

ERD (Entity-Relationship Diagram) for the PostgreSQL database. We'll include the following tables:

1. `users` - To store user information.
2. `support_cases` - To store the details of each support case.

## ERD

### Users Table

- `user_id` (Primary Key)
- `username`
- `email`
- `password_hash`
- `created_at`
- `updated_at`

### Support Cases Table

- `case_id` (Primary Key)
- `title`
- `description`
- `status`
- `created_at`
- `updated_at`
- `user_id` (Foreign Key referencing `users`)

Base de datos:

```
DATABASE_URL =  
"postgresql://support_user:yourpassword@localhost/support_case_db"
```

URLS front:

<http://localhost:3000/track-case>

<http://localhost:3000/create-case>

URL back:

<http://localhost:8000/>



So I need to create an app (including backend, frontend, test, and database) that basically do the following to 1: the user should be able to create support cases that will be storage into a transactional data base, for example: registrer a new bulking agent or a new client (please put the basic information that will be needed on the colums of the database) 2. track the support cases that are already created. I have to be able to write, read, change or do any action that I might require. For this app I have the following requeriments:

for backend: create an ERD on the data base of the table in postgresql where the data will be store for each support case or requirement. Some of the fields for the tables are: database, eschema, query, user, (give me more examples of data that I can use for this case) Also I need yo create an API using FastAPI to manage all the tables where we only can insert and read data. this using python.

for frontend: using typescript and react, create a form where the user can place all the information that will be store on the postgres database. using endpoints post. Also I need another form where the user can track and old support case so he can get the info from the database including filters.

all of this will be test on localhost so please all the endpoints are meant for localhost. Also I will be using VisualStudio Code. Please give me all the files step by step and organized as a project



NataliaFigueroaT24

i have this at the frontend side: import axios from 'axios';

```
const API_URL = 'http://localhost:8000';
```

```
export const createSupportCase = async (caseData: any) => { const response = await axios.post(`${API_URL}/support_cases/`, caseData);  
return response.data; };
```

```
export const getSupportCase = async (caseId: number) => { const response = await axios.get(`${API_URL}/support_cases/${caseId}`); return  
response.data; };
```

this at the backend: # CORS configuration origins = [ "http://localhost:3000", ]

```
app.add_middleware( CORSMiddleware, allow_origins=origins, allow_credentials=True, allow_methods=["*"], allow_headers=["*"], )
```

```
app.include_router(user.router) app.include_router(support_case.router)
```


#### GitHub Copilot

Given the frontend and backend code snippets you've provided, your setup seems generally correct for a scenario where the frontend (running on <http://localhost:3000>) communicates with a backend (running on <http://localhost:8000>). The CORS configuration on the backend side appears to be set up to allow requests from the frontend.

However, if you're encountering a "Network Error" when making requests from the frontend to the backend, here are a few specific areas to check based on the provided code:

1. **Backend Server Running:** Ensure that the backend server is running and accessible at <http://localhost:8000>. You can test this by using tools like Postman or curl to make a request directly to the backend URL.
2. **Correct Endpoint Paths:** Verify that the paths you're trying to access on the backend (`/support_cases/` and `/support_cases/{caseId}`) are correctly defined in your backend router. The error might occur if the paths in the frontend requests do not match the actual

How can I troubleshoot and debug network errors in my frontend application?

 GitHub Copilot




Hi @NataliaFigueroaT24, how can I help you?



NataliaFigueroaT24

tell me why i am getting this error: ERROR Network Error AxiosError: Network Error at XMLHttpRequest.handleError  
(http://localhost:3000/static/js/bundle.js:49557:14) at Axios.request (http://localhost:3000/static/js/bundle.js:49995:41) at async  
createSupportCase (http://localhost:3000/static/js/bundle.js:125:20) at async handleSubmit  
(http://localhost:3000/static/js/bundle.js:199:20) when trying to interact with the frontend

 GitHub Copilot

The error message "AxiosError: Network Error" typically indicates an issue with the network request made from your frontend application using Axios. Here's a step-by-step breakdown of potential causes and checks:

1. **CORS Policy:** The most common issue is related to the Cross-Origin Resource Sharing (CORS) policy. If your frontend and backend are served from different origins (different domains, ports, or protocols), the browser will block the request unless the server includes the appropriate CORS headers.
2. **Server Not Running:** Ensure that the backend server you're trying to reach is up and running. If the server is down or the URL is incorrect, Axios will not be able to make a successful request.