



Manual Técnico

Página Web
Inventario Forestal Nacional

Autores del manual
Carlos Martín Pinto Grisales
Edinson Ferney Beltrán Gómez

Grupo
5A

Fecha de finalización
17/11/2025

Índice

| | |
|---|----------|
| Introducción | 2 |
| Requisitos y Requerimientos | 2 |
| Descripción general del sistema | 2 |
| Arquitectura y Tecnologías | 2 |
| Estructura del Proyecto | 4 |
| Base de datos | 5 |
| APIs y Endpoints | 6 |
| Mantenimiento y Actualización | 9 |
| Consejos y buenas prácticas de uso | 9 |
| Soporte y contacto | 9 |

Introducción

Requisitos y Requerimientos

Descripción general del sistema

Arquitectura y Tecnologías

Para el Frontend utilizamos tecnologías tales como React, Javascript, en despliegue preferimos desplegarlo en vercel.com.

Para el backend utilizamos varias tecnologías tales como FastAPI, Wordpress, Python, para generar la comunicación entre el Frontend y la base de datos, separando responsabilidades, en total se crearon 3 backend que corresponden a Login, Brigadas y Levantamiento de datos, cada backend ha sido desplegado en un vercel independiente.

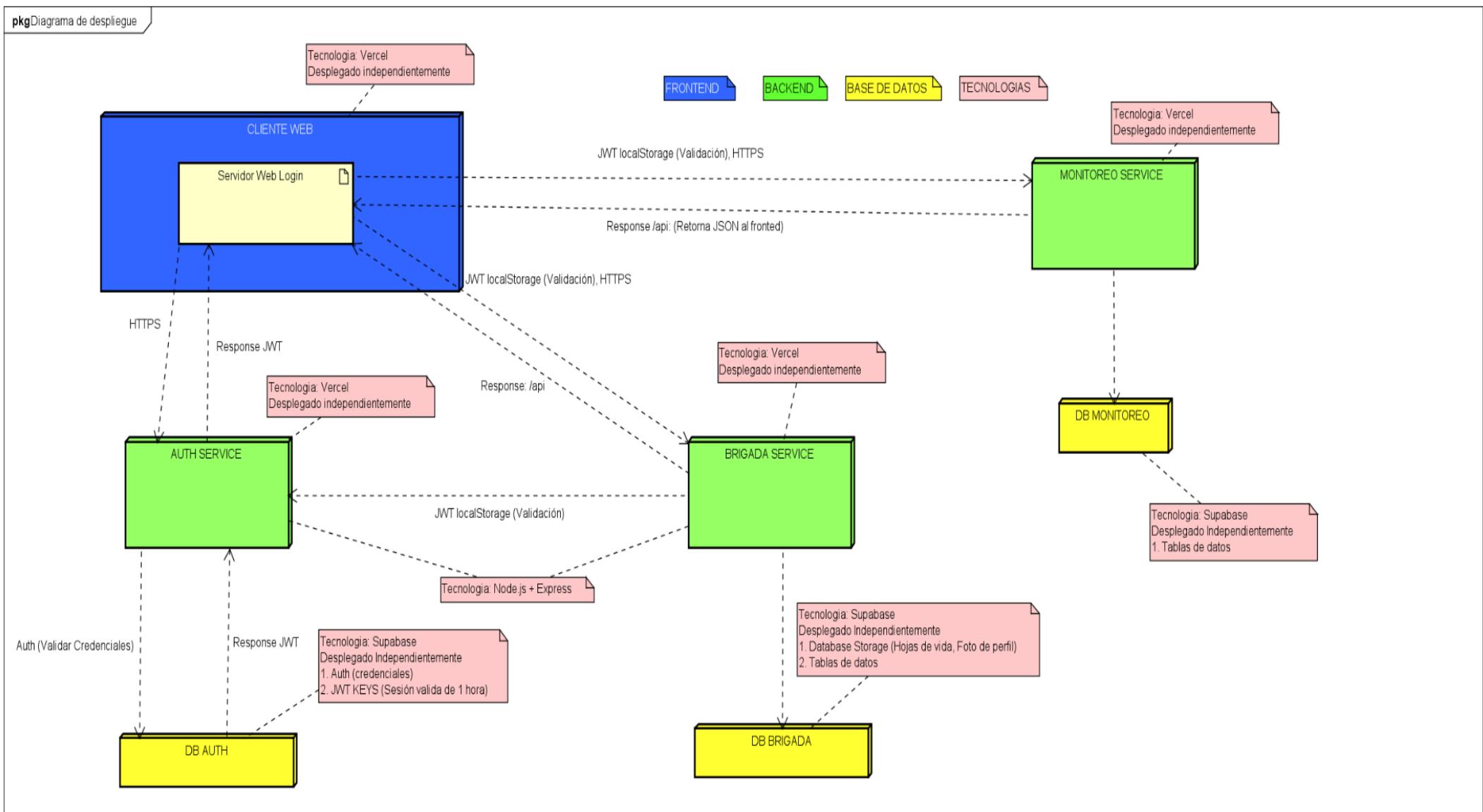
Hemos utilizado tecnologías tales como Supabase para desplegar 3 bases de datos, para Login, Brigadas y Monitoreo (Levantamiento de datos).

Hemos optado por una arquitectura de microservicios, en total 3 microservicios independientes de cada uno, cada uno con un backend independiente desplegado y comunicado correctamente.

Nota:

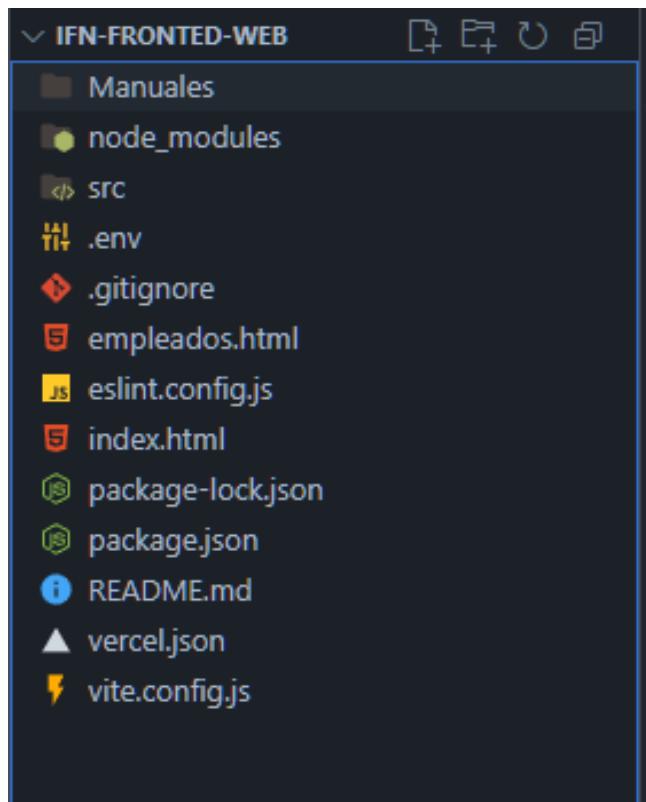
Hemos decidido utilizar el Framework de FastAPI por su documentación automática y respuesta a peticiones ante otros Frameworks.

Diagrama de despliegue



Estructura del Proyecto

Actualmente el Frontend está estructurado de la siguiente manera



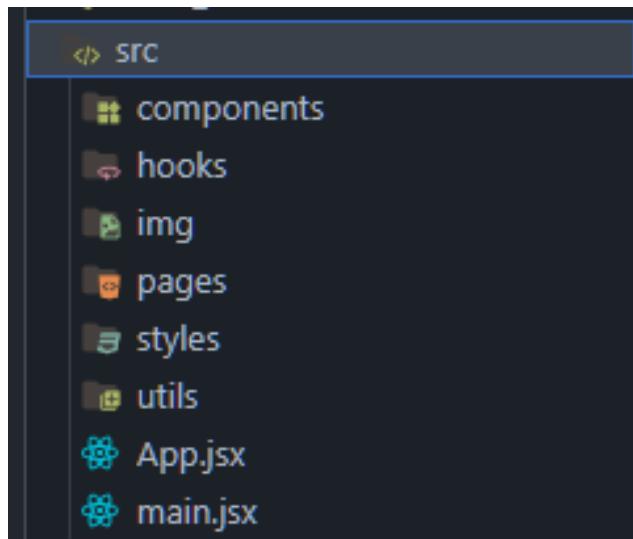
En esta imagen podemos observar una carpeta de “Manuales” donde se guardan todos los manuales (Usuario y Técnico).

Tenemos un archivo .env donde guardamos las variables secretas que requieren seguridad.

En el .gitignore es el archivo donde guardamos los archivos del proyecto que no queremos que se carguen en el GitHub.

En el archivo vercel.json guardamos las configuraciones básicas requeridas por Vercel para ser desplegado.

En el archivo [vite.config.js](#) guardamos la configuración básica requerida por Vite para funcionar en acoplamiento con React.



En esta imagen podemos observar el contenido de la carpeta src donde tenemos nuestro proyecto, en el cual encontramos otra carpeta de assets, donde principalmente es una carpeta creada por React automáticamente donde se comparten recursos tales como imágenes.

En la carpeta de **componentes** se guardan las páginas que se utilizan en la parte del Frontend.

En la carpeta **hooks** guardamos el controlador de la sesión el cual verifica la autenticidad del token del usuario logueado.

En la carpeta **img** guardamos las imágenes que vamos a llamar durante el transcurso de la ejecución del sistema.

En la carpeta de **pages**, es donde controlamos las rutas del Frontend y administramos de mejor forma las rutas del administrador y el usuario, separando responsabilidades.

En la carpeta de **styles** guardamos los estilos de cada apartado.

En la carpeta de **utils** guardamos un archivo JSON el cual guarda atributos de Tailwind que nos ayudan a tener en cuenta ciertos diseños.

Base de datos

APIs y Endpoints

Hemos utilizado APIs como FastAPI y REST API WordPress.

LOGIN FastAPI

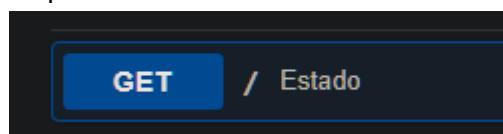
Esté es el /docs de la API FastAPI, con sus rutas a la cual puedes acceder con este enlace <https://fast-api-login-six.vercel.app/docs>, que controla las rutas del backend de Login y solo utilizamos el Auth de Supabase.

The screenshot shows the 'default' endpoint documentation for the API. It includes:

- A 'GET' method for the '/Estado' endpoint.
- A 'POST' method for the '/login' endpoint, labeled 'Login Endpoint'.
- A 'POST' method for the '/registrar' endpoint, labeled 'Registrar Brigadista Endpoint'.
- A 'POST' method for the '/eliminar-usuario' endpoint, labeled 'Eliminar Usuario'.

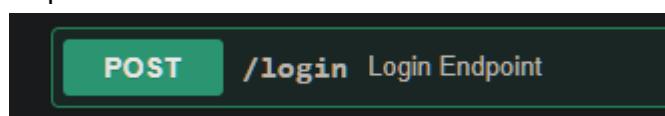
Ruta <https://fast-api-login-six.vercel.app/>

Es para ver el estado actual de la API.



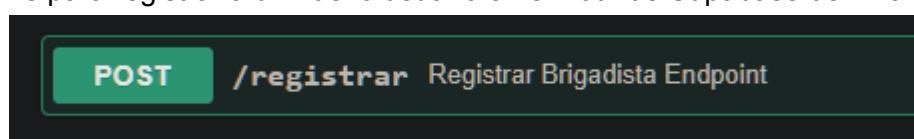
Ruta <https://fast-api-login-six.vercel.app/login>

Es para confirmar las credenciales de un usuario en el Auth de Supabase.



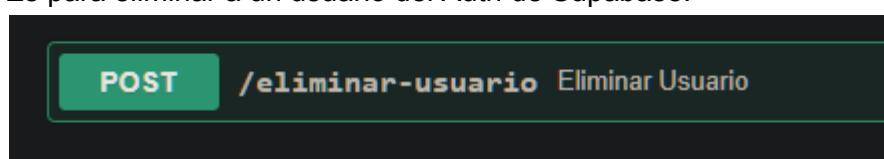
Ruta <https://fast-api-login-six.vercel.app/registrar>

Es para registrar a un nuevo usuario en el Auth de Supabase del microservicio de Login.



Ruta <https://fast-api-login-six.vercel.app/eliminar-usuario>

Es para eliminar a un usuario del Auth de Supabase.



BRIGADAS FastAPI

Esté es el /docs de Login FastAPI, con sus rutas a la cual puedes acceder con este enlace <https://fast-api-brigada.vercel.app/docs>, que controla las rutas del backend como brigadas, conglomerados, usuarios y asignaciones.

The screenshot shows the API documentation for 'Brigadas' using FastAPI. It includes sections for 'default' and other endpoints. The 'default' section lists several endpoints:

- GET / Estado**
- GET /brigadas Ver Brigadas**
- GET /comentarios/{idConglomerado} Ver Comentarios**
- GET /usuarios/{correo} Usuario Informacion**
- POST /brigada-brigadista Brigada Brigadista**
- GET /brigadas-usuario/{usuario_id} Ver Brigadas**
- GET /brigadas/{idbrigada} Ver Brigada**
- POST /crear-brigada Crear Brigada**
- POST /generar-comentario Generar Comentario**
- PUT /usuario-actualizar Usuario Actualizar**
- GET /hoja-vida/{nombre} Hoja De Vida**

Below the 'default' section, there are two more sections:

- POST /registrar Registrar Usuario Endpoint**
- DELETE /comentario/{idComentario} Eliminar Comentario**

Ruta <https://fast-api-brigada.vercel.app/>

Es para ver el estado actual de la API.

The screenshot shows the API documentation for the 'default' endpoint. It displays a single endpoint:

- GET / Estado**

Ruta <https://fast-api-brigada.vercel.app/brigadas>

Es para ver todas las brigadas existentes en la base de datos de Supabase.

The screenshot shows the API documentation for the 'brigadas' endpoint. It displays a single endpoint:

- GET /brigadas Ver Brigadas**

Ruta <https://fast-api-brigada.vercel.app/comentarios/idConglomerado>

Es para ver todos los comentarios referentes a un conglomerado en específico a través de su ID.

GET

/comentarios/{idConglomerado} Ver Comentarios

Ruta <https://fast-api-brigada.vercel.app/usuarios/martin5435412@gmail.com>

Es para ver la información completa de un usuario a través del correo electrónico registrado en el sistema.

GET

/usuarios/{correo} Usuario Informacion

Ruta <https://fast-api-brigada.vercel.app/brigada-brigadista>

Es para solicitar todas los empleados que estan asignados a una brigada.

POST

/brigada-brigadista Brigada Brigadista

Ruta https://fast-api-brigada.vercel.app/brigadas-usuario/usuario_id

Es para ver las brigadas a las que está asignado un usuario a partir de su ID

GET

/brigadas-usuario/{usuario_id} Ver Brigadas

Ruta https://fast-api-brigada.vercel.app/brigadas/brigada_id

Es para solicitar la información completa de una brigada en específico a través de la ID.

GET

/brigadas/{idbrigada} Ver Brigada

Ruta <https://fast-api-brigada.vercel.app/crear-brigada>

Es para crear una brigada en el Supabase.

POST

/crear-brigada Crear Brigada

Ruta <https://fast-api-brigada.vercel.app/generar-comentario>

Es para registrar un nuevo comentario en una brigada.

POST

/generar-comentario Generar Comentario

Ruta <https://fast-api-brigada.vercel.app/usuario-actualizar>

Es para que el usuario presente en la página pueda cambiar su nueva información, tales como región, descripción y teléfono..

PUT /usuario-actualizar Usuario Actualizar

Ruta <https://fast-api-brigada.vercel.app/hoja-vida/nombre>

Es para solicitar la hoja de vida de un usuario en específico, utilizando el nombre del archivo obtenido a través de un atributo guardado en la tabla Usuario.

GET /hoja-vida/{nombre} Hoja De Vida

Ruta <https://fast-api-brigada.vercel.app/registro>

Para registrar a un usuario en la tabla Usuarios de el microservicio de Brigadas.

POST /registro Registrar Usuario Endpoint

Ruta https://fast-api-brigada.vercel.app/comentario/conglomerado_id

Para eliminar un comentario en específico utilizando el ID.

DELETE /comentario/{idComentario} Eliminar Comentario

Mantenimiento y Actualización de la Arquitectura de Microservicios

1. Actualizar API KEYS

Los enlaces a cada API están protegidos en un archivo llamado .env que se encuentra en la carpeta principal de cada servicio, en él podrán ir cambiando los enlaces a las APIs dependiendo si lo requiera.

2. Dependencias actualizadas

Debe estar al tanto del versionamiento de las dependencias y sus actualizaciones, en caso de fallar una dependencia, debe mantenerlas en su última versión para un correcto uso y soporte.

3. Despliegue actualizado en Vercel

Para realizar realizar en el software que está desplegado en Vercel, deberá subir los cambios al mismo repositorio donde se encuentra, ya sea por medio de GitHub Desktop o directamente desde el GitHub, estos cambios automáticamente se verán reflejados en la página desplegada.

Consejos y buenas prácticas de uso

1. Documentar cada cambio en el mismo archivo que está actualizando.
2. Nunca actualice directamente en el servidor de despliegue, primero haga pruebas unitarias en su computador, desplegando de forma local del software y verificando su funcionamiento correcto.
3. Nunca utilices las llaves de APIs o información sensible por fuera del archivo .env por temas de seguridad.

Soporte y contacto

Correo: cpinto5@udi.edu.co, ebeltran7@udi.edu.co

Teléfono: 3114589229

Horario: Lunes a sábado, 9:00 a.m. a 8:00 p.m.