

Ejecución en ambiente de desarrollo local

Prerequisitos

- Tener un gestor de base de datos
 - Clonar el repositorio del proyecto
 - Instalar las dependencias de node con el comando “npm i” en cada carpeta (frontend, backend)
 - Tener una base de datos “test” ya creada vacia.
1. Crear archivo de variables de entorno. env para el frontend y el backend
 - a. Variables frontend

```
📁 .env Frontend
1
2 VITE_HOST=0.0.0.0
3 VITE_PORT=3000
4 VITE_BACKEND_URL="http://localhost:4000"
5
```

VITE_HOST=0.0.0.0 significa que el servidor Vite estará escuchando en todas las interfaces de red de la máquina. VITE_PORT=3000 significa que el servidor Vite estará escuchando en el puerto 3000.

La variable VITE_BACKEND_URL se utiliza para especificar la URL del servidor backend. En este caso, el servidor backend está corriendo en <http://localhost:4000>.

- b. Variables Backend

```
📁 .env Backend
1 DB_NAME="test"
2
3 # DB
4 DB_USER="root"
5 DB_PASSWORD=""
6 DB_HOST="localhost"
7 DB_PORT=3306
8 DB_DRIVER="mariadb"
9 JWT_SECRET=""
10
11 # configuración cors
12 ORIGIN="http://localhost:3000"
13
14
15 # configuración server
16 IP="0.0.0.0"
17 PORT=4000
```

Las primeras seis variables (DB_NAME, DB_USER, DB_PASSWORD, DB_HOST, DB_PORT, DB_DRIVER) están relacionadas con la configuración de la base de datos. Estas variables se utilizan para definir el nombre de la base de datos (DB_NAME), el usuario (DB_USER), la contraseña (DB_PASSWORD), el host (DB_HOST), el puerto (DB_PORT) y el controlador de la base de datos (DB_DRIVER) que tu aplicación utilizará para conectarse a la base de datos.

La variable JWT_SECRET es utilizada para firmar y verificar los tokens JWT en tu aplicación. JWT significa "JSON Web Token", que es un estándar de la industria para la creación de tokens de acceso que permiten la autenticación de usuarios y la transmisión segura de información. (Esta variable no debe estar vacía)

La variable ORIGIN se utiliza para configurar el CORS (Cross-Origin Resource Sharing) en tu aplicación. CORS es una política de seguridad que permite o restringe las solicitudes de recursos a tu servidor desde un origen diferente.

Finalmente, las variables IP y PORT se utilizan para configurar la dirección IP y el puerto en el que tu servidor estará escuchando las solicitudes entrantes.

2. Ingresar a las carpetas frontend y backend en una terminal separada para cada una y ejecutar "npm run dev" para correr el proyecto. A continuación, se muestra la ejecución de ejemplo.

```
PS C:\Users\Tuki\Desktop\10 semestre\IniciativaSer\Frontend> npm run dev

> plantilla@0.0.0 dev C:\Users\Tuki\Desktop\10 semestre\IniciativaSer\Frontend
> vite

VITE_APP_HOST: 0.0.0.0
VITE_APP_PORT: 3000

VITE v4.5.2 ready in 683 ms

  → Local:   http://localhost:3000/
  → Network: http://172.18.0.1:3000/
  → Network: http://192.168.4.38:3000/
  → press h to show help
Browserslist: caniuse-lite is outdated. Please run:
  npx update-browserslist-db@latest
  Why you should do it regularly: https://github.com/browserslist/update-db#readme
```

```
PS C:\Users\Tuki\Desktop\10 semestre\IniciativaSer\Backend> npm run dev

> nodejs-postgresql-restapi@1.0.0 dev C:\Users\Tuki\Desktop\10 semestre\IniciativaSer\Backend
> nodemon src/index.js

[nodemon] 3.1.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node src/index.js`
Executing (default): SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE' AND TABLE_NAME = 'ambitodominioarea' AND TABLE_SCHEMA = 'test'
Executing (default): CREATE TABLE IF NOT EXISTS `ambitodominioarea` (`id` INTEGER auto_increment , `nombre` VARCHAR(255), `createdAt` DATETIME NOT NULL, `updatedAt` DATETIME NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB;
Executing (default): SHOW INDEX FROM `ambitodominioarea`
Executing (default): SELECT `id`, `nombre`, `createdAt`, `updatedAt` FROM `ambitodominioarea` AS `ambitodominioarea` WHERE `ambitodominioarea`.`nombre` = 'Arquitectura' LIMIT 1;
datos de ambito creados automáticamente
Executing (default): SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE' AND TABLE_NAME = 'comuna' AND TABLE_SCHEMA = 'test'
Executing (default): INSERT INTO `ambitodominioarea` (`id`,`nombre`,`createdAt`,`updatedAt`) VALUES (DEFAULT,?,?,?);
Executing (default): INSERT INTO `ambitodominioarea` (`id`,`nombre`,`createdAt`,`updatedAt`) VALUES (DEFAULT,?,?,?);
```

3. El proyecto se ejecutará en la ruta local definida `http://localhost:3000/`

