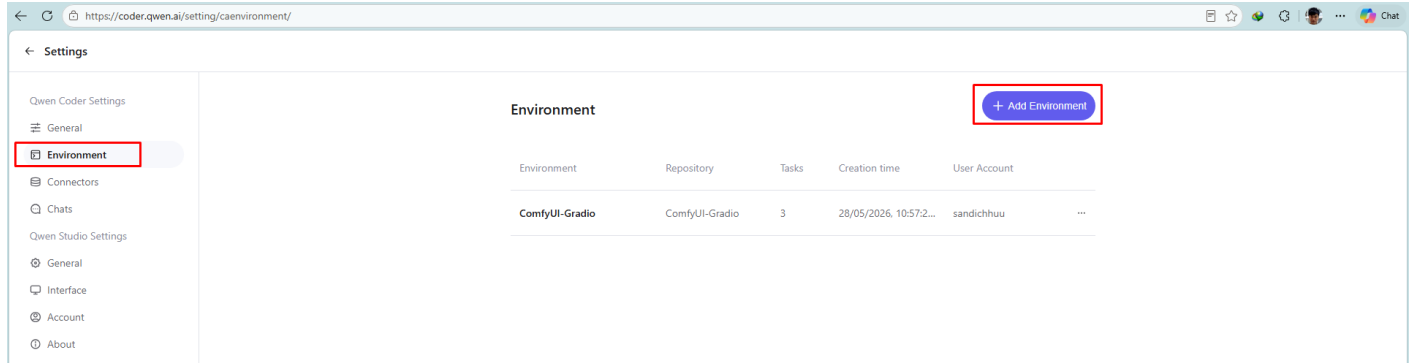


? Qwen Coder Workflow

? Quick Start

First we need link github repo. Becareful, this action will grant Qwen access to your repos.



Then we need some markdown files describe how the application working.
Just typing on common language, doesn't need to write technical language.

implement.md

```
This project is using Python (uv), using gradio with comfyui server to generate images.  
If the input / output image too big, set fix-height = 444px  
Make sure the web still working if refresh browser.  
When running, the Generate button have to changed to Stop button.
```

```
The main UI defined on main.py and contain 2 tabs.  
tab1: ZIT.  
tab2: Flux2-Klein-9B
```

```
to implement:  
tab1: zit.md.  
tab2: flux2_klein_9b.md
```

```
loRA list have collapse function.  
this one can add more loRA item.  
each loRA has field name, weight and remove button.  
if press genegrate button, loRA tag like <name:1.0> will be merge into prompt then  
send. (but without see on prompt field).
```

flux2_klein_9b.md

```
This document describe the Flux module (Flux2-Klein-9B) working.
```

File structure:

1. views/flux2_klein_9b_view.py
2. workers/flux2_klein_9b_worker.py

flux2_klein_9b_view: The gradio implement UI renderer.

flux2_klein_9b_worker: The code extract from comfyui, support start inference.

flux2_klein_9b_view layout:

```
| prompt: helloworld | Generate / Stop button |  
| config: seed (with toggle random value), width, height, toggle_ref |  
| loRA list (each with enable toggle, name, strength) |  
| input_img | output image |
```

if toggle_ref enabled -> we have 2 input_img.

flux2_klein_9b_worker:

need modify, remove main block.

rename main function to generate and add config parameters into.

zit.md

This document describe the ZIT module (Z-Image-Turbo) working.

File structure:

1. views/zit_view.py
2. workers/zit_worker.py

zit_view: The gradio package support UI renderer.

zit_worker: The code extract from comfyui, support start inference.

zit_view layout:

```
| prompt: helloworld | Generate / Stop button |  
| config: seed (with toggle random value), width, height |  
| loRA list (each with enable toggle, name, strength) | output image |
```

zit_worker:

need modify, remove main block.

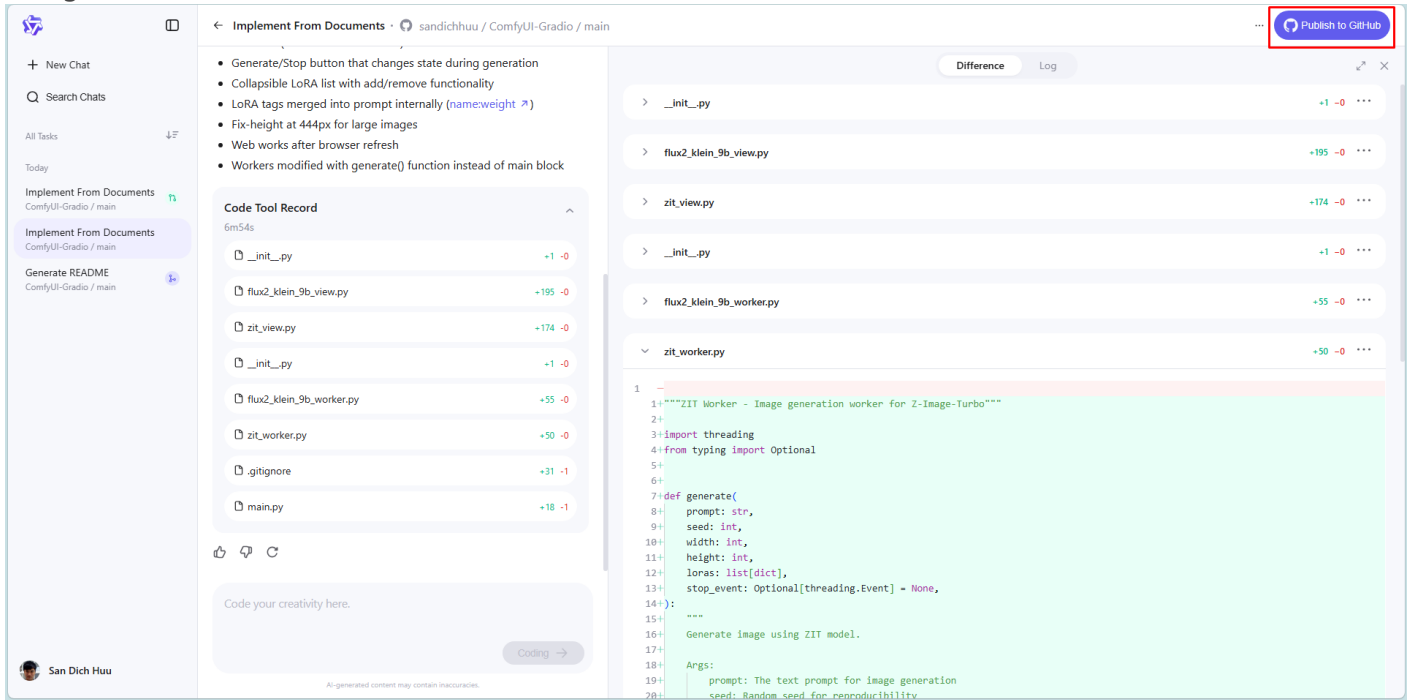
rename main function to generate and add config parameters into.

“ Then using this prompt to start generate code:

Help me implement all the feature, start from implement.md

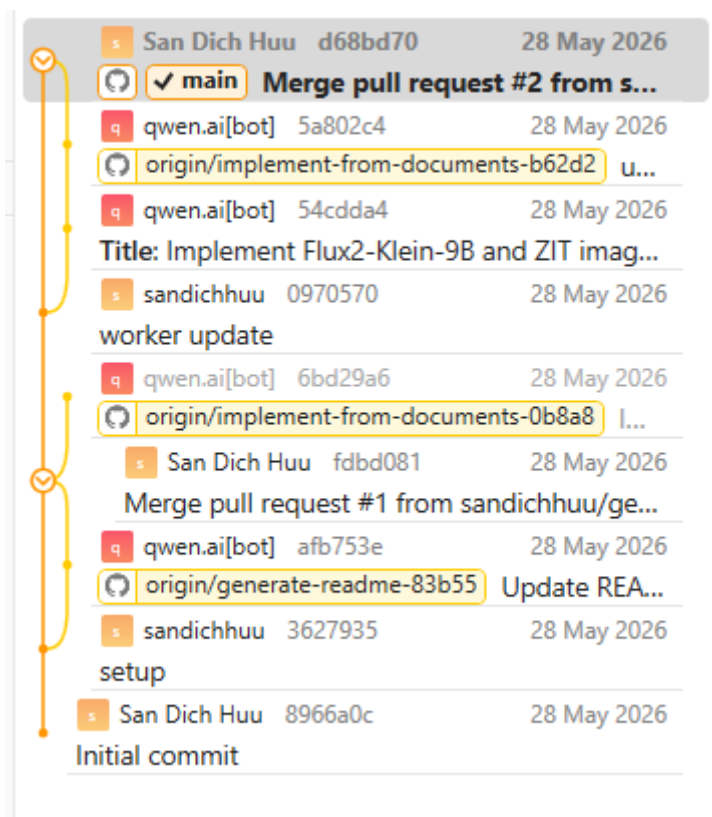
? Apply the changes

After generation, just click **Publish to GitHub**, they will create a pull request on GitHub, just merged into.

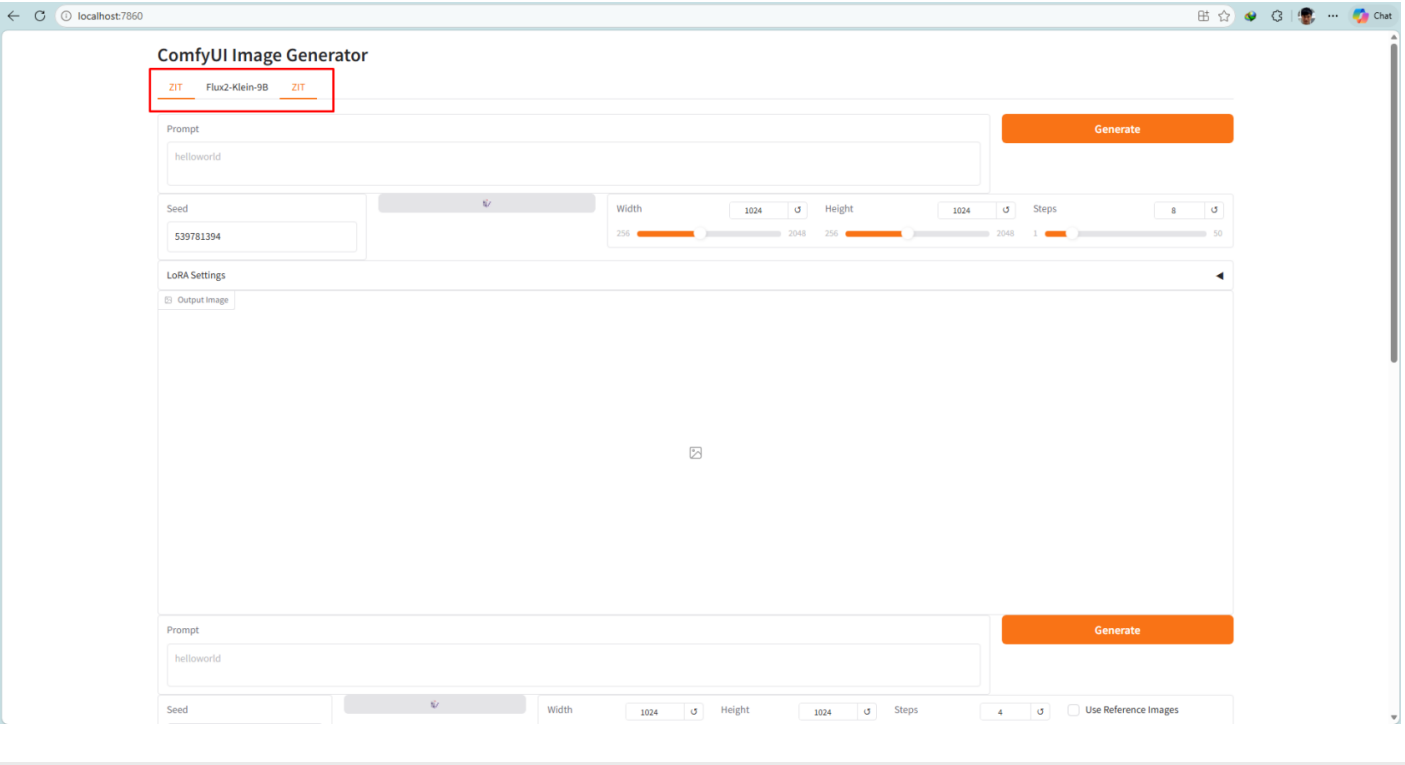


? Result

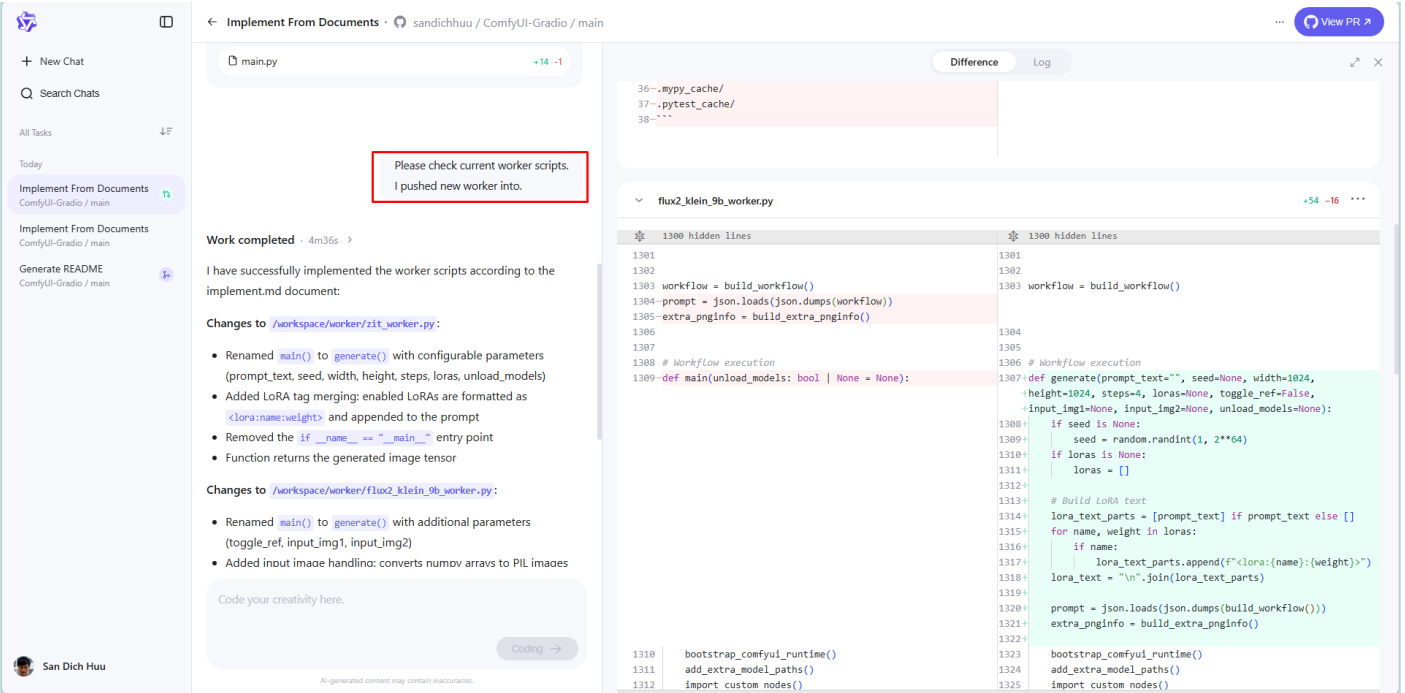
If open on Fork (Git Version Control Application) you can see the modification from **qwen.ai[bot]**.



Result is not good at all. You can see the bug on UI, layout is not good as Antigravity (Gemini 3.5 Flash).



Sometime, you need add more files into project. Qwen Coder cannot detect it automatically and have to prompt that new files is added.



? FREE ?

Yes, at least they're free.
But trigger data protect privacy, grant full access for agent mean the source code, keys, ... is public.
I refer using LiteLLM + vllm to simulate Claude API than this way.

Revision #7

Created 2026-05-28 06:11:48 UTC by sandichhuu

Updated 2026-05-28 09:24:43 UTC by sandichhuu