



Proyecto Semestral: Entrega 1

Grupo 27

1. Backend

a. Instalación de dependencias

Para correr el proyecto en un ambiente local, es recomendable utilizar un entorno virtual:

- Asegúrate de tener Python 3.9 o superior instalado.
- Crear un entorno virtual con el siguiente comando:

```
python -m venv venv
```

- Activar el entorno virtual:

- En Windows, ejecuta:

```
venv\Scripts\activate
```

- En Linux/macOS, ejecuta:

```
source venv/bin/activate
```

se deben instalar las siguientes dependencias:

- Python 3.9 o superior
- MongoDB (servidor local o instancia remota)
- Las siguientes librerías de Python:
 - fastapi
 - uvicorn
 - pymongo
 - python-dotenv
 - paho-mqtt

Para instalar las dependencias, ejecutar:

```
pip install fastapi uvicorn pymongo python-dotenv paho-mqtt
```

b. Archivo .env

Crear un archivo llamado `.env` en la carpeta raíz del proyecto se debe tener el siguiente contenido:

```
MONGO_URI =  
MQTT_BROKER=  
MQTT_PORT=  
MQTT_USER=  
MQTT_PASSWORD=  
AUTHO_DOMAIN=  
AUTHO_AUDIENCE=  
URL_FRONTEND=
```

Las variables serán enviadas al ayudante corrector

c. Correr la aplicación

Para levantar el servidor localmente con Docker Compose, en la raíz del proyecto, donde se encuentra el archivo `docker-compose.yml`, ejecutar el siguiente comando:

```
docker-compose up --build
```

Con este comando se construirá y levantará los contenedores necesarios para el backend, es decir la API, mongodb y el cliente mqtt.

```
http://127.0.0.1:8000  
http://localhost:8000/
```

2. Frontend

a. Instalación de dependencias

Para correr el proyecto en un ambiente local, primero se debe ejecutar el siguiente comando desde la terminal de la carpeta raíz del proyecto:

```
cd arquisisvitefrontend
```

Para instalar las dependencias, ejecutar:

```
npm install
```

b. Archivo .env

Crear un archivo llamado `.env` en la carpeta:

```
arquisisvitefrontend
```

Se debe tener el siguiente contenido:

```
PORT=5173  
VITE_AUTHO_DOMAIN=  
VITE_AUTHO_AUDIENCE=  
VITE_AUTHO_CLIENT_ID=  
VITE_BASE_URL=
```

Las variables serán enviadas al ayudante corrector

c. Correr la aplicación

se puede correr el servidor con el siguiente comando en la terminal:

```
npm run dev
```

El servidor se abrirá en el navegador en la siguiente URL:

```
http://localhost:5173
```