

Práctica Aplicación de 12 Factores

La presente práctica busca que cada estudiante ponga en práctica varios de los conceptos de Aplicación de 12 factores o *12 Factors App*. Se tiene un proyecto escrito en Flask el cual es una simple aplicación de listado de inventario de vehículos. La misma utiliza un servidor Postgres para almacenar los datos.

El proyecto, aunque funcional es un proyecto poco maduro; por ello se solicita a cada estudiante el que implemente los siguientes principios.

1. Repositorio (*code base*).
2. Dependencias.
3. Configuración.
4. Servicios de Apoyo.
5. Exposición de Puertos.
6. Concurrencia.
7. Procesos Administrativos.

En cuanto a algunos puntos: **4. Servicios de Apoyo**) Se espera que se considere Postgres como un servicio de apoyo, pues ha de correr contra cualquier servidor Postgres sin fricción alguna. **6. Concurrencia**) Flask puede ejecutar con `flask run`, sin embargo no es lo más óptimo, por lo que se espera que se utilice un servidor WSGI que maneje múltiples solicitudes a la vez. **7. Procesos Administrativos**) Ahora mismo no se tienen datos en la app, por lo que se requiere que provean un comando, ya sea utilizando `click` o algún otro método para cargar datos iniciales.

Está práctica tiene un valor de 5 puntos, que equivalen al porcentaje de completitud de la práctica misma.

Observaciones

- Las dependencias a instalar son las siguientes. Nótese que SQLAlchemy es la `1.4`, versiones más recientes no son compatibles. Pueden utilizar `pip`, `pip-tools`, o cualquier otra herramienta para manejar las dependencias.

```
Flask
Flask-Migrate
Flask-SQLAlchemy
SQLAlchemy==1.4
psycopg2-binary
```

- Para iniciar una instancia de Postgres utilizando Docker pegar en la consola:

```
docker run --name cars_api -p "5432:5432" --memory="128mb" -e  
POSTGRES_PASSWORD=postgres -e POSTGRES_DB=cars_api -d postgres:14
```

- Para iniciar por primera vez la app han de *migrar la base de datos* por lo que han de ejecutar lo siguiente

```
# En Windows  
set FLASK_APP=app.py  
# En Linux: export FLASK_APP=app.py  
  
# Migrar la BBDD  
flask db migrate  
flask db upgrade
```

- Para ejecutar la app

```
flask run
```

- Pueden utilizar la colección, y el ambiente de postman dentro del proyecto para probar la API en cuestión.