API Overview:

The function of this API is to power the AI Weather Stylist application. It then provides endpoints to retrieve real-time weather data and generate outfit recommendations using OpenAI, and it also enforces a secure credit-based system for user access and functionality.

Endpoint Documentation:

Method	Endpoint	Description	Auth	Rate Limit
GET	/weather?city=…	Fetches live weather info	Yes	3 credits/day
POST	/ask-ai	Gets AI suggestion based on weather	Yes	1 credit/use

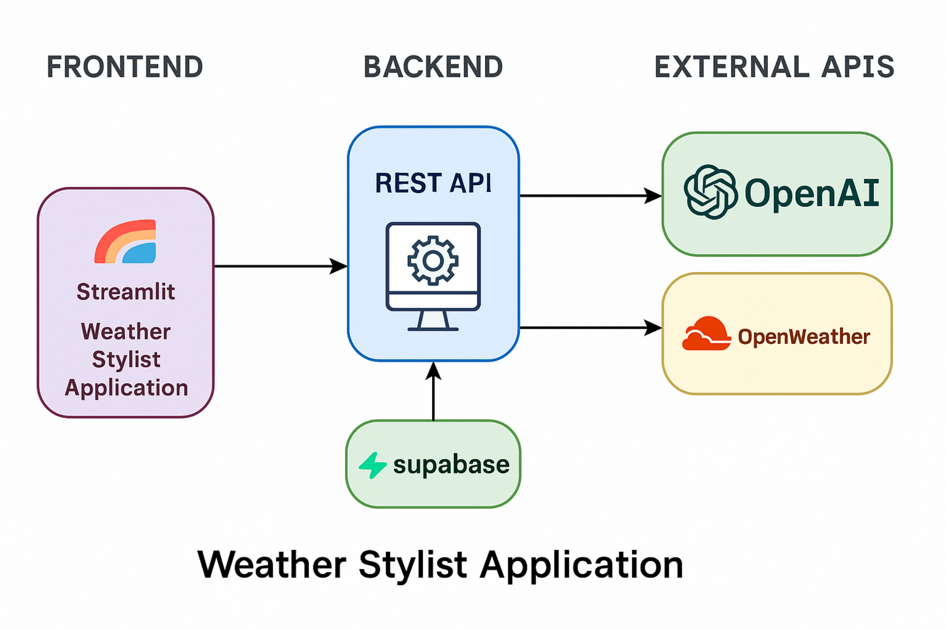
Authentication:

Supabase is used to secure authentication. Users must create an account, then log in and receive a session token, which is included in the Authorization header for every request made by the user.

Credit Limiting:

When a user successfully creates an account, the user is given a limited credit. If credit runs out, the user is directed to purchase more credits, and further API calls are blocked until they reset every 24 hours or the user makes a credit purchase.

System Architecture Diagram:



Components Explained:

- **Client (Streamlit Web App) →** User interface for login, weather check, and AI outfit recommendation
- **Auth (Supabase) →** Handles user login/signup and session management
- Backend API (FastAPI or Flask) → Custom API that :
  - Proxies weather request from OpenWeather
  - Sends prompts to OpenAI for outfit suggestions.
  - Enforces credit limits
- **External APIs**:
  - OpenWeather for real-time weather data
  - OpenAI for outfit generation
- **Database (Supabase) →** Stores user data and credit usage
- **Deployment**: Dockerized and hosted on Render/Vercel for Streamlit