Second Prompt for ChatGPT

Alright, so here is what we do now.

The goal is to build now the entire structure that allows the following:
-A user can sign up and create an account
-The user gets an email to confirm his email address
-When the user confirms the address he gets redirected to the login page and is automatically on a free subscription (Initial Influencer creation, 3 separate image creations per month, limitless chatting)
-After the user logged in he sees in the dashboard the amount of credits and an info which subscription he has (free subscription)
-When the user clicks a button to subscribe to the pro subscription the stripe webhook allows the checkout and after successful payment the user gets redirected to the dashboard, where he sees now that he has the pro subscription (20 image creation per month, limitless chatting, gets 20 credits every month until subscription gets cancelled)
-The dashboard doesn't contain any settings or buttons relevant for the actual function of the saas yet. Its a provisionally dashboard just to validate and set up the entire sign up / login process. The dashboard gets redesigned later.

For the Flask app we go with these minimal routes for now:

- POST /api/upgrade → start Checkout (first purchase)
- POST /api/billing-portal → create a Portal session URL for management
- POST /api/stripe/webhook → reflect changes in your DB (free ↔ pro)
- GET /api/me → show current plan/status

For now we need the following html pages located at the following domains:

-sign-up located: https://gptsweetheart/sign-up
-login located: https://gptsweetheart/login
-redirect: located: https://gptsweetheart/redirect
-dashboard: located: https://gptsweetheart/dashboard

I show you again the exact schema of my Supabase data base. In a first step I want you to only give me the four minimal routes we need in a python script, that also already imports the right libraries etc. The goal is to have an entire minimalistic working Flask app that can already run on my server and with which I can already test the entire sign up and login process.

Give me just the py script and then in a bullet point outlook, show me what else needs to be done afterwards to get the app running. Show all the little steps that need to be done on the server (Nginx, Config Files, Environments etc) to get the app running in the next steps.

Also outline which static html pages we need and what needs to be done to get those rendered correctly as well.

Right now on my server there is only docker, nginx and n8n installed, nothing else as far as I know.

Take as much time as you need to create a fantastic result. This is very very important to me. Think step by step.