

**MIDDLEWARE**

# DESCRIPCIÓN BREVE:

Motor tecnologico en un punto intermedio, el cual apoyara a conectar los servicios de FrontAPP y estos tenerlos expuestos en un socket web.

# Control de Versiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Descripción | Fecha | Encargado |
| 1.0 | Primer lanzamiento | 28/06/2023 | Keny Cornejo |

# Glosario

* Web Socket: es un protocolo de red basado en TCP que establece cómo deben intercambiarse datos entre redes
* Middleware: sistema de software que ofrece funciones y servicios de nube comunes para las aplicaciones, de modo que los desarrolladores y los equipos de operaciones puedan diseñarlas e implementarlas con mayor eficiencia.
* Python: es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código
* Postgresql: es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto
* API: es un intermediario entre dos sistemas, que permite que una aplicación se comunique con otra y pida datos o acciones específicas.
* FrontAPP: es una plataforma colaborativa de comunicación con el cliente diseñada para ayudar a los equipos a administrar sus correos electrónicos, mensajes y otros canales de comunicación en un lugar centralizado.
* Variables de entorno: son cadenas que contienen información acerca del entorno para el sistema y el usuario que ha iniciado sesión en ese momento.
* Thread: es una unidad de ejecución dentro de un proceso. Es una secuencia de instrucciones que el sistema operativo puede programar y ejecutar de forma independiente. Los subprocesos permiten la ejecución simultánea de múltiples tareas dentro de un programa, lo que permite el paralelismo y mejora el rendimiento general.

# Antecedentes

Se maneja una aplicación móvi la cual ya tiene su funcionalidad definida y completada, y se observó que se necesitaba como una herramienta adicional un botón de comunicación para poder charlar con un experto dentro del área, con la finalidad de tener mensajería ágil y personalizada para cada usuario, y a su vez mantener los registros históricos de los mismos. Por las limitaciones y por temas de seguridad de las aplicaciones moviles se recurrio a implementar un middleware para agilizar dichas comunicaciones y centralizar los puntos criticos dentro del flujo de la misma.

# Proyecto

Para la realizacion del Middleware se utilizo el lenguaje de programación Python y sin la utilizacion de frameworks de soporte (como Django) con las siguientes librerias adicionales:

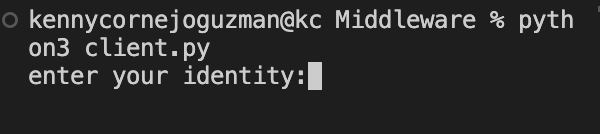
* python-dotenv
* interval-timer
* requests
* psycopg2-binary

A su vez, se utilizo como motor de base de datos Postgresql con la finalidad de mantener integra la información y tener registros de si existio una falla de comunicación tanto con la aplicación movil como con el entorno brindado por FrontAPP. En dicha aplicación se realizo un archivo Docker con el objetivo de mantener en contenedores los servicios creados y poder agilizar su montaje en un ambiente con una previa instalación de Docker. A su vez dentro del proyecto se configuraron las siguientes variables de entorno:

* HOST=Host donde sera montada la aplicación
* PORT=Puerto donde sera montada la aplicación
* HOST\_DB=Host donde se encontrara la base de datos
* USER\_DB=Usuario de autenticación de la base de datos
* PW\_DB=Clave de autenticación de la base de datos
* PORT\_DB=Puerto de conexión hacia la base de datos
* NAME\_DB=Nombre de la base de datos que se desea comunicarse
* BEARER=Token dado por Fronatpp para la comunicación
* CHANNEL=Canal de comunicación para Frontapp.

# Comunicaciones e interacción

Para poderse conectar con la aplicación, se necesita un web socket de tipo cliente, definir el puerto y el host (estos definidos en las variables de entorno), y lo primero que pide el Middleware es que se ingrese el identificador del usuario:



Esto con la finalidad de poder disminuir contenido durante la trama, ya que se hara una comunicación desde cada aplicación hacia el Middleware. Luego en el mensaje 2 hacia N, estos son de comunicación directa con el Frontapp.

