**DOCUMENTO DE ARQUITECTURA DE SOFTWARE**

PROYECTO DE TIPO CAMBIO

Mario Kevin Huarcaya Monterrey

Tabla de contenido

[1. Introducción 3](#_Toc100298410)

[1.1. Propósito 3](#_Toc100298411)

[1.2. Alcance 3](#_Toc100298412)

[2. Arquitectura del Sistema 3](#_Toc100298413)

[2.1 Diagrama de secuencia 3](#_Toc100298414)

[2.1. Vista Lógica 4](#_Toc100298415)

[2.1.1. Descripción 4](#_Toc100298416)

[2.1.2. Diagrama de clases 4](#_Toc100298417)

[2.2. Vista de Componentes 4](#_Toc100298418)

[2.3. Vista de Datos 5](#_Toc100298419)

[3. Calidad 6](#_Toc100298420)

# Introducción

## Propósito

El Documento de Arquitectura de Software muestra un resumen general sobre la arquitectura del sistema que se tomó en consideración al desarrollar el proyecto*.*

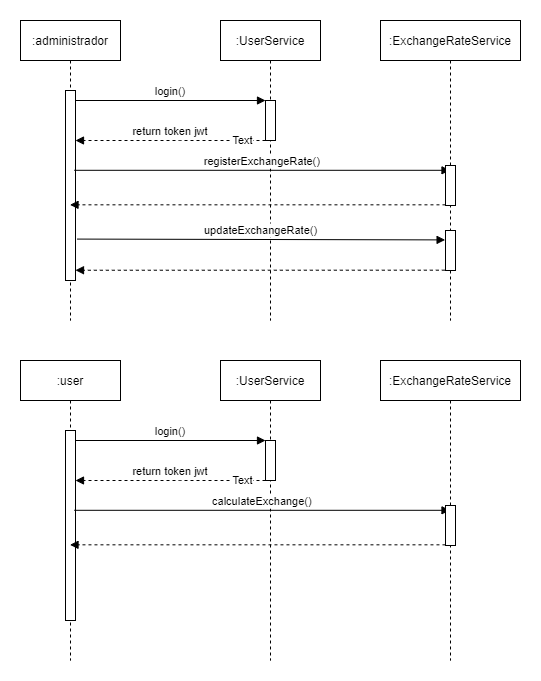
## Alcance

El Documento de Arquitectura de Software contiene la arquitectura del sistema, abarca la definición de la arquitectura del producto a través del diagrama de secuencia, la definición de las capas de la arquitectura, diagrama de clases, componentes y diseño de la base de datos. También se mostrará los resultados de las pruebas de rendimiento.

# Arquitectura del Sistema

## Diagrama de secuencia

*Diagrama de Secuencia*

**

## Vista Lógica

### Descripción

**Capa de Base de Datos**, se encarga de almacenar toda la información, y tenerla disponible en cualquier momento en la aplicación.

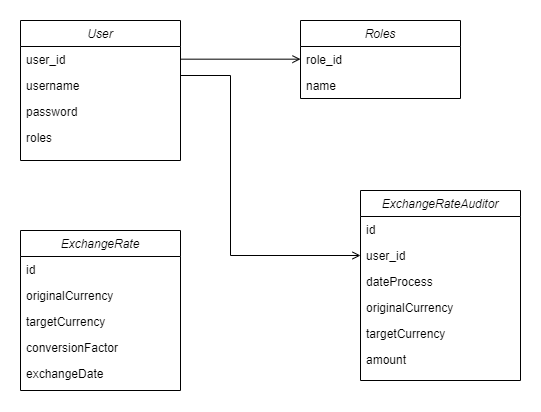
Para esta capa se utilizará como motor de base de datos PostgresSQL dentro de un contenedor docker.

**Capa de Aplicaciones**, Donde se instalan todas las aplicaciones, interactua con el usuario a través del explorador y accede a la base de datos para obtener información. El lenguaje de programación utilizado en esta capa es Java con el framework Spring Boot. La aplicación estará contenia dentro de un contenedor Docker.

**Capa de usuario o cliente**, Para esta capa se utilizará Angular como framework de Javascript dentro de un contenedor docker. El usuario podrá conectarse a través de un explorador.

### Diagrama de clases

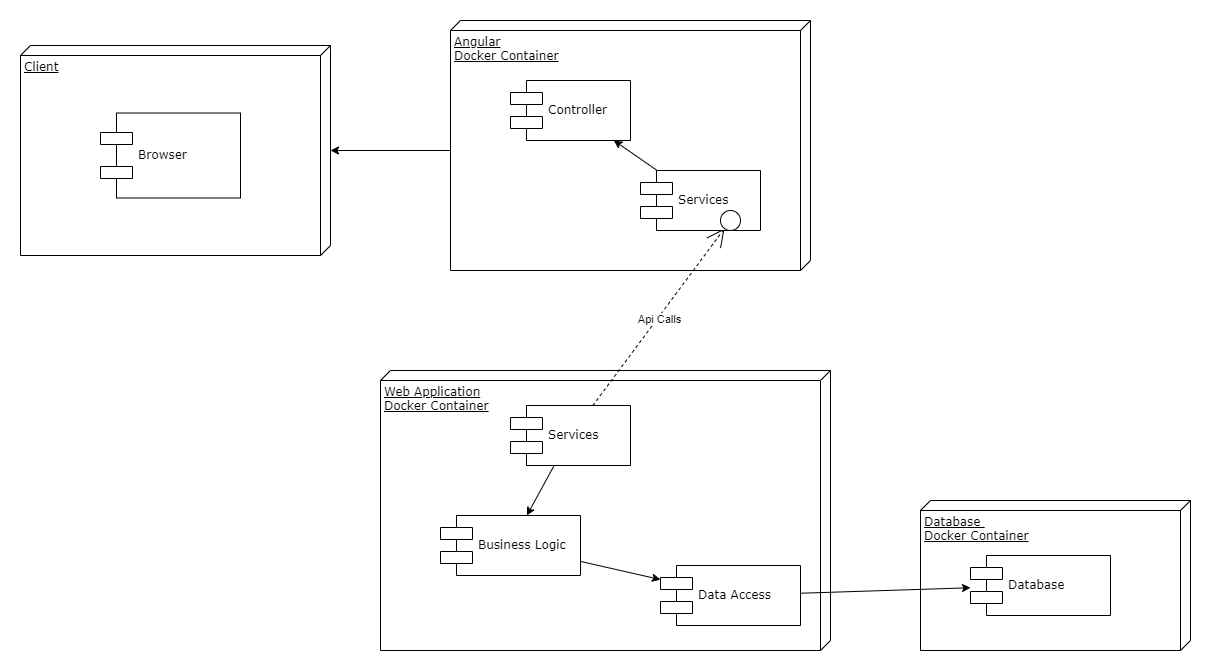
*Diagrama de Clases*

**

## Vista de Componentes

Los servicios estarán dispuestos en contenedores Docker.

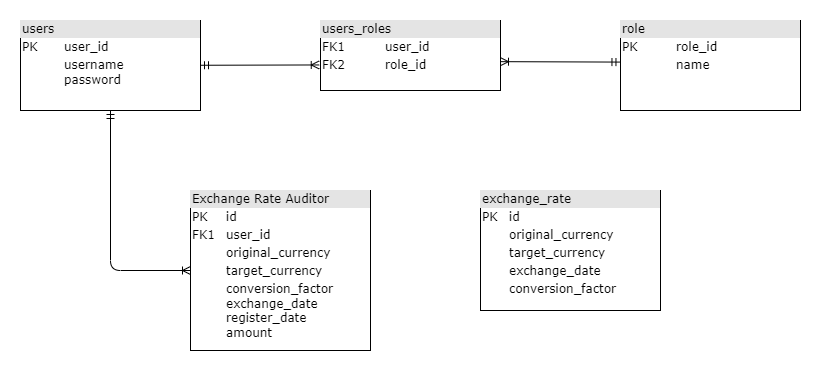
*Diagrama de componentes*

**

## Vista de Datos

Diagrama de base de datos PostgresSQL

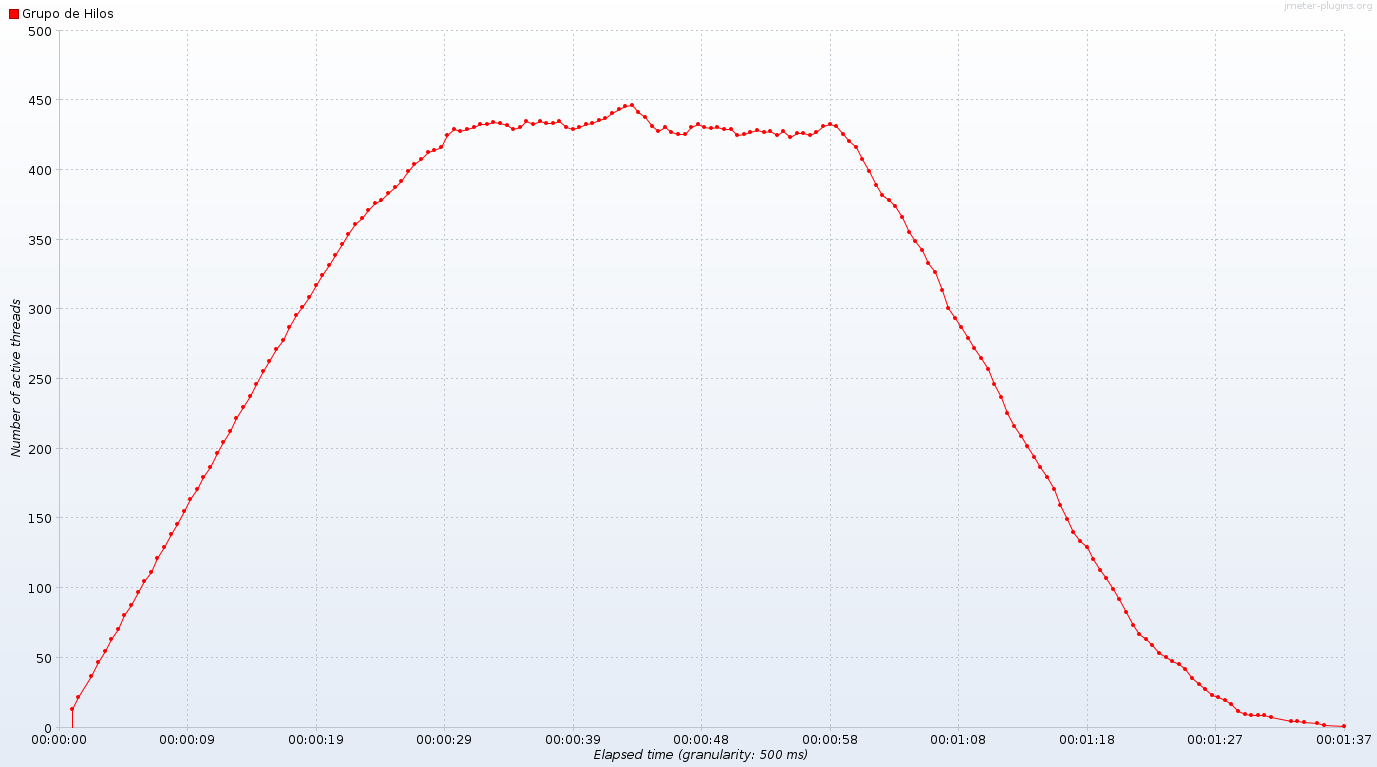
*Diagrama de base de datos*

**

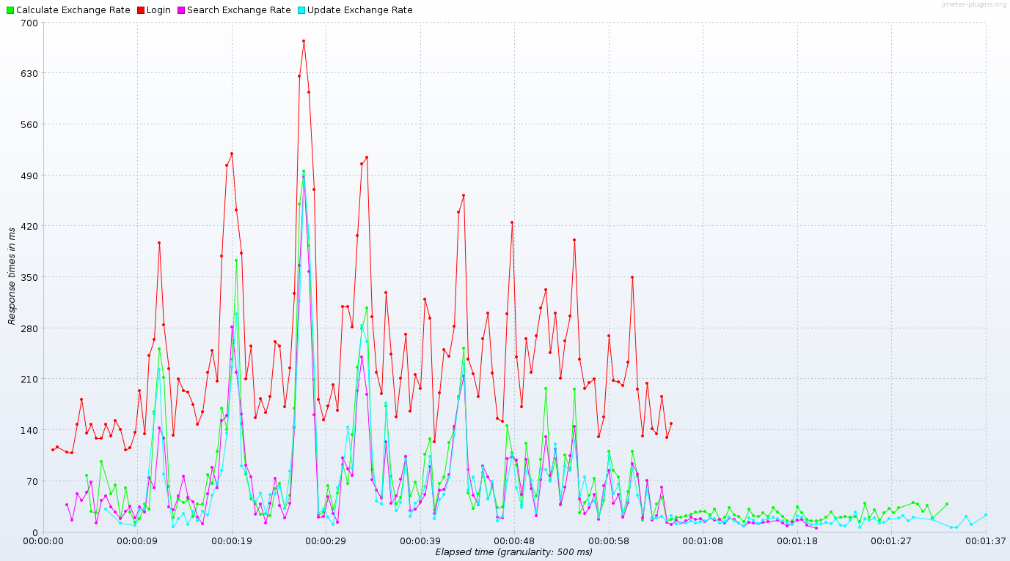
# Calidad

Para comprobar la funcionalidad y el rendimiento del sistema, se realizaron pruebas de rendimiento usando JMeter. Se realiza una simulación de interacción con el sistema.

Las pruebas alcanzaron un aproximado de 450 hilos simultáneos de un total de 10000 hilos interactuando con el sistema.



Se observa que el tiempo de respuesta se encuentra alrededor de los 700 milisegundos entre cada endpoint del sistema.



Se muestra en el siguiente cuadro el tiempo de respuesta. Vemos que el 99% percentil ronda los 500 milisegundos a 700 milisegundos de respuesta.

