

UNIVERSIDAD CATÓLICA ANDRÉS BELLO  
FACULTAD DE INGENIERÍA  
ESCUELA DE INGENIERÍA INFORMÁTICA  
CÁTEDRA DE SISTEMAS OPERATIVOS  
PROFESOR RÓMULO RODRÍGUEZ  
ALUMNOS RAYNOLD GUERRERO Y JESÚS DÍAZ  
DOCUMENTACIÓN DE PRÁCTICA 8: SERVICIOS WEB

- Requerimientos.
  - NodeJS: versión 20.2.0
  - NPM: versión 9.6.7
  - NestJS: versión 9.5.0
  - PostgreSQL: versión 15.3
- Proceso de compilación de ser requerido.
  - Una vez clonado el proyecto de github, se accede a la carpeta del proyecto, se ejecuta el comando por consola `npm install` para instalar todas las dependencias del proyecto.
  - Para ejecutar la aplicación en modo desarrollo watch se ejecuta el comando desde la terminal: `npm run start:dev`
  - Para transpilar la aplicación de typescript a javascript se utiliza el comando: `npm nest build`
  - Para ejecutar la aplicación en producción se usa el comando: `npm run start:prod`
- Configuración.
  - Clonar proyecto de github:
    - Link HTTP:  
<https://github.com/jesusdicenucab/distribuidos-practica8.git>
    - Link ssh: [git@github.com:jesusdicenucab/distribuidos-practica8.git](https://github.com/jesusdicenucab/distribuidos-practica8.git)
  - Leer archivos README.md y README2.md
  - Crear archivo `.env` en la carpeta raíz del proyecto.
  - Definir los valores que se encuentran en el archivo `.env.template` en el archivo `.env` creado.

```
■ DATABASE_HOST= Host donde se encuentra alojada la base de
datos. Suele ser localhost si es en local.
■ DATABASE_PORT= Puerto donde escucha la base de datos. El puerto
predeterminado para PostgreSQL suele ser el 5432.
■ DATABASE_USERNAME= Usuario que tiene acceso a la base de
datos
■ DATABASE_PASSWORD= Contraseña del usuario que tiene acceso a
la base de datos
■ DATABASE_NAME= NOMBRE DE LA BASE DE DATOS
```

■ SERVER\_PORT= PUERTO DONDE SE EJECUTARÁ EL SERVIDOR

Ejemplo:

```
DATABASE_HOST=localhost
DATABASE_PORT=5432
DATABASE_USERNAME=postgres
DATABASE_PASSWORD=postgres
DATABASE_NAME=practica8
SERVER_PORT=3850
```

- Despliegue de estructuras para persistencia de datos.  
Para la persistencia de datos se utilizó el ORM TypeORM. Se definieron las entidades de la aplicación utilizando este ORM.
- Ejecución del servicio.
  - Para ejecutar la aplicación en modo desarrollo watch se ejecuta el comando desde la terminal: `npm run start:dev`
  - Para transpilar la aplicación de typescript a javascript se utiliza el comando: `npm run build`
  - Para ejecutar la aplicación en producción se usa el comando: `npm run start:prod`
  - El url de la aplicación es `localhost:SERVER_PORT:api/v1`
  - El url para acceder a la interfaz gráfica de swagger es: `localhost:SERVER_PORT:swagger`
    - Ejemplo: si se definió en el archivo `.env` la variable de entorno `SERVER_PORT` con el valor 3850, entonces la url para acceder a la interfaz gráfica de swagger es: `localhost:3850/swagger`.
    - Para acceder a la interfaz gráfica de swagger, la aplicación debe estar ejecutándose en modo desarrollo watch. Para ejecutar la aplicación de esta forma se utiliza el comando por consola: `npm run start:dev`