



Manual de Instalación

BeCation

Gonzalo Gimenez

Dylan Seltzer

Diego Sanchez

Nicolas Nieto

Emilio Pereira

Ezequiel Baqueiro

Javier Rojas

Luis Meza



INDICE

CLONAR EL PROYECTO	3
Repositorio	3
Clonar Repositorio	3
Elección de rama	3
RAMAS	4
Documentation	4
Backend	4
Frontend	4
ABRIR EL PROYECTO	5
Directorio	5
Visual Studio Code	5
INICIALIZACIÓN DEL SERVIDOR	6
NPM	6
.ENV	6
Base de Datos	7
Abrir el servidor	8
Swagger UI	8
INICIALIZACIÓN DE LA APLICACIÓN	10
NPM	10
.ENV	10
Abrir la aplicación	10
Comenzar con su uso	10

CLONAR EL PROYECTO

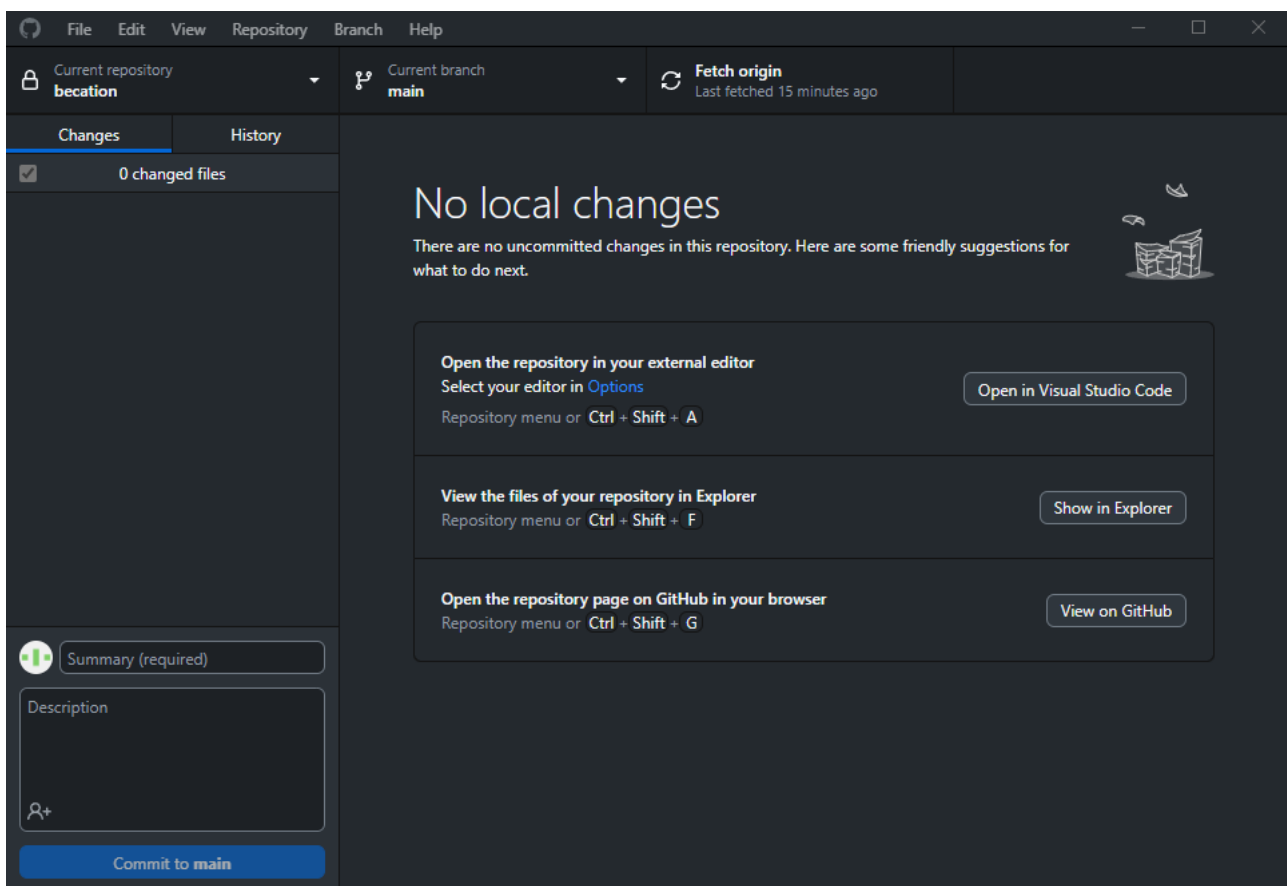
Repositorio

Para comenzar, deberá entrar al repositorio de BeCation en GitHub y copiar la URL del proyecto para luego clonarla.

<https://github.com/DylanseET32/becation.git>

Clonar repositorio

Una vez haya copiado la URL, deberá abrir GitHub Desktop y seleccionar la opción de “Clonar repositorio”, o en inglés “Clone repository”. Ahí pegará la URL que copió anteriormente y se instalará el directorio de “becation”.



Elección de rama

Al haber clonado el proyecto, podrá visualizar 3 ramas diferentes además de la rama principal “main”. Estas son: Documentation, Backend y Frontend. Deberá elegir una rama para poder inicializarla.

RAMAS

Documentation

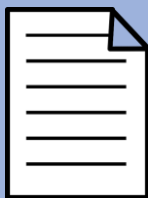
En la rama Documentation podrá ver toda la documentación hecha. Encontrará el Ante Proyecto, DER, DCU, GANTT, Manual de Instalación, Manual de Recursos Humanos y Manual de Usuario.

Backend

En la rama Backend encontrará todo el procesamiento y la gestión de datos de la aplicación. Aquí tendrá la API, la base de datos y la documentación de los endpoints realizado con Swagger. Desde aquí podrá incializar el servidor de la aplicación.

Frontend

La rama Frontend es donde podrá ver la interfaz del usuario, con su diseño y estilos, que está conectada al Backend a través de la API para que la funcionalidad de esta sea correcta. Desde esta rama podrá inicializar la aplicación de BeCation.



ABRIR EL PROYECTO

Directorio

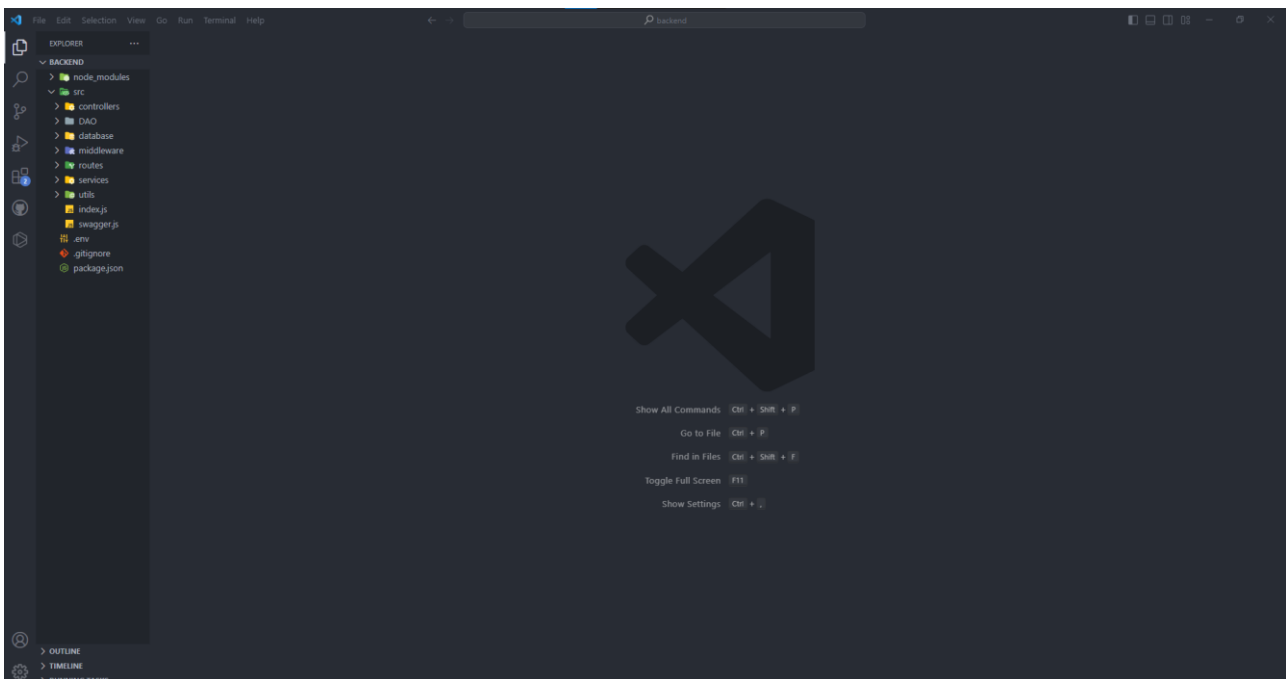
Después de haber clonado el repositorio, seguirá la siguiente ruta en su explorador de archivos: “C:\Users\(\su usuario)\Documents\GitHub”

Allí se encontrará con el directorio del proyecto.

Visual Studio Code

Para el siguiente pasó deberá tener instalado el Visual Studio Code.

Seleccionará la rama Backend para poder inicializar el servidor de la aplicación y a continuación insertará la carpeta “backend”, que se encuentra en el directorio de “becation”, a Visual Studio Code.



INICIALIZACIÓN DEL SERVIDOR

NPM

Para comenzar con la inicialización del servidor, primero deberá instalar el sistema de gestión de paquetes para Node.js ingresando en la consola `npm install`. Esto instalará una carpeta llamada `node_modules`.

.ENV

Verá que dentro de backend tendrá un archivo llamado `.env`, el cual tendrá diferentes variables de entorno las cuales tendrán que ser modificadas acorde a donde suban el proyecto.

PORT: El puerto donde va a estar trabajando el backend.

EMAIL_ADDRESS: Aquí se tendrá que poner una dirección de mail.

EMAIL_PASS: En caso de tener un `email_address`, deberá tener una contraseña con privilegios.

DOMAIN_FRONTEND: Aquí va el dominio del frontend que se va a usar.

JWT_SECRET: Aquí deberán generar una contraseña secreta, es decir, una contraseña aleatoria generada por el administrador del proyecto.

JWT_EXPIRATION: Es en cuánto se vencerá la clave generada anteriormente.

JWT_EXPIRATION_EMAIL: Es el tiempo de validez que tendrá un mail.

SQL_HOST: El nombre del host SQL.

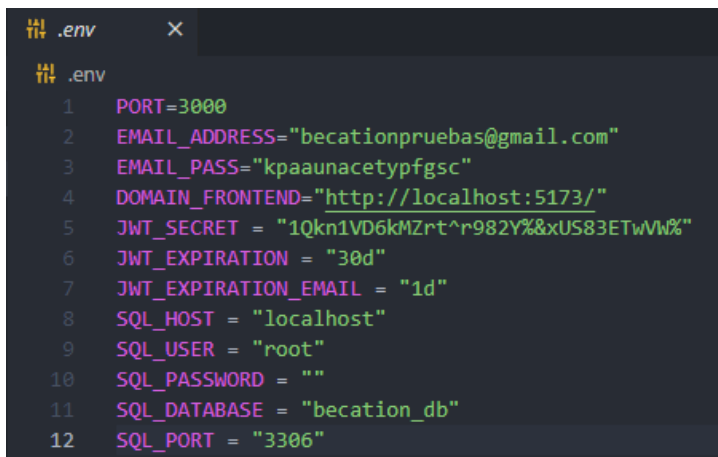
SQL_USER: Nombre del usuario SQL.

SQL_PASSWORD: La contraseña del usuario SQL.

SQL_DATABASE: El nombre de la base de datos.

SQL_PORT: El puerto SQL

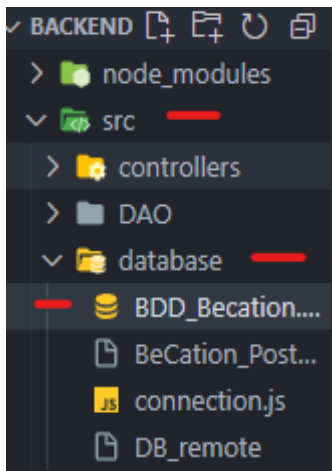
Todos los datos SQL son necesarios para conectar con la base de datos.



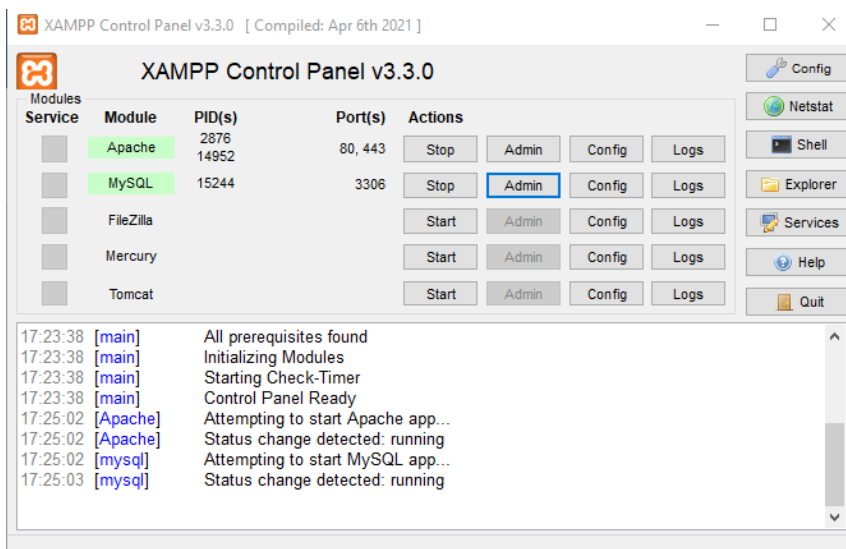
```
.env
1  PORT=3000
2  EMAIL_ADDRESS="becationpruebas@gmail.com"
3  EMAIL_PASS="kpaaunacetyfgsc"
4  DOMAIN_FRONTEND="http://localhost:5173/"
5  JWT_SECRET = "1Qkn1VD6kMZrt^r982Y%&xUS83ETwVW%"
6  JWT_EXPIRATION = "30d"
7  JWT_EXPIRATION_EMAIL = "1d"
8  SQL_HOST = "localhost"
9  SQL_USER = "root"
10 SQL_PASSWORD = ""
11 SQL_DATABASE = "becation_db"
12 SQL_PORT = "3306"
```

Base de datos

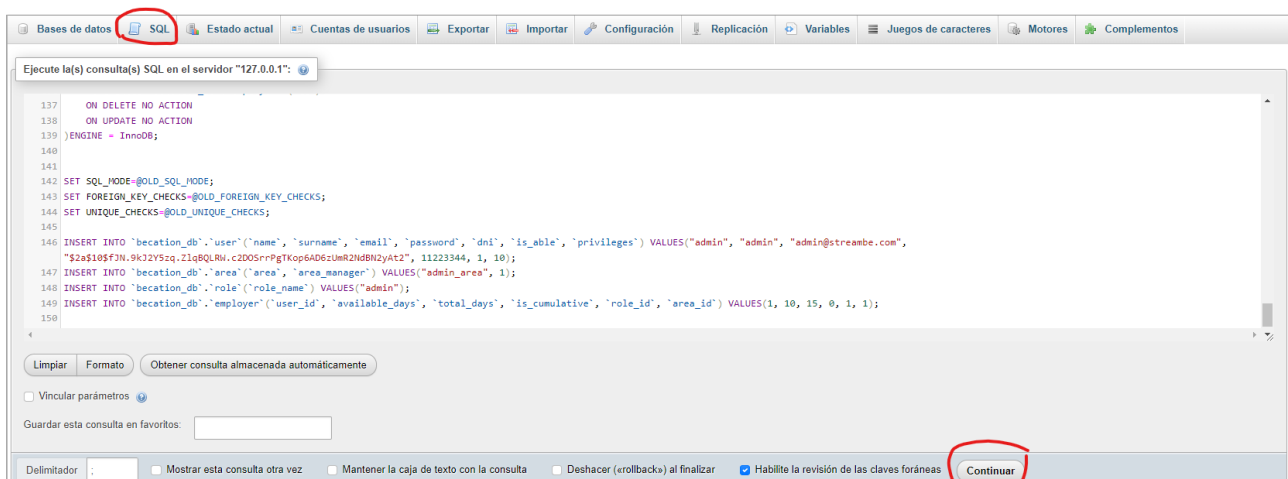
Luego deberá copiar el archivo sql de la Base de Datos que se encuentra en la dirección src/database.



Y para continuar deberá tener el XAMPP instalado para iniciar MySQL y Apache. Una vez iniciados tocará en el botón Admin de MySQL.



Eso lo redirigirá a <http://localhost/phpmyadmin/> donde podrá pegar la base de datos que anteriormente ha copiado en la opción que dice SQL y una vez haberlo insertado, tocará en Continuar.



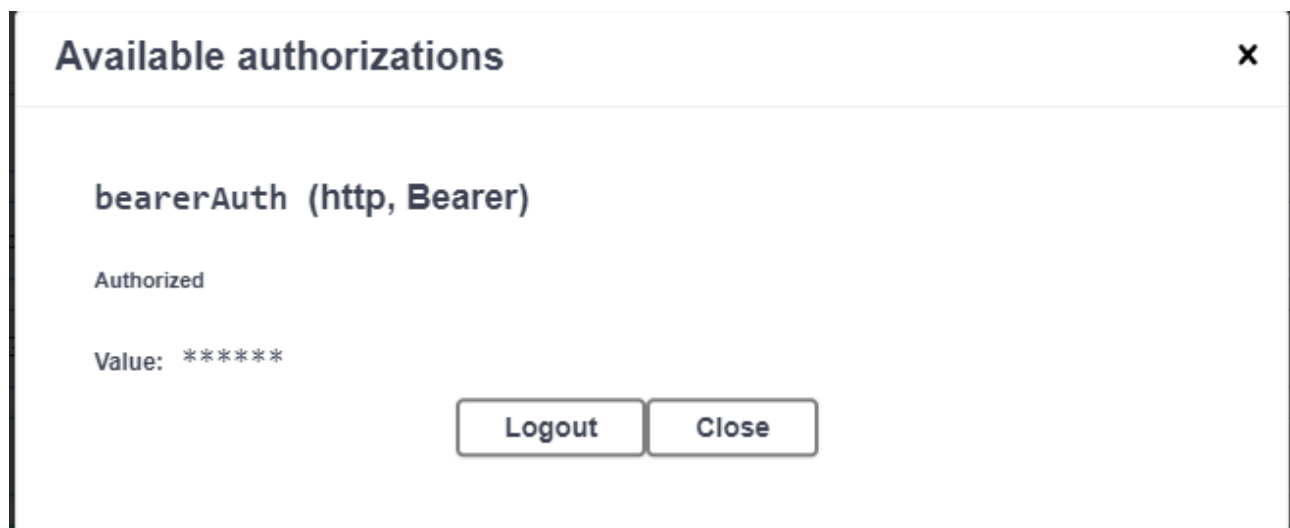
A continuación, ya podrá inicializar el servidor.

Abrir el servidor

Volviendo al Visual Studio Code abrirá la consola y escribirá el siguiente comando `npm run dev`. Aquí se iniciará el servidor y por consola tendrá la dirección del mismo. Esta la pegará en su navegador de preferencia y se le mostrará la documentación de la API.













Swagger UI

Swagger se utilizó para la documentación de cada endpoint. Al haber inicializado el servidor podrá ver y probar cada endpoint desde Swagger UI. Aquí encontrará los GET, PATCH, POST y DELETE, tanto para las áreas de la empresa, los empleados que tendrá la misma, los roles (cada empleado tendrá un rol), las vacaciones que cada empleado podrá pedirse, e incluso desde aquí podrá deshabilitar un usuario o resetear una contraseña. Para poder comenzar con su uso deberá loguearse. Verá una opción que dice Login, donde ejecutará los datos a ingresar. En email pondrá “admin@gmail.com” y en contraseña “admin”. Al ejecutarla se le pasará un token el cual tendrá que pegarlo en el botón de Authorize y será autorizado para probar cada endpoint.













BeCation API 1.0.0 OAS 3.0Authorize 

Area ^

GET	/v1/area/getAllAreas	Get all areas	 
GET	/v1/area/getAreaById/{id}	Get area by ID	 
GET	/v1/area/getAreaByColumn/{name}	Get area by column	 
POST	/v1/area/addArea	Add area	 
PATCH	/v1/area/editArea/{id}	Edit area	 
DELETE	/v1/area/deleteArea/{id}	Delete area	 

Employer ^

GET	/v1/employer/getAllEmployers	Get all employers	 
GET	/v1/employer/getAllEmployersByArea/{id}	Get all employers by area	 
GET	/v1/employer/getEmployerById/{id}	Get all employers by id	 
GET	/v1/employer/getEmployer	Get employer	 
POST	/v1/employer/addEmployer	Add employer	 

Así se mostrarán los endpoints para poder utilizar con Swagger. Cada uno con una descripción de que función tiene, y dentro podrá ver los detalles y tendrá un ejemplo de ayuda.

INICIALIZACIÓN DE LA APLICACIÓN

Para comenzar con la inicialización de la aplicación deberá seleccionar la rama de frontend y abrir su respectiva carpeta en Visual Studio Code.

NPM

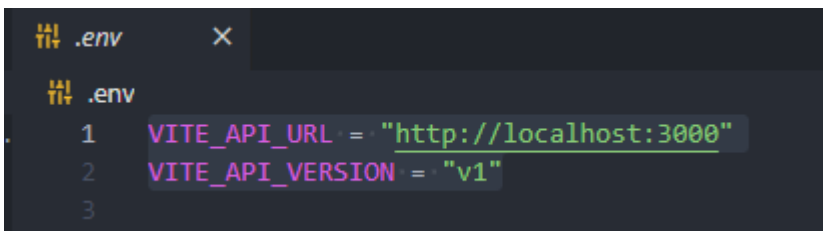
Una vez estado ubicado en la carpeta frontend desde Visual Studio Code, deberá abrir la consola e insertar el comando `npm install`, como hizo al momento de inicializar el servidor en backend

.ENV

Al igual que en el backend, en el frontend encontrará un archivo `.env` el cual servirá para que funcione el frontend.

VITE_API_URL: Aquí ira la dirección donde se abrirá el frontend

VITE_API_VERSION: Aquí ira la versión de la API. En el caso de que en un futuro se modifique la API, la versión será una nueva.



```
.env
1 VITE_API_URL = "http://localhost:3000"
2 VITE_API_VERSION = "v1"
3
```

Abrir la aplicación

Como siguiente paso, en la consola escribirá el comando `npm run dev`. Entrará a la dirección que le pasa por consola para poder abrir la aplicación. La misma podrá abrirla desde su navegador de preferencia.

Comenzar con su uso

Una vez inicializada la aplicación, deberá loguearse con su cuenta. En este caso, la del administrador.

Email: `admin@gmail.com`

Password: `admin`

Listo, ya podrá comenzar con su uso.

¡Gracias por elegirnos! Que disfrute de BeCation.