

Manual de Instalación

La aplicación se ha desarrollado pensando en un entorno de funcionamiento en Windows, bajo dos servidores, uno en NodeJS y otro en NodeJS y Vuetify. Por ello, se incluye el manual de instalación con los pasos básicos de instalación, como instalación de paquetes y configuraciones previas.

Requisitos de los servidores

Para el backend, es necesario tener los siguientes programas instalados:

- NodeJS (v16.10.0 o mayores)
- MySQL Community Server 8.0, destacar de aquí, que se guarde el usuario y contraseña del perfil de la base de datos, ya que se usará más tarde.

Para el frontend, solo es necesario tener Node.js instalado.

Instalación del backend en el equipo

Para el backend, es necesario obtener la URL del repositorio GitHub que aloja el backend. Para ello se recomienda realizar los siguientes comandos por terminal en el directorio que se desee instalar el backend:

```
mkdir backend  
  
cd backend  
  
git clone https://github.com/apereg21/TrainingToLearn.git
```

Tras estos pasos, el backend estará instalado, pero queda por instalar los paquetes necesarios y la configuración de archivos, para su correcto funcionamiento.

Instalación de paquetes y configuración del backend

Para que el backend funcione correctamente, se necesita de la instalación de paquetes y la configuración de archivos.

En cuanto a la instalación de paquetes, una vez instalado Node.js en el equipo, se procede a la introducción del siguiente comando por terminal, en el directorio en el que se instaló el backend:

```
cd TrainingToLearn/TraningToLearn  
npm install sequelize mysql2 express crypto-js elliptic cors --save
```

Una vez se hayan instalado los paquetes, toca configurar un archivo para el correcto funcionamiento del backend. Siguiendo los pasos de la guía de instalación, el archivo a configurar se encuentra en la siguiente ruta: “backend/TrainingToLearn/TrainingToLearn/config”.

El archivo es llamado config.json. Este archivo, es el encargado de preparar el funcionamiento del ORM Sequelize. Este paquete permite el realizar peticiones con la base de datos. En el archivo, se puede observar la siguiente los siguientes campos:

- **username** → En esta variable se pone el nombre del usuario de MySQL Community Server en este caso. Normalmente, si no se tocó nada en durante la configuración de MySQL Server, está predefinido como root.
- **password** → En esta variable se pone la contraseña del usuario de MySQL Community Server en este caso.
- **database** → En esta variable se pone el nombre de la base de datos creada en MySQL Community Server en este caso. Aquí el nombre esta predefinido en “database_blockchain”, pero puede ser cambiado.
- **host** → En esta variable se pone el nombre del usuario de MySQL Community Server en este caso. Normalmente, si no se tocó nada en durante la configuración de MySQL Server, está predefinido en la dirección 127.0.0.0.

- **dialect** → En esta variable, es el lugar donde se pone el dialecto que se utiliza. En este caso, es mysql, pero sequelize permite el uso de más dialectos, si se desea (MongoDB, PostgreSQL, MariaDB, ...).

Tras rellenar todos los campos convenientes, no queda más por configurar en el backend.

Instalación del frontend en el equipo

Para el frontend es necesario obtener la URL del repositorio GitHub que aloja el frontend. Para ello se recomienda realizar los siguientes comandos por terminal en el directorio que se desee instalar el frontend:

```
mkdir frontend  
  
cd frontend  
  
git clone https://github.com/apereg21/trainingtolearn-app.git
```

Tras estos pasos, el frontend estará instalado, pero queda por instalar los paquetes necesarios y la configuración de archivos, para su correcto funcionamiento.

Instalación de paquetes y configuración del frontend

Para que el frontend funcione correctamente, se necesita de la instalación de paquetes. Una vez instalado Node.js en el equipo, se procede a la introducción del siguiente comando por terminal, en el directorio en el que se instaló el frontend:

```
cd trainingtolearn-app  
  
npm install -g @vue/cli  
  
npm install axios
```

Tras instalar los paquetes, no queda más por configurar en el frontend y en el backend.

Puesta en Marcha

Para poder probar el correcto funcionamiento de la parte frontend y de la parte backend, se procede a explicar cómo se inicia tanto el servidor backend como el servidor frontend.

Para poder arrancar el servidor frontend. El usuario tiene que acceder a la carpeta donde instaló el contenido del repositorio de Github en el interior de la carpeta trainingtolearn-app e introducir el siguiente comando mediante una consola/terminal:

```
npm run serve
```

Pasado un tiempo, si todo ha ido bien, la consola devolverá un mensaje notificando donde se puede acceder al frontend desde el navegador con la dirección <http://localhost:8080/> o <http://localhost:8081/>.

Para poder arrancar el servidor backend. El usuario, en un primer momento, tiene que haber creado al menos la base de datos en MySQL Server. Si no lo ha hecho, se deberá crear, recordando que el nombre de esta tiene que ser `database_blockchain` si no ha decidido cambiar este parámetro en la configuración de `config.json`. El usuario tras esto tiene que acceder a la carpeta donde instaló el contenido del repositorio de Github en el interior de la carpeta `TrainingToLearn/TrainingToLearn` e introducir el siguiente comando mediante una consola/terminal:

```
node server.js
```

Pasado un tiempo, si todo ha ido bien, la consola devolverá un mensaje notificando que se está escuchando peticiones por el puerto 3000. Tras esto, el usuario tiene que acceder a la siguiente dirección, desde un navegador: <http://localhost:3000/maintenance/restoreDB>, para que sequelize pueda rellenar la base de datos con las tablas automáticamente y poder trabajar de manera normal.