

# MEMORIA PARCIAL 2

## Diseño de la Base de Datos

**Estructura:** El diseño de la base de datos utiliza **MongoDB**, una base de datos NoSQL orientada a documentos. La base de datos se llama TextMe y contiene una colección principal:

**mensajes:**

- Representa los mensajes intercambiados entre usuarios.
- **Atributos:**
  - **\_id:** (*ObjectId*) Identificador único del mensaje, asignado automáticamente.
  - **de:** (*String*) Correo electrónico del remitente.
  - **para:** (*String*) Correo electrónico del destinatario.
  - **asunto:** (*String*) Asunto del mensaje, con un límite de 80 caracteres.
  - **timestamp:** (*Integer*) Marca de tiempo del mensaje en formato Unix.
  - **contenido:** (*String*) Texto del mensaje, sin restricciones de tamaño.
  - **adjunto:** (*String, opcional*) URL de una imagen o archivo adjunto.

## **URI de la Base de Datos**

El proyecto utiliza **MongoDB Atlas** para alojar la base de datos en la nube. La conexión se realiza a través del URI definido en el archivo .env:

```
MONGO_URI=mongodb+srv://yaelmartin:yaelmartin@cluster0.ziao0.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0
```

---

## Tecnologías Utilizadas

**Lenguaje de Programación:** Python, utilizado para implementar la API REST.

### **Frameworks y Librerías**

1. **FastAPI:**
  - Framework para construir la API REST.
  - Ofrece documentación interactiva automática con OpenAPI en /docs.
2. **Uvicorn:**

- Servidor ASGI para ejecutar la aplicación FastAPI.

### 3. **PyMongo:**

- Cliente para interactuar con la base de datos MongoDB.

### 4. **Python-Dotenv:**

- Manejo de variables de entorno almacenadas en un archivo .env.

## **Entorno Virtual**

Para garantizar la portabilidad y el aislamiento de dependencias, se configuró un entorno virtual:

```
python -m venv venv
```

## **Instalación de Dependencias**

Dentro del entorno virtual, se instalaron las siguientes dependencias:

```
pip install fastapi uvicorn pymongo python-dotenv email-validator
```

```
python -m pip install "pymongo[srv]"
```

Todas están recogidas en un requirements.txt que se ejecuta de la siguiente forma:

```
pip install -r requirements.txt
```

---

## **Instrucciones de Instalación y Despliegue en local**

### 1. **Clonar el repositorio o descomprimir el zip:**

```
git clone https://github.com/yaELITAMB/FEBRERO2Y3-TEXTME-2021.git
```

```
cd FEBRERO2Y3-TEXTME-2021.git
```

```
cd backend
```

### 2. **Configurar el entorno virtual y activarlo:**

**En Mac:**

```
python3 -m venv venv
```

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

### **En Windows:**

```
python -m venv venv
```

```
.\venv\Scripts\Activate (para PowerShell)
```

```
venv\Scripts\activate (para Command Prompt)
```

(para salir del entorno virtual en ambos casos: deactivate)

### **3. Instalar dependencias:**

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

```
python -m pip install "pymongo[srv]"
```

ó

```
pip install -r requirements.txt
```

### **4. Ejecutar el servidor:**

```
uvicorn main:app --reload
```

### **5. Acceder a la API:**

- Documentación interactiva: <http://127.0.0.1:8000/docs> o <http://localhost:8000/docs> .

## **Despliegue en cloud:**

<https://febrero2y3-textme-2021-production.up.railway.app/>

---

## **Limitaciones y Problemas Encontrados**

- 1. Endpoint de mensajes pending**
- 2. Puerto de despliegue**