**Next steps -** Get working locally in development mode. Then send to git in production mode.

Expo web app

Buttons not working

**Optimize recipe quality is goal**

Must deploy 2 web services on Redner, (1) for backend (RecipeGenerator-API) and (2) another for the frontend (RecipeGnerator).

Replace your-secure-secret-key with a random, secure string (e.g., generate one using python -c "import secrets; print(secrets.token\_hex(16))").

1d4b23f8785c0a88020a65b633792b08

**Start-local.bat and deploy.bat simplify local debugging and enhancement and deployment to Render.**

**Deploy.bat doesn’t fully work…..stops before git commands.**

**Last step is Render hosting**

Grok is modifying these files

C:\Users\bshoe\OneDrive\Desktop\game\_theory\cooking\RecipeGenerator\app\(tabs)\index.tss

And

C:\Users\bshoe\OneDrive\Desktop\game\_theory\cooking\RecipeGenerator\app.py

There is 2 render website projects, for front and back ends. It’s way complex… can it be simplified?

Modularized on 4/6/2025

C:\Users\bshoe\OneDrive\Desktop\game\_theory\cooking\RecipeGenerator\

│

├── app.py # Backend: Main Flask app

├── recipes\_data.py # Backend: Recipes and flavor pairs

├── recipe\_generator.py # Backend: Recipe generation logic

├── helpers.py # Backend: Utilities

│

└── app/

└── (tabs)/

├── index.tsx # Main frontend component

├── RecipeCard.tsx # Recipe display component

├── FavoritesList.tsx # Favorites display component

├── AffiliateSection.tsx # Affiliate products section

├── InputSection.tsx # Input fields and pickers

├── useRecipe.ts # Recipe fetching and state management

├── useFavorites.ts # Favorites management

├── data.ts # Common ingredients, styles, categories

├── styles.ts # Styles