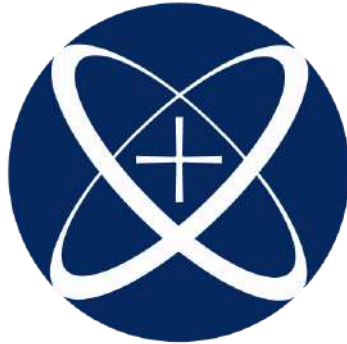


ITESO, UNIVERSIDAD JESUITA DE GUADALAJARA

TECNOLOGÍAS DE DESARROLLO DEL CLIENTE

Profesor: Francisco Sevilla



**ITESO, Universidad
Jesuita de Guadalajara**

Proyecto Integrador

POR

ISAAC ERNESTO VÁZQUEZ SANDOVAL - 745369

MATEO HERNANDEZ GUTIERREZ - 742023

Datos Generales

Link de Proyecto:

<https://neurotype-prod-frontend.onrender.com>

El servidor (no usar este)

<https://neurotype-prod-backend.onrender.com>

Repositorio:

<https://github.com/isaac-evs/neurotype>

(prototipo en angular)

<https://github.com/EurekaTESO/neurotype-angular>

Variables de Entorno: .env

```
SECRET_KEY=t7IzOFk41j736Pmja
ALGORITHM=HS256
ACCESS_TOKEN_EXPIRE_MINUTES=30
SQLALCHEMY_DATABASE_URI=
"postgresql://postgres:rootpostgrespassword@database1.cnmigicey46e.us-east-2.rds.amazonaws.com:5432/postgres"

AWS_ACCESS_KEY_ID=AKIAT4GVRLVWFXD7V5VM
AWS_SECRET_ACCESS_KEY=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
AWS_S3_BUCKET_NAME=neurotype-bucket

OPENAI_API_KEY=sk-proj-E62DvWRzYoXOE2BSNSo47-E89TsuHogI7ZgfQ30ZU3oE4Gvl40yHNho4KYRi59BHDJSqY5f6wST3BibkFJTFFD-jGjuXxb1ShXOgrpDNdl3vfNtYQf2ZEsqkX2KpV0AF_2xDtxuWaakwelulLqYJYr6y8T4A

GOOGLE_CLIENT_ID=556952347377-ujs9djrl3t27lbtvopn3u9fk61ot62j.apps.googleusercontent.com
GOOGLE_CLIENT_SECRET=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
```

Pasos para ejecutar el repositorio:

```
git clone https://github.com/isaac-evs/neurotype.git
```

```
cd neurotype/backend
```

```
touch .env (insertar datos)
```

```
cd ..
```

```
cd ..
```

```
docker-compose build -up
```

<http://localhost:3000/> (Vistas) o <http://localhost:8000/> (Servidor)

Nombre: Neuro Type

Descripción: Es una aplicación diseñada para personas interesadas en cuidar y mejorar su salud mental. Los usuarios pueden registrar sus emociones y pensamientos diarios mediante texto, y la app organiza esta información en un calendario personal, identificando tendencias emocionales a lo largo del tiempo.

La aplicación cuenta con un procesador de lenguaje natural que depende que tipo rol tengas, que detecta tendencias en los textos y genera análisis y gráficas en base a ello, también se conecta con un chatbot de soporte que se adapta a las tendencias emocionales del usuario y lo apoya.

¿Qué hace?:

- Detecta patrones en los textos
- Detecta palabras clave
- Detecta tendencias emocionales
- Ofrece asistencia emocional por medio de chat.

¿Qué NO hace?:

- No es un sistema de gestión de pacientes
- No es un CRM, ni planificador de sesiones

Tecnologías

Backend:

FastAPI:

Será el núcleo del backend para procesar textos mediante APIs rápidas y asíncronas, manejando eficientemente las operaciones de análisis de lenguaje natural.

PostgreSQL:

Se utilizará PostgreSQL como base de datos relacional para almacenar de forma estructurada y eficiente las transcripciones, notas y datos del usuario. Es ideal para manejar información que requiere relaciones entre tablas (usuarios, roles, entradas, etc.).

Procesamiento de Lenguaje Natural

SpaCy:

Utilizaremos esta biblioteca robusta para tareas avanzadas de procesamiento de texto, como análisis sintáctico, extracción de palabras clave y detección de emociones.

Hugging Face API:

Como complemento o alternativa, Hugging Face API proporcionará modelos preentrenados avanzados para análisis de textos, especialmente en casos donde se requiera mayor personalización.

Frontend**React:**

React permitirá una interfaz dinámica y responsiva, ideal para una experiencia fluida y moderna.

TailwindCSS:

Nos ayudará a mantener estilos modulares y altamente personalizables en la aplicación.

Roles de Usuario y flujos:**Versión Lite (Gratis)**

Esta versión está dirigida a usuarios que desean monitorear su bienestar emocional de forma personal y diaria. Ofrece funcionalidades básicas de análisis de texto para detectar y gestionar emociones mediante herramientas sencillas.

Flujo de trabajo:

- El usuario puede escribir notas diarias sobre su estado emocional o importar texto.
- Las notas son procesadas por **spaCy**, que detecta emociones y patrones básicos de sentimientos.

- Los resultados se visualizan en gráficos que muestran tendencias emocionales de manera sencilla.

Roles de usuario

Lite:

Diseñado para usuarios que desean una experiencia simple y accesible. Este rol permite:

- Registrar y analizar textos de manera básica para identificar emociones clave.
- Visualizar tendencias emocionales en un calendario personal.
- Ideal para quienes buscan una introducción al análisis de emociones sin funcionalidades avanzadas.

Plus:

Pensado para usuarios que buscan un análisis más profundo y personalizado. Este rol incluye:

- Acceso a herramientas avanzadas de análisis emocional utilizando modelos de NLP más sofisticados, como Hugging Face.
- Identificación de tendencias emocionales complejas, con detalles como intensidad emocional y palabras clave relacionadas.
- Un chatbot personalizado que adapta sus respuestas y sugerencias a las tendencias emocionales detectadas.
- Gráficas y reportes avanzados que permiten una visión clara de los patrones emocionales a lo largo del tiempo.

Módulos y niveles de acceso:

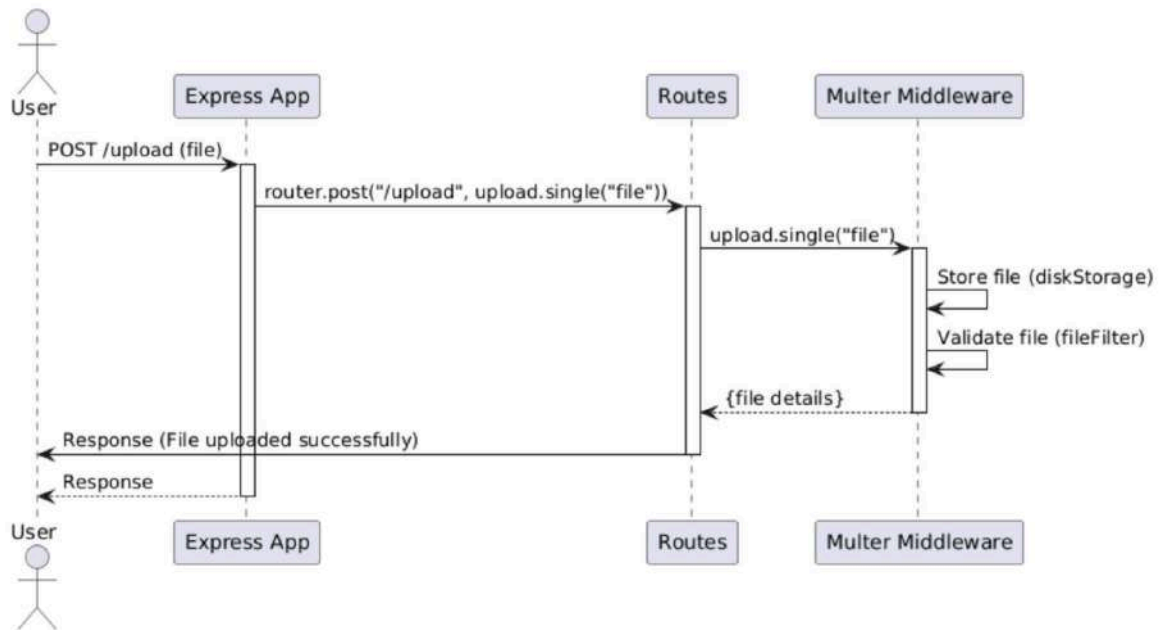
Módulo	Acción	Lite	Plus
Autenticación			
	Registro	✓	✓
	Inicio de sesión	✓	✓
Gestión de Notas			
	Exportar notas	✓	✓
	Ingresar texto manualmente	✓	✓
Análisis de Texto			
	Análisis básico con spaCy	✓	
	Análisis avanzado con BERT/GPT		✓
	Detección de emociones y sentimientos básicos	✓	✓
	Detección de variaciones sutiles y patrones complejos		✓
Visualización de Datos			
	Gráficos sencillos de evolución emocional	✓	✓
	Gráficos avanzados e interactivos	✓	✓
	Líneas de tiempo de evolución emocional	✓	✓

Diagrama E-R:

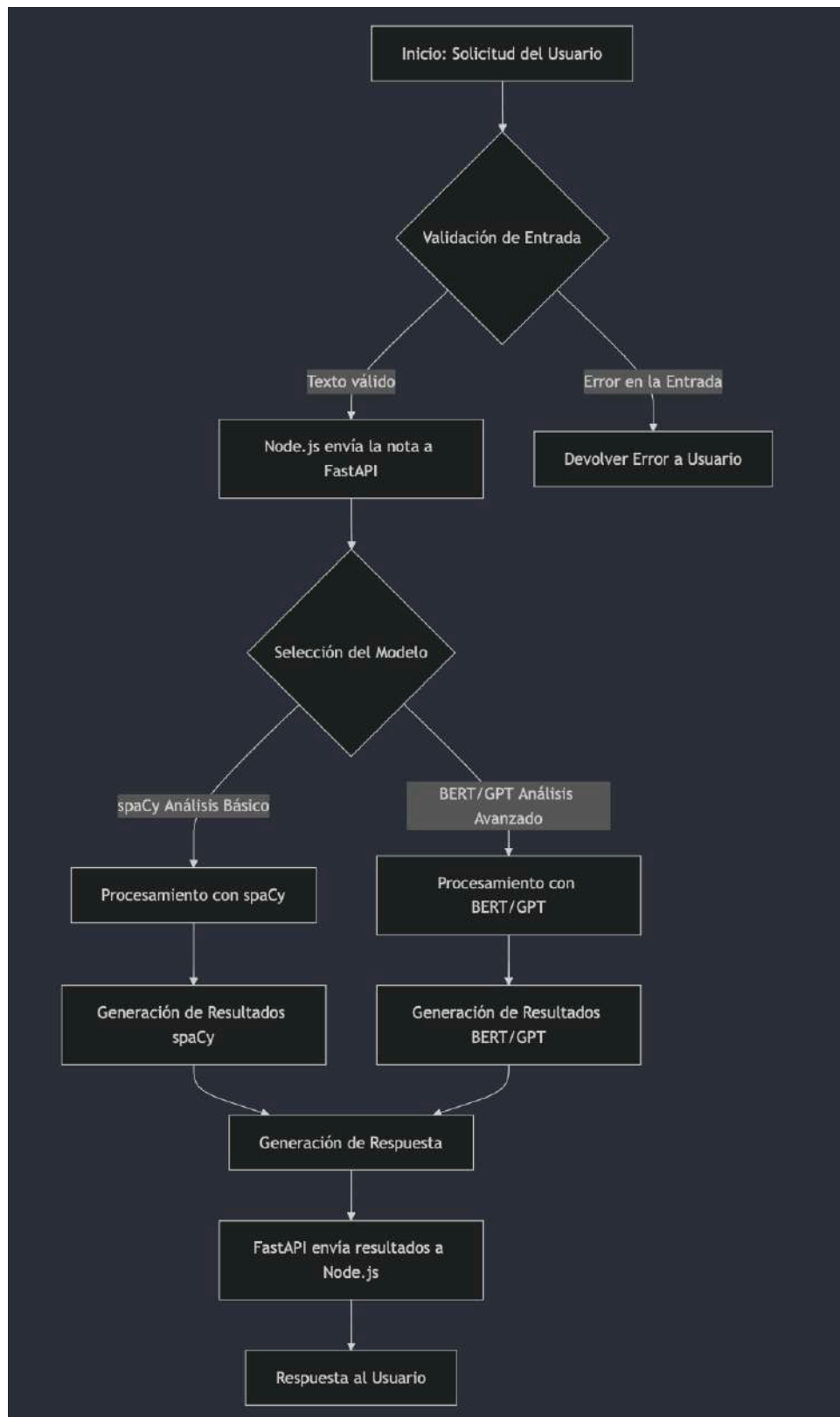


Diagrama de Flujo:

Subir notas



Análisis de notas:



Swagger:

users			^
POST	/register	Register a new user	v
POST	/login	Login and get access token	v
PUT	/select-plan	Select a plan	🔒 v
PUT	/profile	Update user profile	🔒 v
POST	/auth/google	Authenticate with Google	v
GET	/users/me	Get current user	🔒 v
notes			^
GET	/notes/	Read Notes	🔒 v
POST	/notes/	Create Note	🔒 v
PUT	/notes/{note_id}	Update Note	🔒 v
DELETE	/notes/{note_id}	Delete Note	🔒 v
GET	/notes/daily-analysis	Get Daily Analysis	🔒 v
GET	/notes/emotions-summary	Get Emotions Summary	🔒 v
dashboard			^
GET	/dashboard/	Get Dashboard	🔒 v
recommendations			^
GET	/recommendations/	Get personalized recommendations	🔒 v
data			^
GET	/data/export	Export user data	🔒 v

+ /Chat con Socket.IO

link:

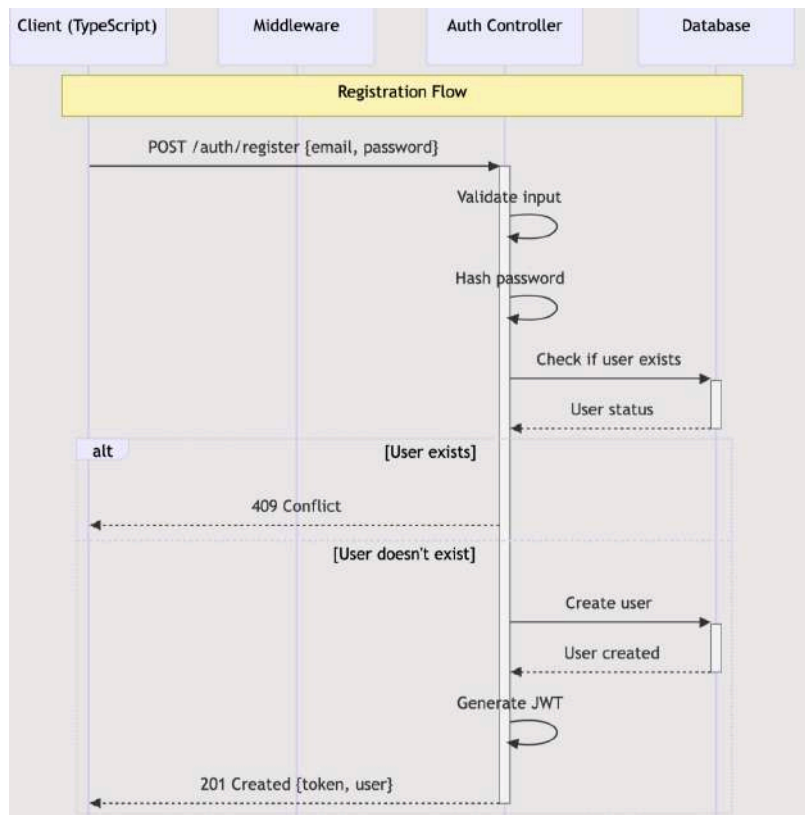
<https://neurotype-prod-backend.onrender.com/docs>

Explicación de cómo se realizó la configuración de Swagger

Swagger ya viene configurado por defecto con FastAPI, se añadieron descripciones y detalles a cada endpoint

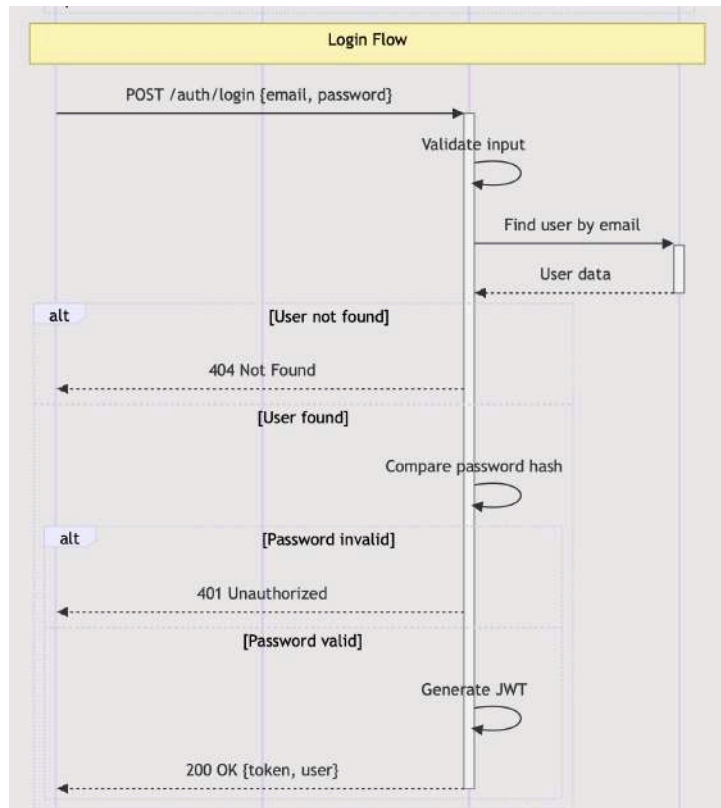
Autenticación de Servicios:

Registro:



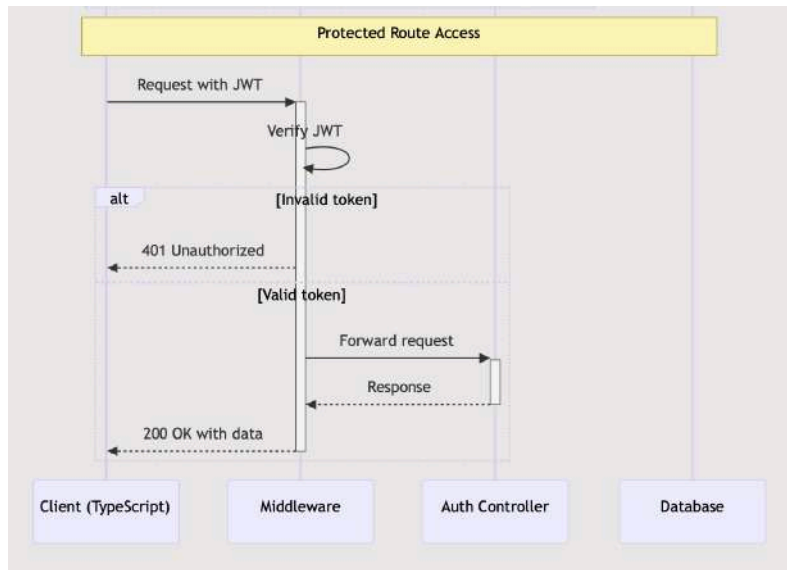
- El usuario crea mediante POST un formulario su cuenta, con
 - email
 - password
- El input es validado
- La contraseña es hasheada
- El usuario se guarda en la base de datos
- Se genera un token de sesión

Inicio de sesión:



- El usuario se loguea mediante un POST con los datos:
 - email
 - password
- El input es extraído y validado
- Se busca de manera asincrónica en la base de datos un usuario con el mail proporcionado
- Si el usuario no se encuentra aparece "Invalid login credentials"
- Si se encuentra se compara el hash de la contraseña, si no coincide da lo siguiente: "Invalid login credentials"
- Si se encuentra, genera un token y se envía como respuesta

Rutas Protegidas:



Autenticación:

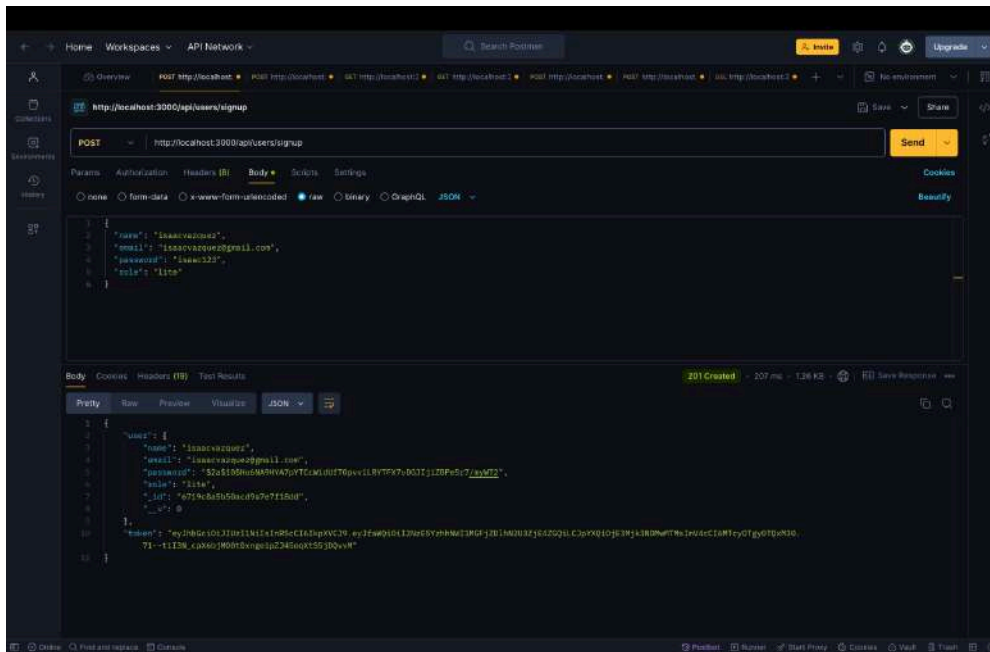
- Extrae JWT
 - Verifica la existencia del token, si no hay: "No token provided"
 - Decodifica el token
 - Busca el usuario en la base de datos
 - Verifica la existencia del usuario
 - Asigna el usuario a req.user
-

Autorización:

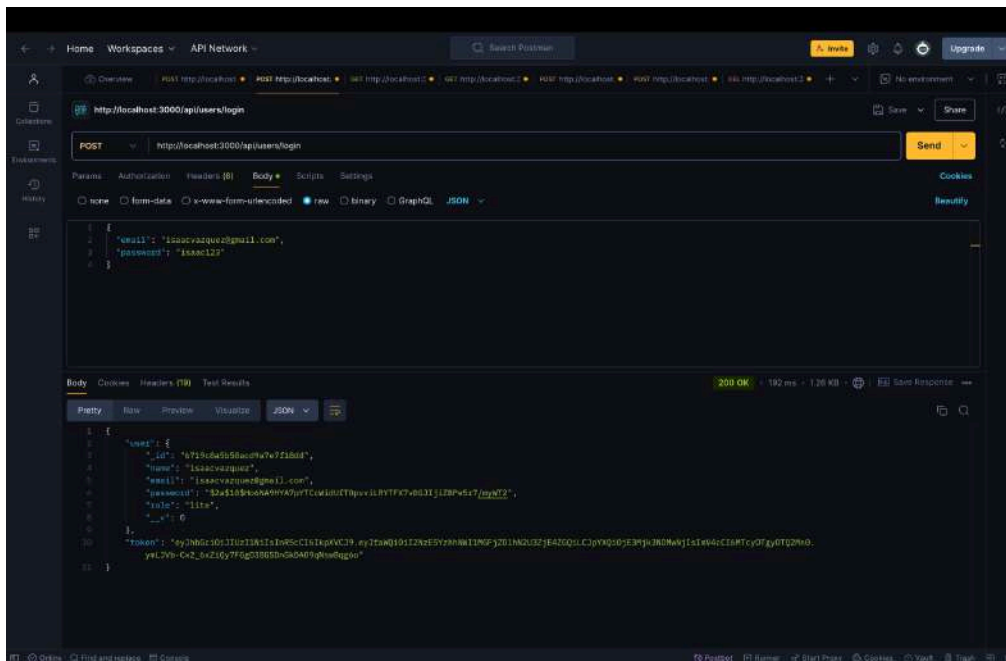
- Verifica si el usuario está autenticado
- Si no está autorizado da = 401 Unauthorized
- Si está autenticado, verifica que el rol del usuario este en la lista de permitidos, si no lo está da = 403 Forbidden
- Si está autenticado, sigue con next()

Ejemplo de validación y capturas:

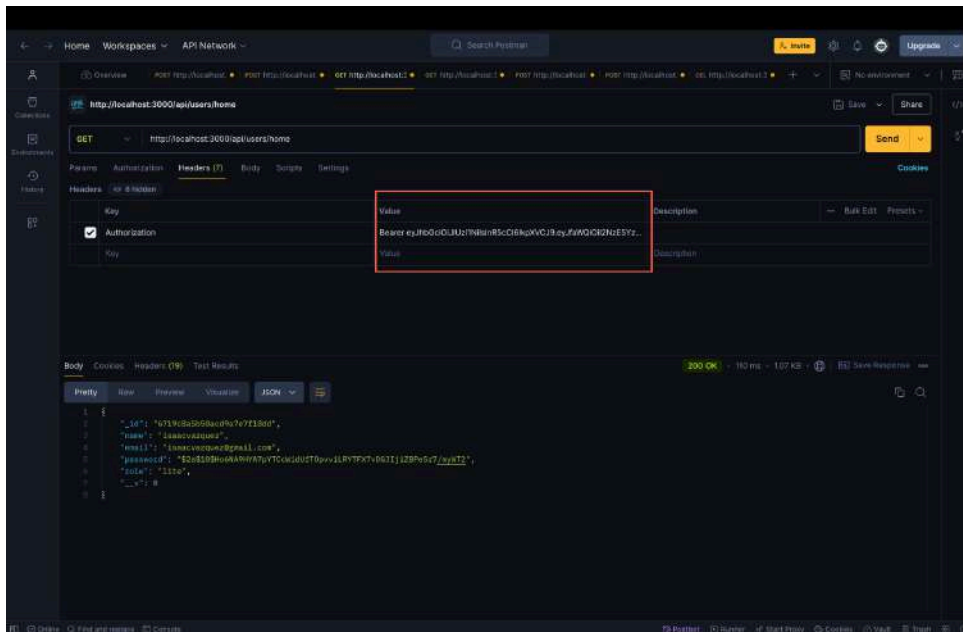
1. El usuario "Isaac", crea su cuenta:



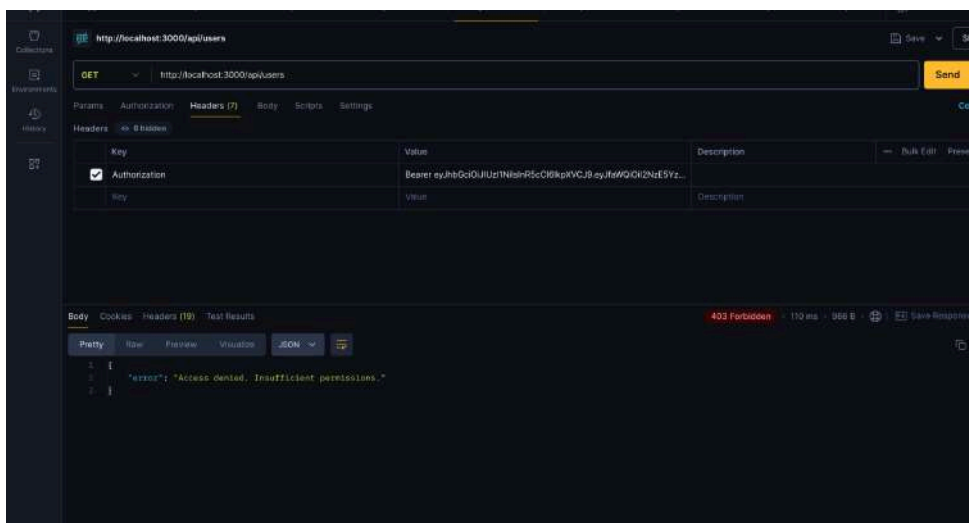
2. El usuario "Isaac", se logea:



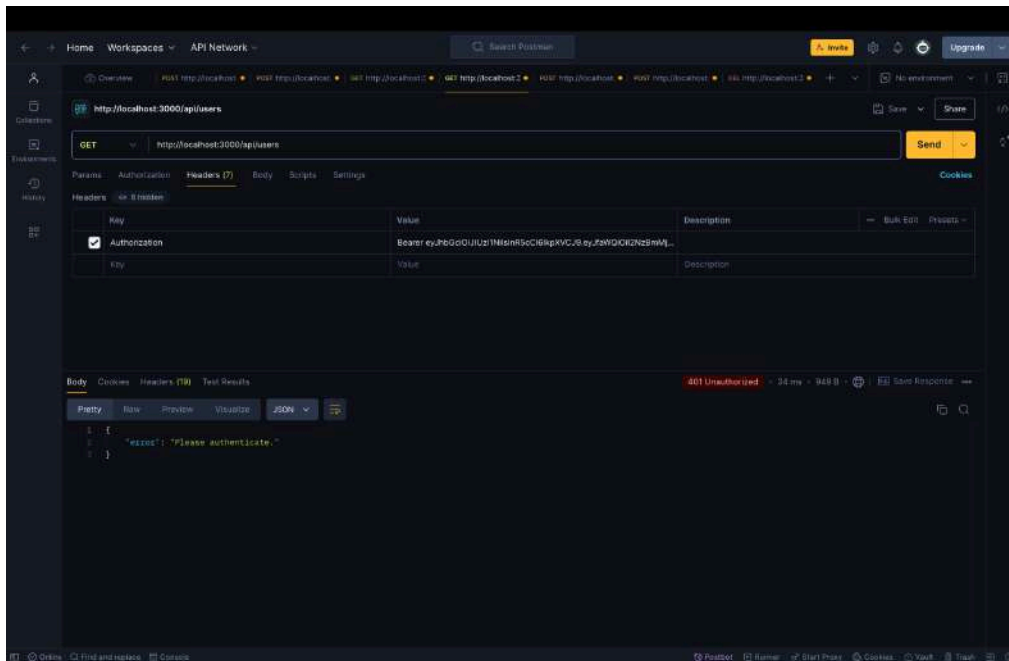
3. El usuario "Isaac"; entra a la sección home donde es **autenticado** mediante su token



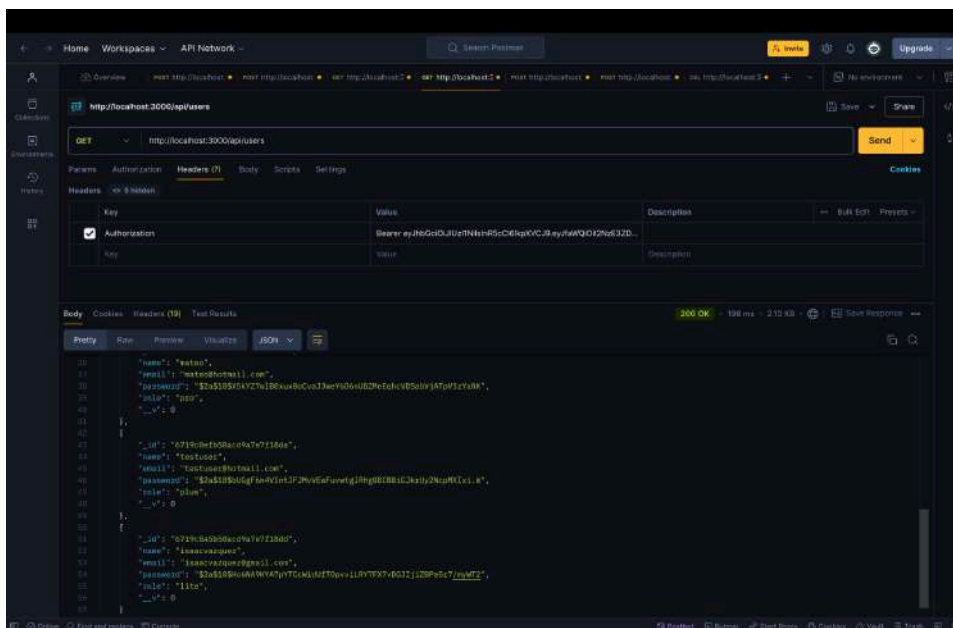
4. El usuario "Isaac"; desea conseguir la lista de todos los pacientes en la base de datos mediante su token, pero al ser versión "lite" no puede, por lo que es **autenticado sin permisos**



5. El usuario "Mateo" al ser usuario "pro" que posee los permisos necesarios, desea conseguir la lista de todos los pacientes en la base de datos , pero no ha iniciado sesión en mucho tiempo por lo que **no es autenticado**.



- El usuario "Mateo" decide volver a iniciar sesión, por lo que se le otorga un nuevo token válido y vuelve a pedir los pacientes de la base de datos si ningun problema, lo que significa que está **autenticado y con permisos**.



Envio de Imagenes por medio de Formularios:

Update Your Profile

Name

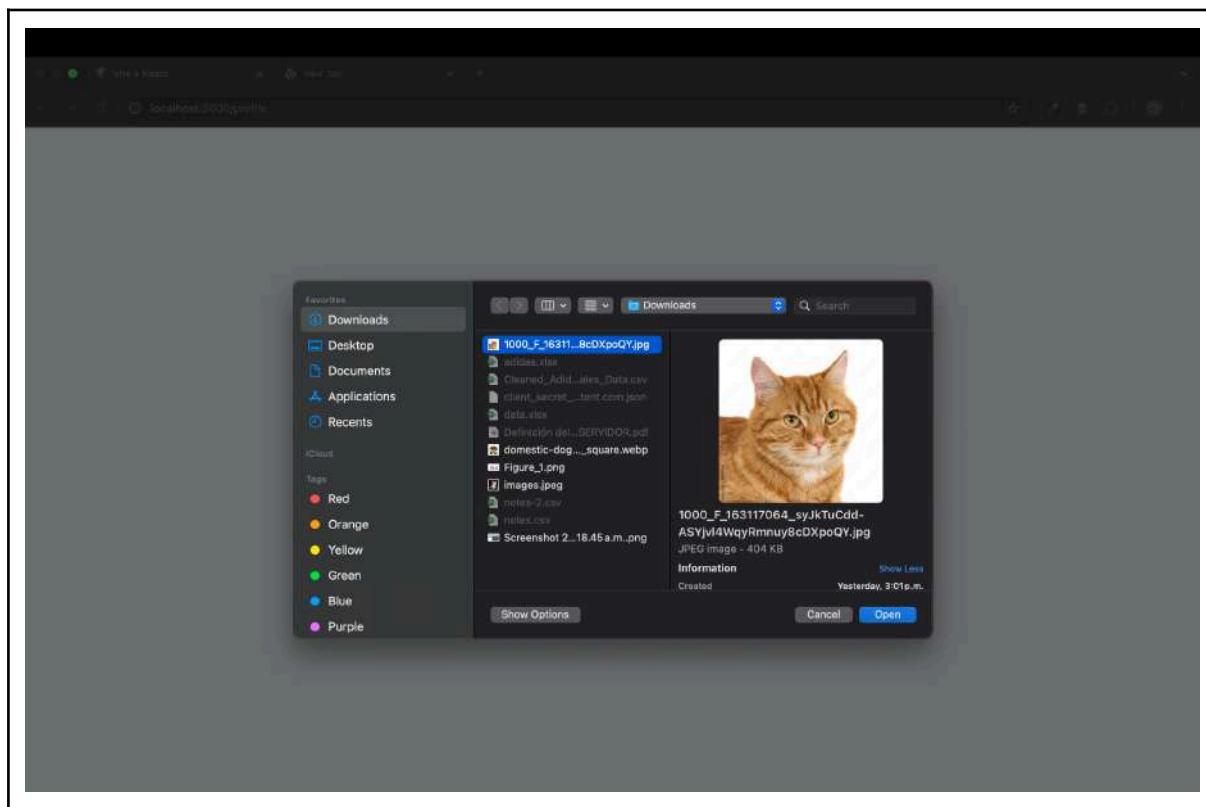
Usuario

Profile Photo

Choose File

No file chosen

Save Profile



Dashboard



Welcome, Usuario

Total Notes: 0

Notes

Calendar

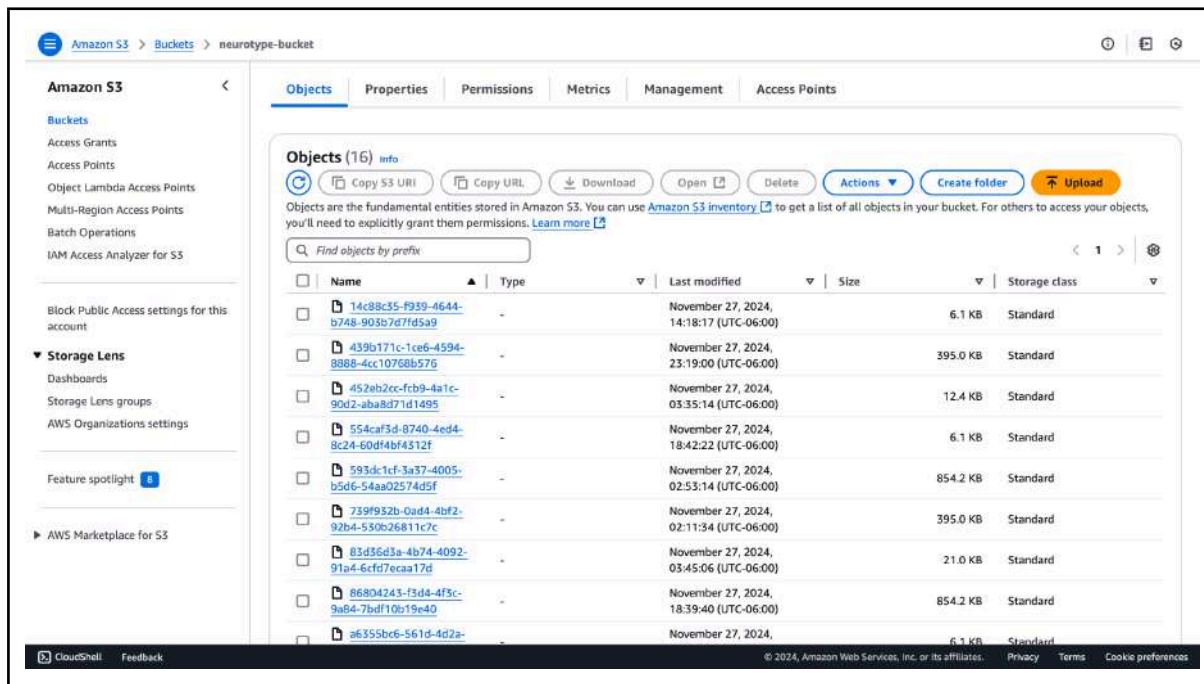
Export Data

C

Prevalent Emotion Today

happy

Objetos en el Bucket S3:



The screenshot shows the Amazon S3 console interface for a bucket named 'neurotype-bucket'. The left sidebar contains navigation options like Buckets, Access Grants, and Storage Lens. The main area is titled 'Objects (16)' and includes a search bar and a table of objects. The table columns are Name, Type, Last modified, Size, and Storage class. The objects listed are:

Name	Type	Last modified	Size	Storage class
14c88c35-f939-4644-b748-903b7d71d5a9	-	November 27, 2024, 14:18:17 (UTC-06:00)	6.1 KB	Standard
439b171c-1ce6-4594-8888-4cc10768b576	-	November 27, 2024, 23:19:00 (UTC-06:00)	395.0 KB	Standard
452eb2cc-fcb9-4a1c-90d2-aba8d71d1495	-	November 27, 2024, 03:35:14 (UTC-06:00)	12.4 KB	Standard
554caf3d-8740-4ed4-8c24-60d4bf4312f	-	November 27, 2024, 18:42:22 (UTC-06:00)	6.1 KB	Standard
593dc1cf-3a37-4005-b5d6-54a02574d5f	-	November 27, 2024, 02:53:14 (UTC-06:00)	854.2 KB	Standard
739f952b-0a44-4bf2-92b4-530b26811c7c	-	November 27, 2024, 02:11:34 (UTC-06:00)	395.0 KB	Standard
83d36d3a-4b74-4092-91a4-6cfd7ecaa17d	-	November 27, 2024, 03:45:06 (UTC-06:00)	21.0 KB	Standard
86804243-f3d4-4f5c-9a84-7bdf10b19e40	-	November 27, 2024, 18:39:40 (UTC-06:00)	854.2 KB	Standard
a6355bc6-561d-4d2a-	-	November 27, 2024,	6.1 KB	Standard



The screenshot shows a web browser window with the address bar displaying the URL: <https://neurotype-bucket.s3.us-east-2.amazonaws.com/439b171c-1ce6-4594-8888-4cc10768b576>. The main content area displays a close-up image of an orange tabby cat with yellow eyes. The browser's taskbar at the bottom shows several open applications, including a terminal window with the command 'curl -O https://neurotype-bucket.s3.us-east-2.amazonaws.com/439b171c-1ce6-4594-8888-4cc10768b576'.

Ruta:

<https://neurotype-bucket.s3.us-east-2.amazonaws.com/439b171c-1ce6-4594-8888-4cc10768b576>

Captura de pantalla uso de Sockets (Socket.IO - REACT) FrontEnd:

Evento de conexión Main.py

```
@sio.event
async def connect(sid, environ):
    query_params = environ.get('asgi.scope').get('query_string', b'').decode()
    token = None
    for param in query_params.split('&'):
        if param.startswith('token='):
            token = param.split('=')[1]
            break
    if token is None:
        await sio.disconnect(sid)
        return
    user = await authenticate_user(sid, token)
    if user is None:
        await sio.disconnect(sid)
        return
    connected_users[sid] = user
    print(f"User {user.id} connected with SID {sid}")
```

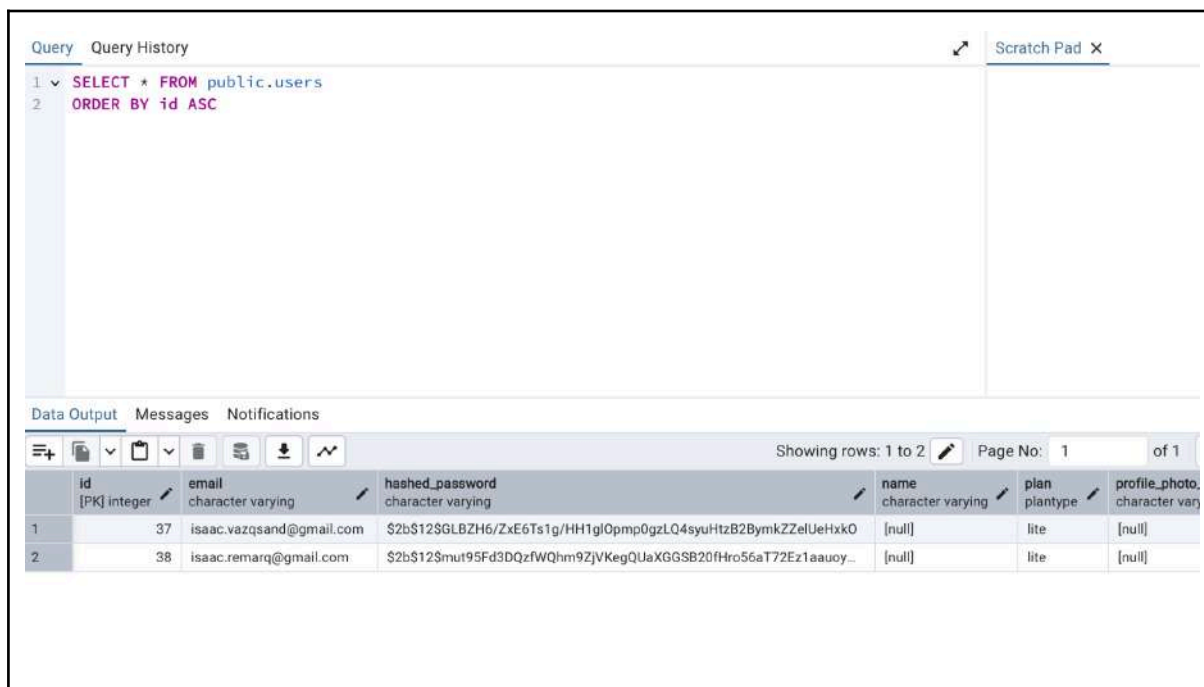
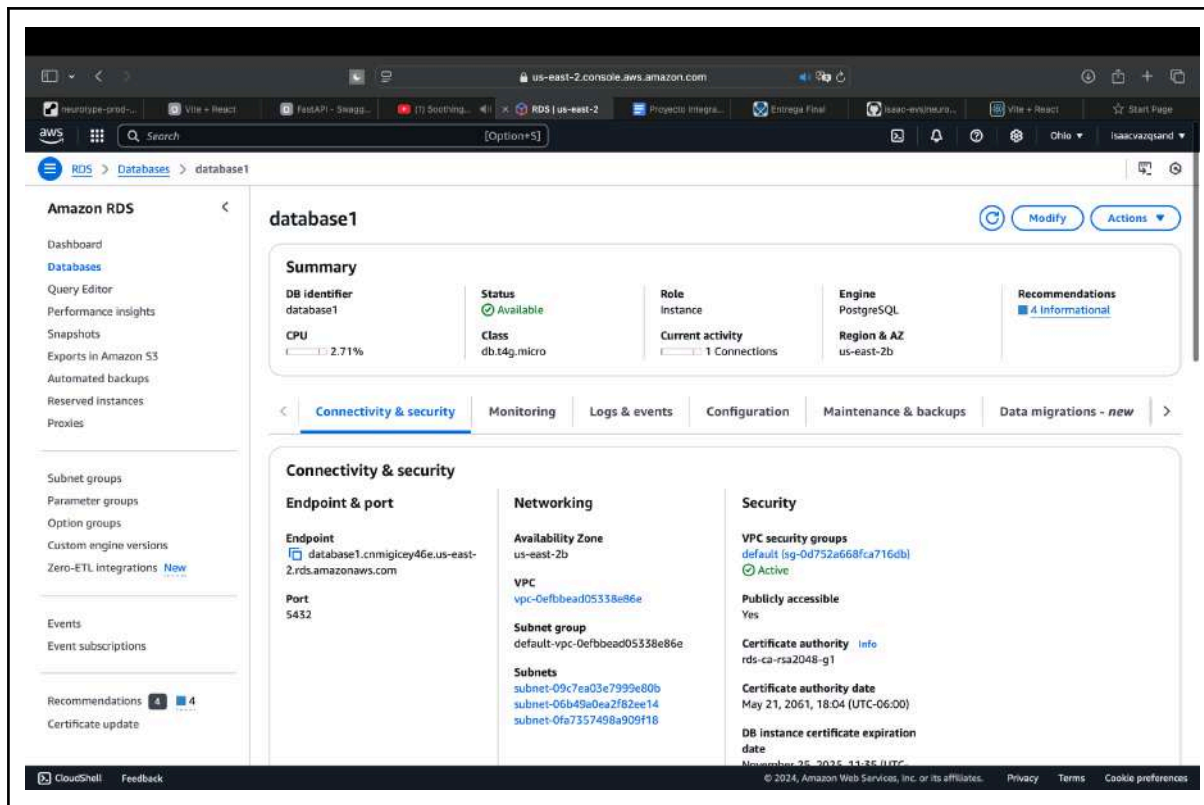
Conexión desde react ChatPage.jsx

```
// Login Handler
const handleLogin = async (e: React.FormEvent<HTMLFormElement>) => {
    e.preventDefault();
    const form = e.target as HTMLFormElement;
    const username = (form.login_username as HTMLInputElement).value;
    const password = (form.login_password as HTMLInputElement).value;

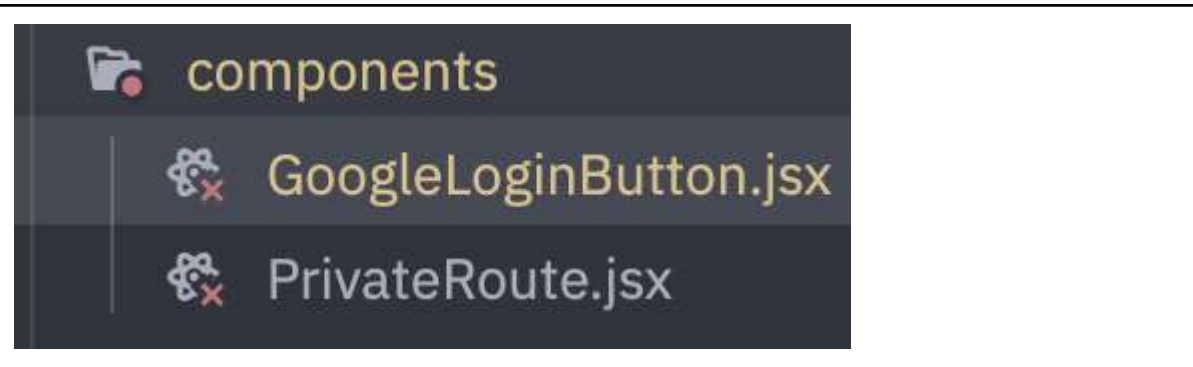
    try {
        const formData = new URLSearchParams();
        formData.append("username", username);
        formData.append("password", password);

        const response = await fetch("/token", {
            method: "POST",
            body: formData,
        });
        const result = await response.json();
        if (response.ok) {
            setToken(result.access_token);
            setCurrentUser(username);
        } else {
            alert("Login failed");
        }
    } catch (error) {
        console.error("Login error:", error);
    }
}
```

Base de Datos (RDS):



Componentes:



Directivas:

ejemplo:



Servicio:



Guardias:

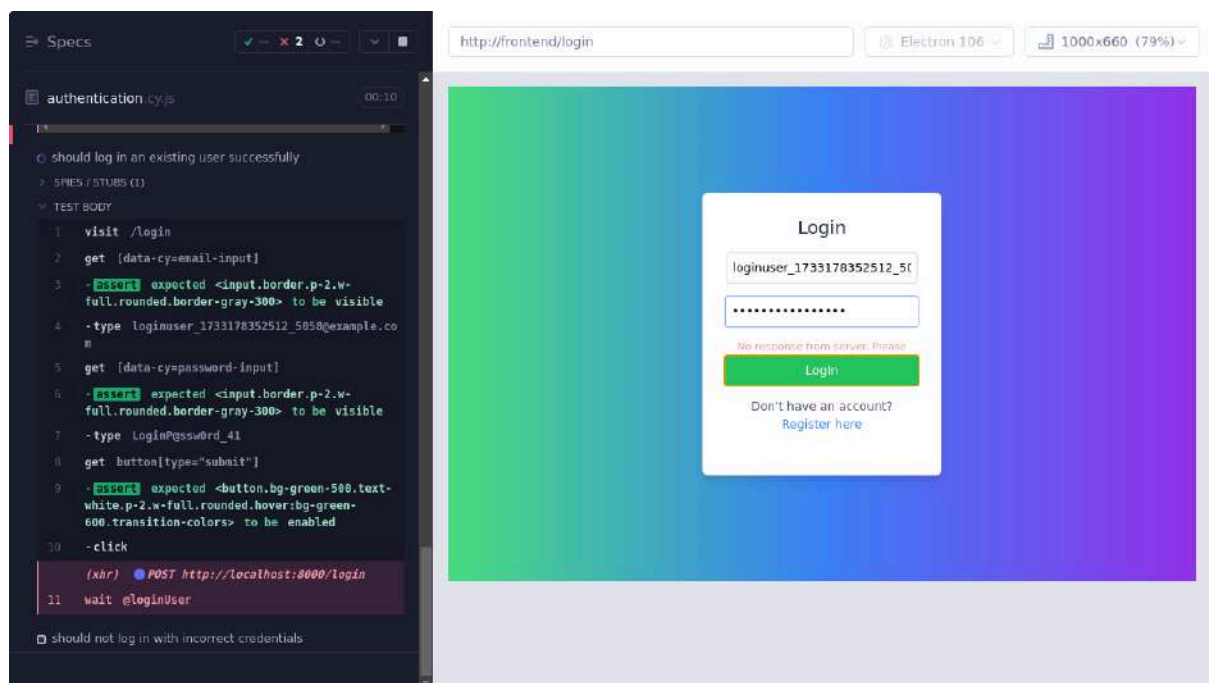
```
2 import React, { useContext } from "react";
3 import { AuthContext } from "../context/AuthContext";
4 import { Navigate } from "react-router-dom";
5
6 export const PrivateRoute = ({ children }) => {
7   const { token } = useContext(AuthContext);
8   return token ? children : <Navigate to="/login" />;
9 };
10
```

Pruebas:

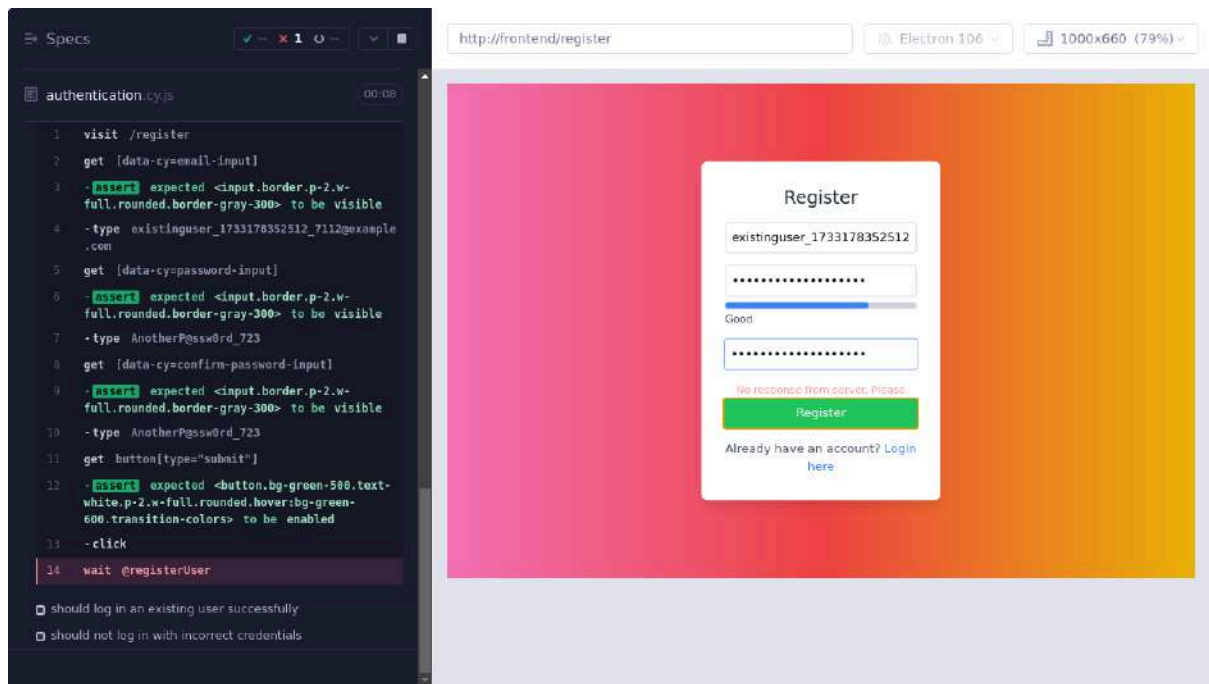
Cypress:

No funciona debido a que no pude hacer que funcione con docker, se necesita un host diferente a localhost, intente con rutas relativas con NGINX o con el host interno de docker, pero el host no respondía a pesar de que los registros funcionan. Solo funciona si modifico el axios a que use el mismo host de cypress, pero en ese caso la app ya no funciona.

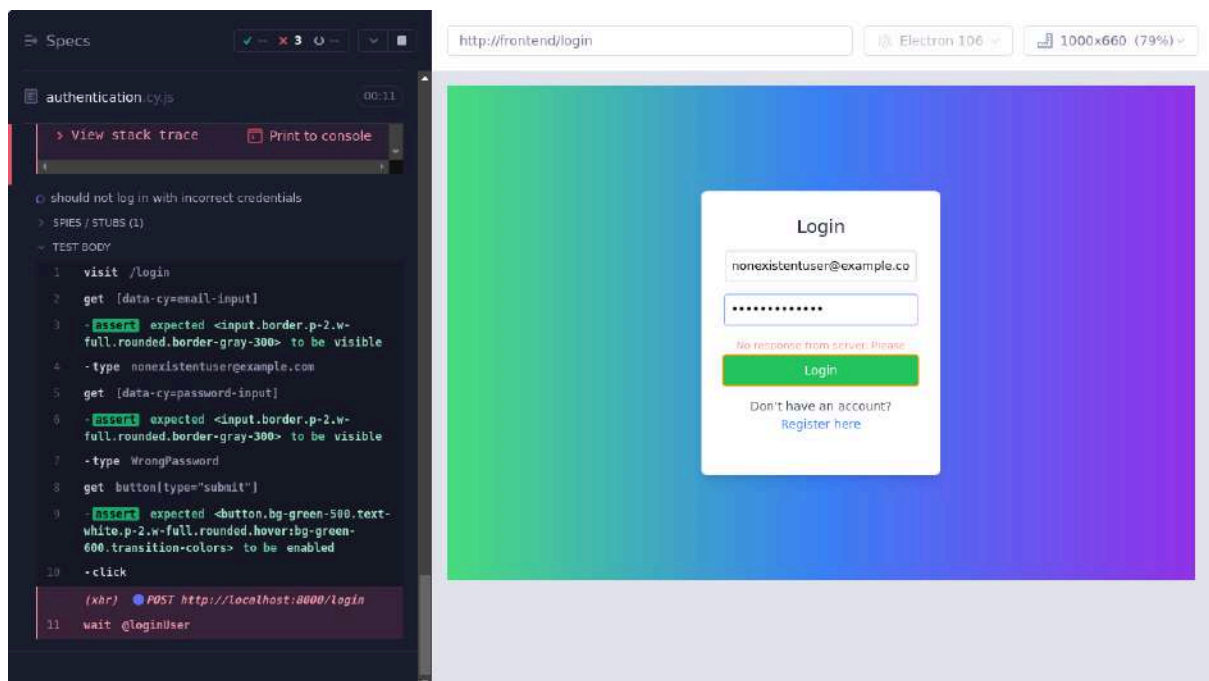
login existing user



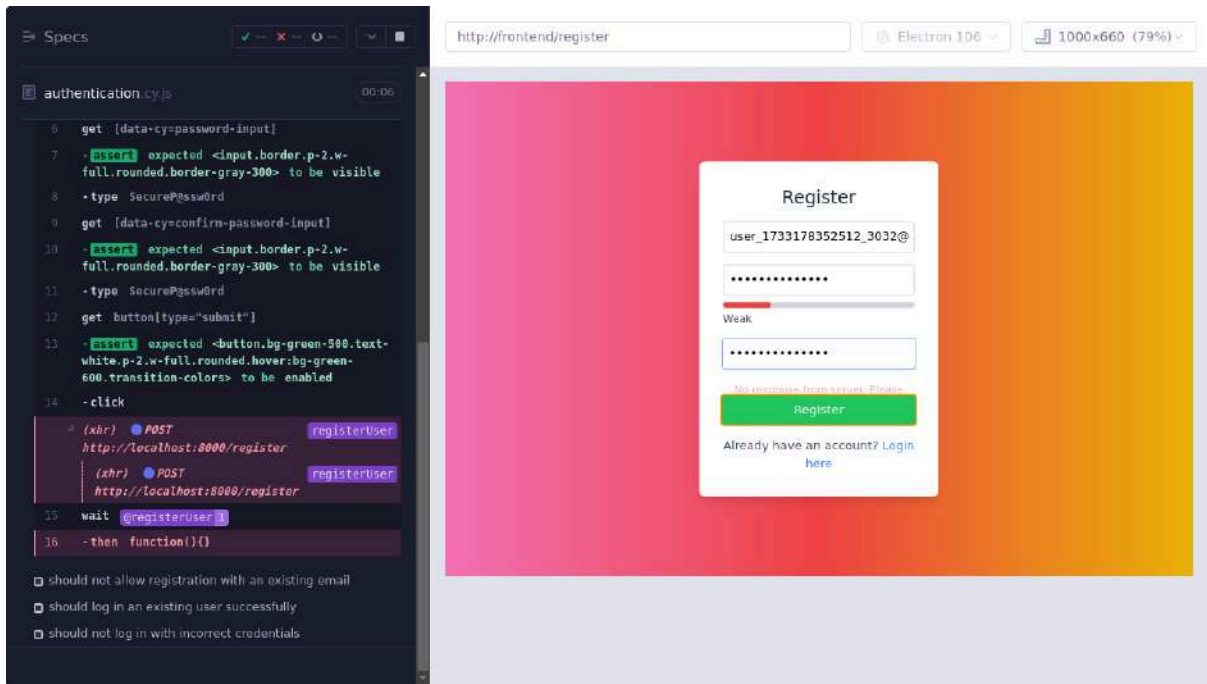
register new user



should not login with incorrect credentials



should not allow registration with similar mail



Jest:

```
cd frontend/neurotype

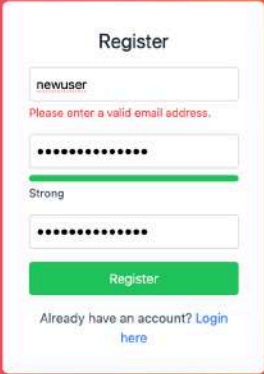
npm i

npm test
```

resultado:

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	85.96	50	81.81	85.96	
api	37.5	0	0	37.5	
axiosInstance.js	37.5	0	0	37.5	11-18
components	86.36	50	100	86.36	
GoogleLoginButton.jsx	84.21	33.33	100	84.21	40,45-49
PrivateRoute.jsx	100	100	100	100	
context	100	100	100	100	
AuthContext.jsx	100	100	100	100	
Test Suites: 1 failed, 2 passed, 3 total					
Tests: 3 failed, 5 passed, 8 total					
Snapshots: 0 total					
Time: 1.254 s					
Ran all test suites.					

Validaciones:



Register

newuser

Please enter a valid email address.

.....

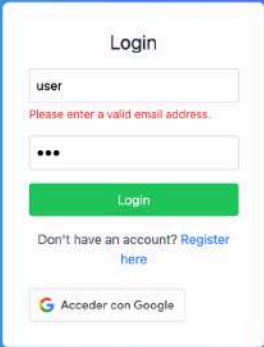
Strong

.....

Register

Already have an account? [Login here](#)

The Register form is centered on a pink-to-yellow gradient background. It contains a title 'Register', an email input field with the text 'newuser', a red error message 'Please enter a valid email address.', a password input field with masked characters '.....', a strength indicator 'Strong' with a green bar, another masked password input field, a green 'Register' button, and a link 'Already have an account? Login here'.



Login


user

Please enter a valid email address.

...

Login

Don't have an account? [Register here](#)

 Acceder con Google

The Login form is centered on a green-to-purple gradient background. It contains a title 'Login', an email input field with the text 'user', a red error message 'Please enter a valid email address.', a password input field with masked characters '...', a green 'Login' button, a link 'Don't have an account? Register here', and a button with the Google logo and text 'Acceder con Google'.

Register

newuser@gmail.com

••••••••



Fair

••••••••••

Passwords do not match.

Register

Already have an account? [Login here](#)

Register

newuser@gmail.com

•••

Password must be at least 8 characters long.



Weak

•••

Register

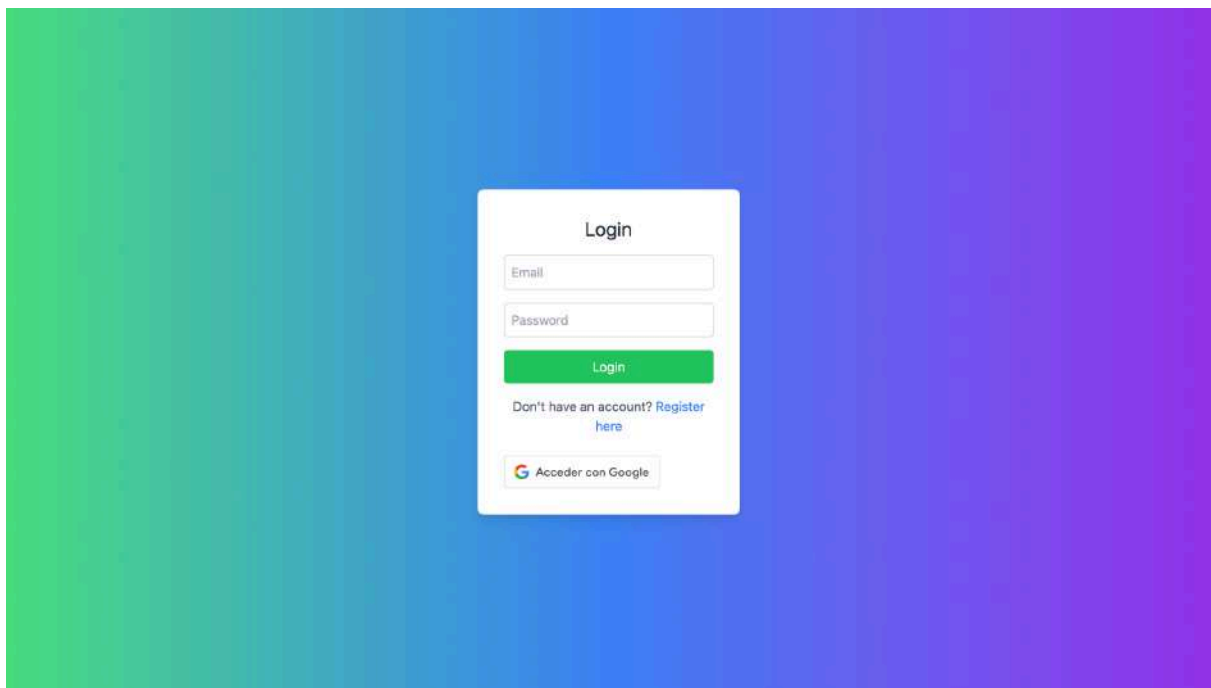
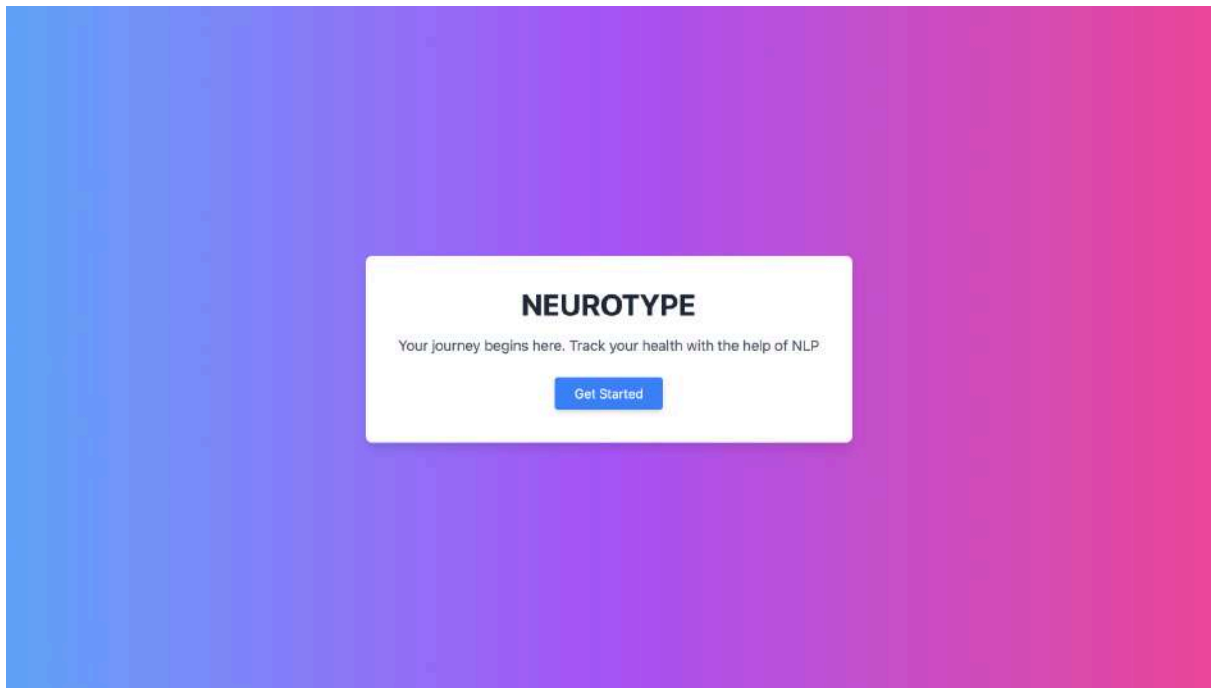
Already have an account? [Login here](#)

Diseño Vistas:

Prototipo:

<https://www.figma.com/proto/T6WKODdgy5h3B1kq2WIXuC/NeuroType-Wireframe?node-id=1-596&t=2MyA7oZ3kuaVTfNg-1>

Producción:



Register

Register

Already have an account? [Login here](#)

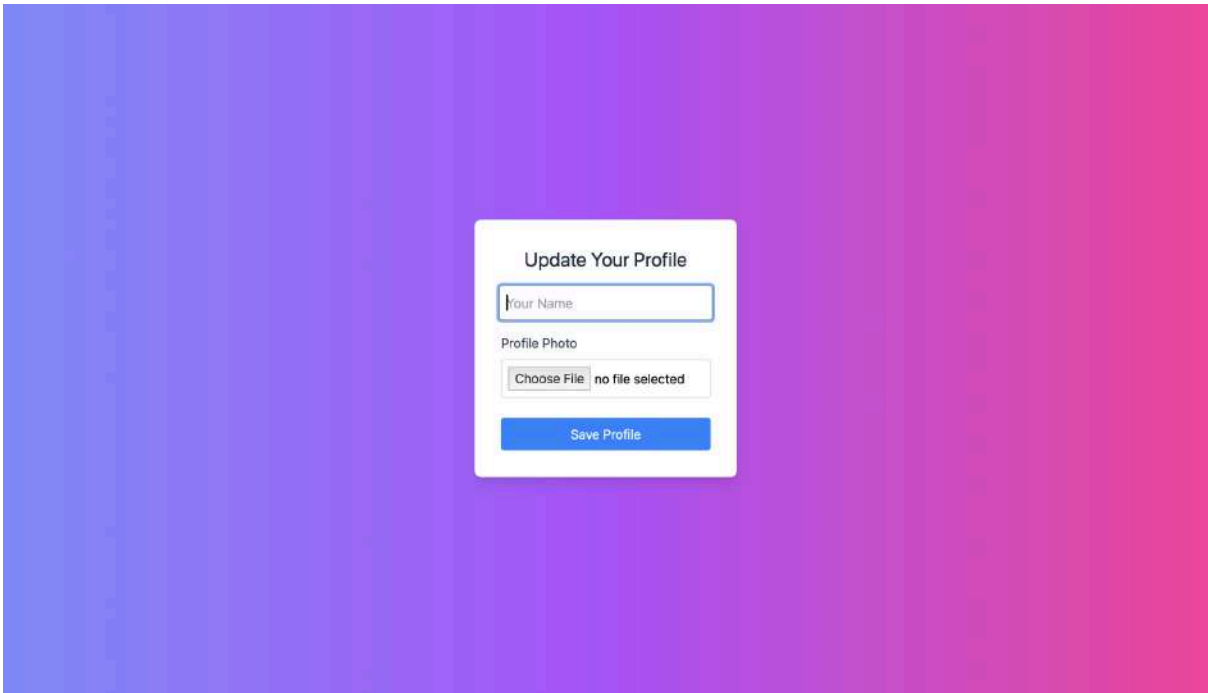
Select Your Plan

Choose a Plan:

Plus - \$9.99/month

Confirm Plan

[← Back to Dashboard](#)



Dashboard

Logout



Welcome, Isaac
Total Notes: 4



Notes



Calendar



Export Data



Chat

Prevalent Emotion Today

happy

Emotion Counts This Week

2

Happy

1

Calm

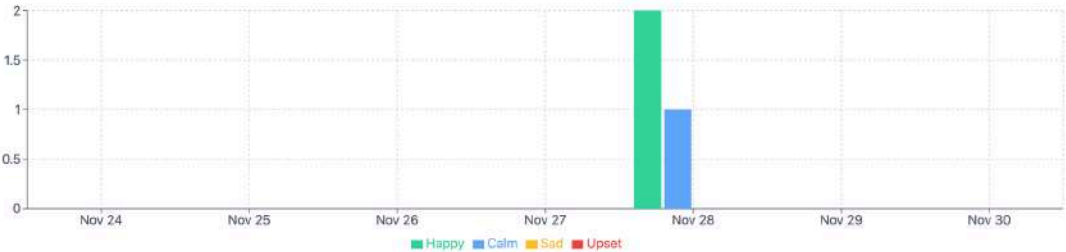
0

Sad

0

Upset

Weekly Emotion Chart



[← Back to Dashboard](#)

[+ New Note](#)

Your Notes

Untitled Note

I am so happy today!

Untitled Note

I feel really calm too!

Untitled Note

I am joyfull

Untitled Note

It feels so refreshing

[← Back to Notes](#)

Im shining like a star

Save

[← Back to Dashboard](#)

Calendar

Today
Back
Next

November 2024

Month
Week
Day

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	31	01	02
03	04	05	06	07	08	09
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

[← Back to Dashboard](#)

Chat

How do I feel today?

You seem to be feeling content today. How are you doing?

Thanks, I have been feeling great lately, my therapy is going great and I feel like life is getting better day by day

That's wonderful to hear! It's great that you're feeling positive and that therapy is going well for you. Remember to continue practicing self-care and reaching out for support if you ever need it. Keep up the good work!

However, yesterday I had a panic attack, what can I do if I have another one?

I'm sorry to hear that you experienced a panic attack. If you have another one, here are some tips that may help: 1. Practice deep breathing: Focus on taking slow, deep breaths to help calm your mind and body. 2. Use grounding techniques: Try to focus on your surroundings by using your senses (e.g., noticing the texture of an object, listening to sounds around you). 3. Challenge negative thoughts: Remind yourself that panic attacks are temporary and that you have the strength to get through it. 4. Practice mindfulness or meditation: Engaging in mindfulness exercises can help you stay present and reduce anxiety. 5. Reach out for support: Talk to a trusted friend, family member, or mental health professional for support and

Thank you! :)

You're welcome! If you ever need support or someone to talk to, feel free to reach out. Take care!

Type your message...
Send

Registro e Inicio de Sesión:

Registro

A registration form titled "Register" is centered on a background with a horizontal gradient from pink to orange. The form contains the following elements:

- Email input field with "user@email.com" entered.
- Password input field with masked characters "••••••••".
- A blue progress bar and the text "Good" below the password field.
- Confirm Password input field.
- A green "Register" button.
- A link: "Already have an account? [Login here](#)".

Inicio de Sesión (Google o mail)

A login form titled "Login" is centered on a background with a horizontal gradient from green to purple. The form contains the following elements:

- Email input field with "isaac" entered, followed by a red error message: "Please enter a valid email address."
- Password input field.
- A green "Login" button.
- A link: "Don't have an account? [Register here](#)".
- A button with the Google logo and the text "Acceder con Google".

Dashboard una vez logueado:

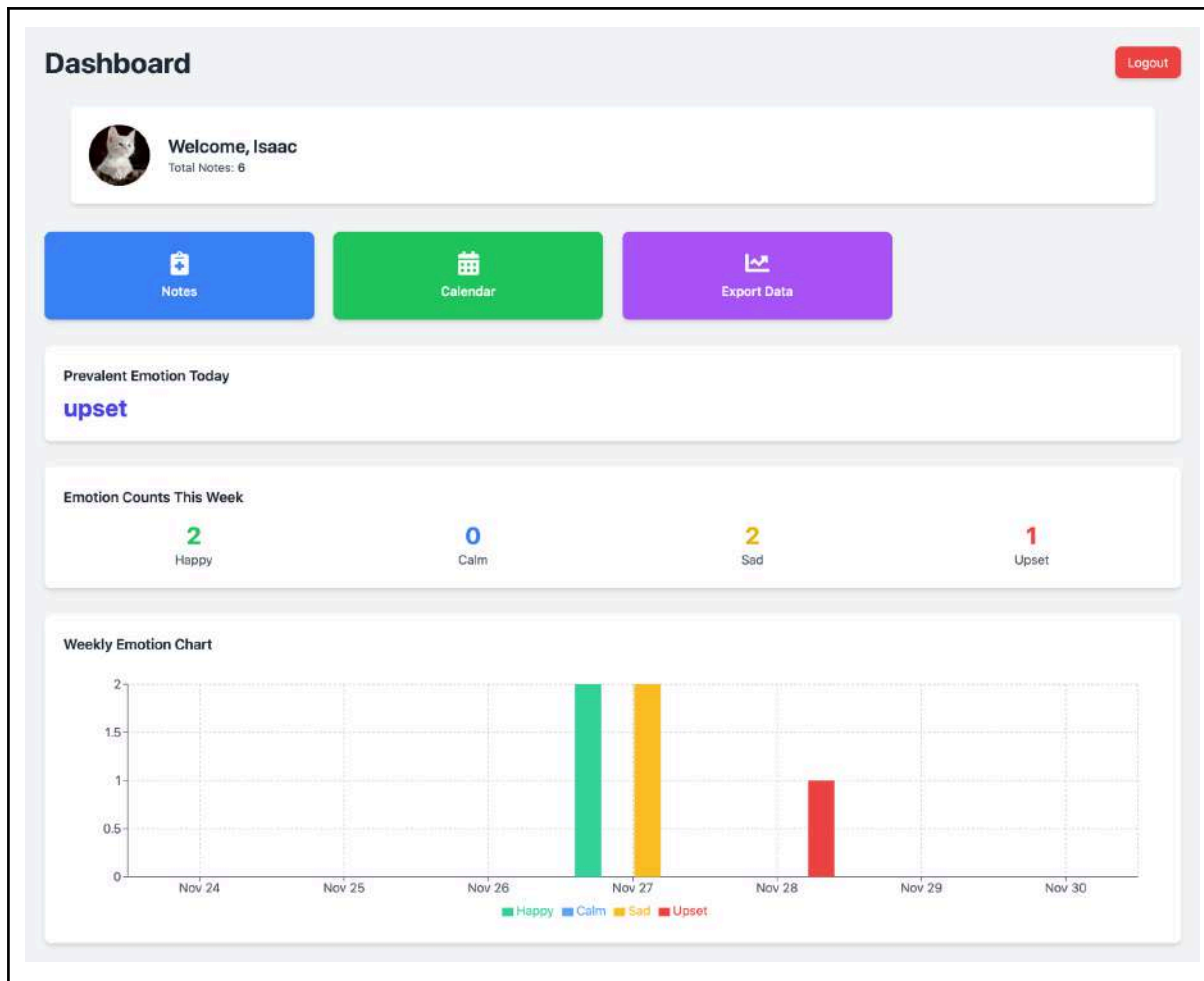


Diagrama de login/register:

