futuro.
2. El sistema debe contar con una arquitectura que permita realizar mantenimiento de manera fácil y eficaz.
3. Se debe implementar protección contra ataques por medio de la encriptación de datos sensibles.
4. Es necesario garantizar que la disponibilidad del software sea mayor al 99%.
5. El sistema debe contar con una documentación ampliamente detallada para evitar futuros problemas en actualizaciones o modificaciones.

## Arquitectura

## Base de datos

## Gestión del Proyecto (SCRUM)

## Desafíos del proyecto

En esta sección se incorporan los desafíos con los que he tenido que lidiar al momento de realizar este proyecto.