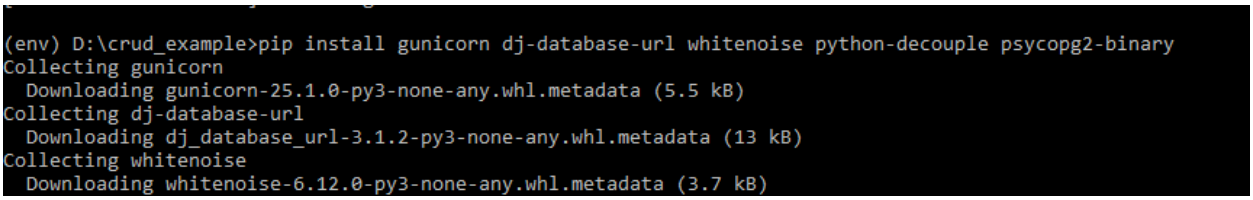
Perfect! Let's go **step by step with a full, detailed guide to deploy a Django project for free on Render**, which is one of the easiest and most reliable free hosts right now. I'll include **all commands, settings, and explanations**, so you can literally deploy in under 15 minutes.

Step 1: Prepare Your Django Project for Deployment

1.1 Install Required Packages

Open your terminal in your project folder and run:

```
pip install gunicorn dj-database-url whitenoise python-decouple psycopg2-binary
```



What these do:

- **gunicorn** → Production server (instead of `manage.py runserver`)
- **dj-database-url** → Easily read database URL (like PostgreSQL)
- **whitenoise** → Serve static files (CSS/JS) in production
- **python-decouple** → Store sensitive data (`SECRET_KEY`) in `.env`
- **psycopg2-binary** → PostgreSQL driver (Render gives free Postgres)

1.2 Create a `.env` File

In your project root:

```
SECRET_KEY='your-secret-key'
DEBUG=False
```

Make sure you **never commit** `.env` to GitHub. Add it to `.gitignore`.

1.3 Update `settings.py`

Add imports at the top

```
import os
from decouple import config
import dj_database_url
```

Set secret key and debug

```
SECRET_KEY = config('SECRET_KEY')
DEBUG = config('DEBUG', default=False, cast=bool)
```

Set allowed hosts

```
ALLOWED_HOSTS = ['your-app-name.onrender.com']  # Replace with your Render
URL
```

Database configuration

At the bottom of `settings.py`, replace default SQLite for production:

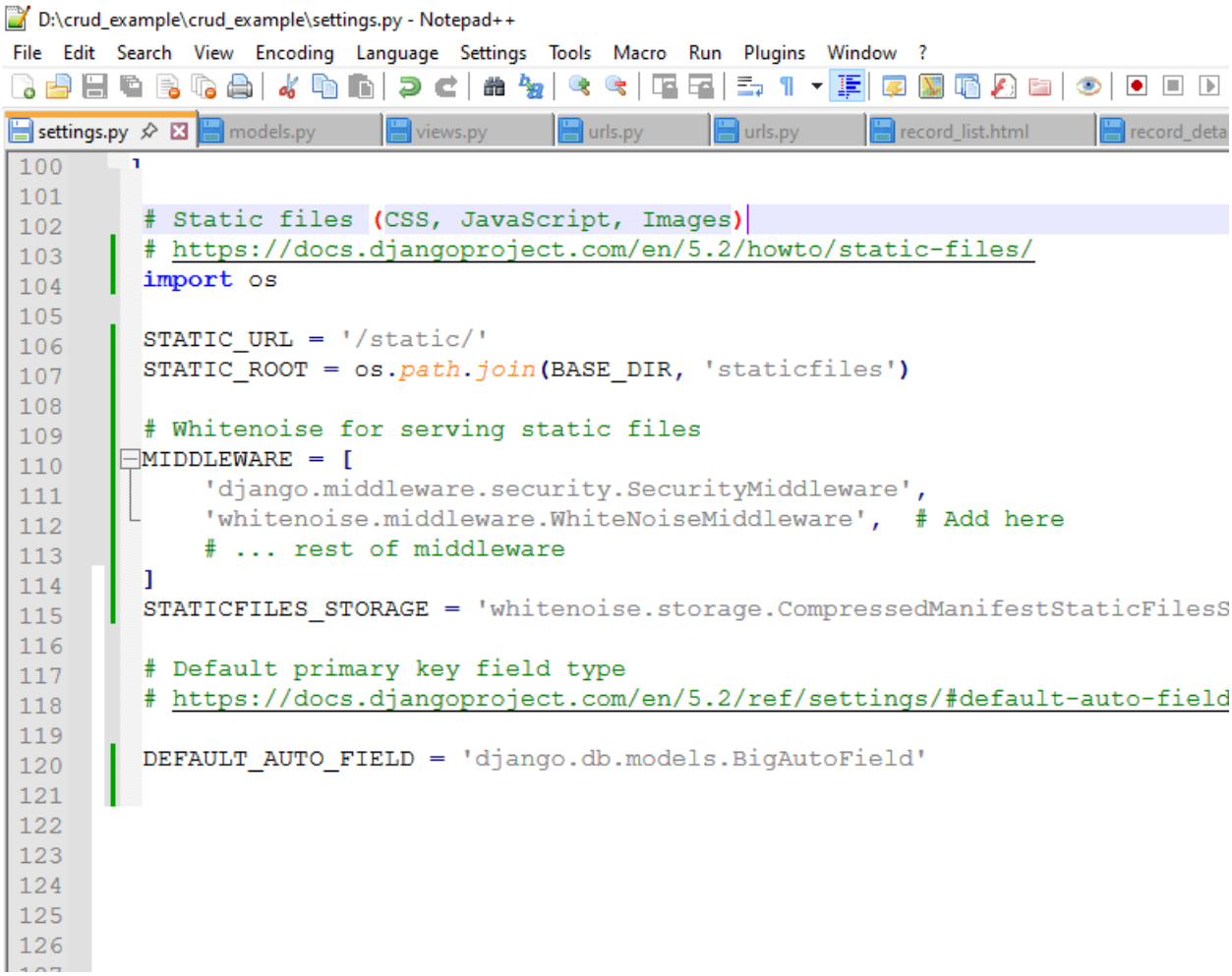
```
DATABASES = {
    'default': dj_database_url.config(
        default='sqlite:///db.sqlite3',  # fallback to SQLite
        conn_max_age=600
    )
}
```

1.4 Configure Static Files

Add to `settings.py`:

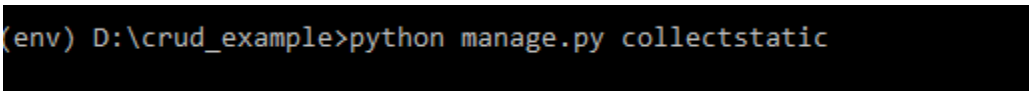
```
STATIC_URL = '/static/'
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')

# Whitenoise for serving static files
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'whitenoise.middleware.WhiteNoiseMiddleware',  # Add here
    # ... rest of middleware
]
STATICFILES_STORAGE =
'whitenoise.storage.CompressedManifestStaticFilesStorage'
```



Then, collect static files locally to check everything works:

```
python manage.py collectstatic
```



1.5 Create `requirements.txt`

Render uses this to install dependencies:

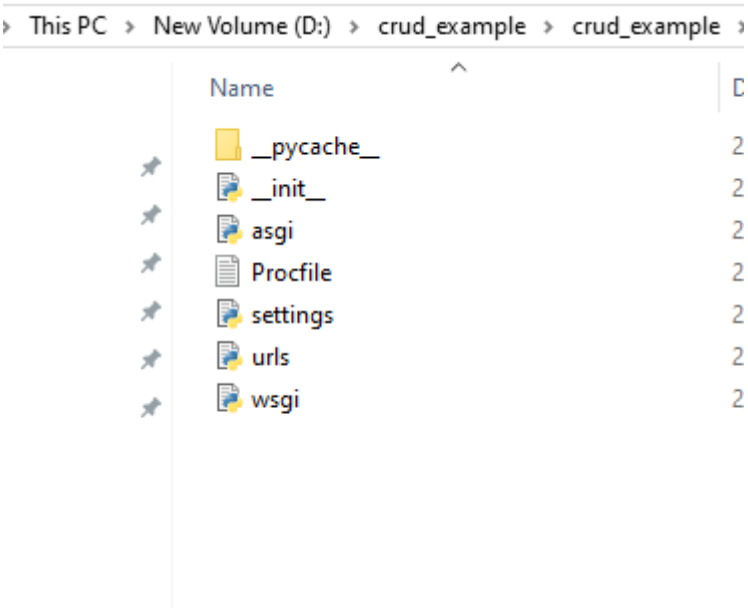
```
pip freeze > requirements.txt
```

1.6 Create `Procfile`

In your project root, create a file named `Procfile` (no extension) with:

```
web: gunicorn myproject.wsgi
```

Replace `myproject` with your Django project folder name (where `wsgi.py` is located).



Note:-

Embed migrations in your code (hacky, not recommended)

You could technically run migrations in `wsgi.py`:

```
import os
from django.core.wsgi import get_wsgi_application
from django.core.management import call_command

os.environ.setdefault("DJANGO_SETTINGS_MODULE", "crud_example.settings")

# Run migrations automatically (dangerous on each start!)
call_command('migrate', interactive=False)
application = get_wsgi_application()
```

- □ Pros: works on free Render

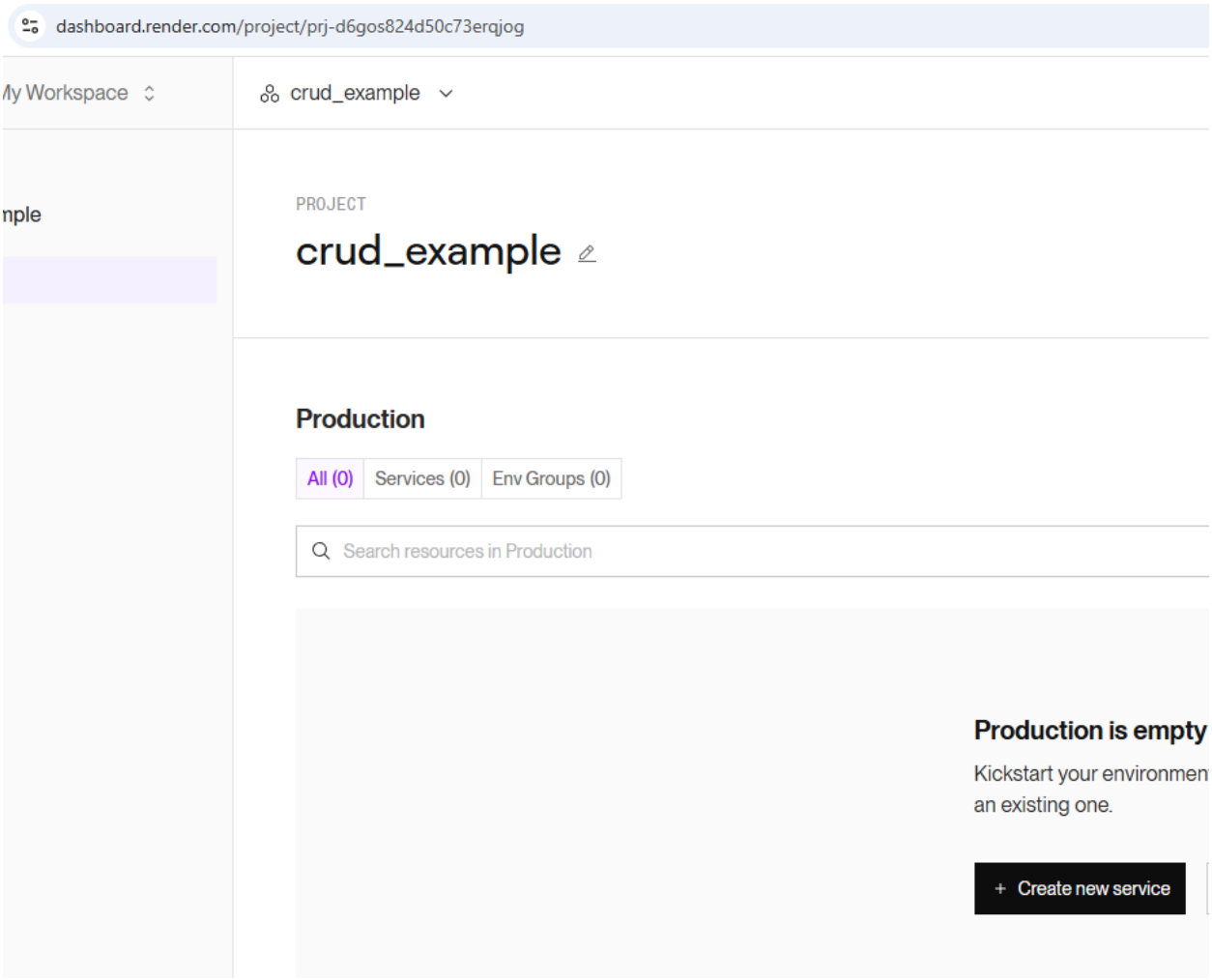
Step 2: Push to GitHub

Render connects to GitHub for deployment.

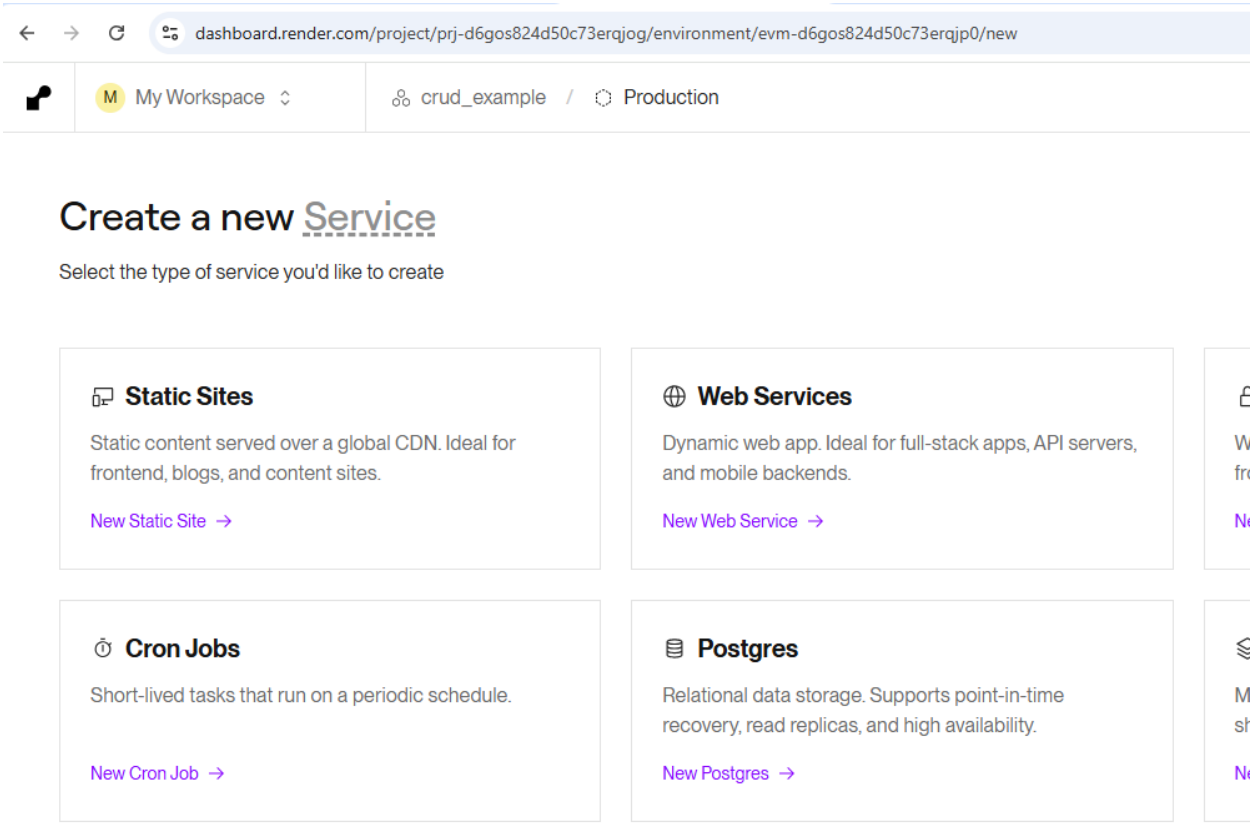
```
git init
git add .
git commit -m "Ready for Render deployment"
git branch -M main
git remote add origin https://github.com/username/repo.git
git push -u origin main
```

Step 3: Create a Free Render Account

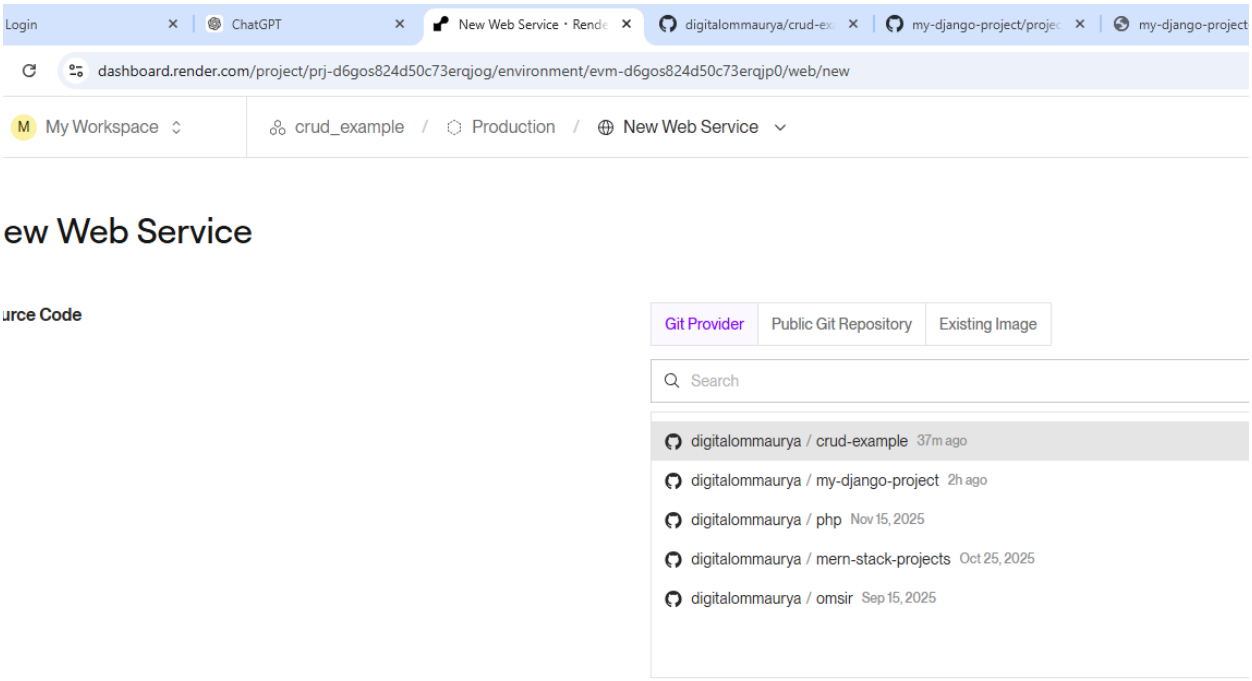
1. Go to Render.com → Sign up.
2. Connect your **GitHub account**.



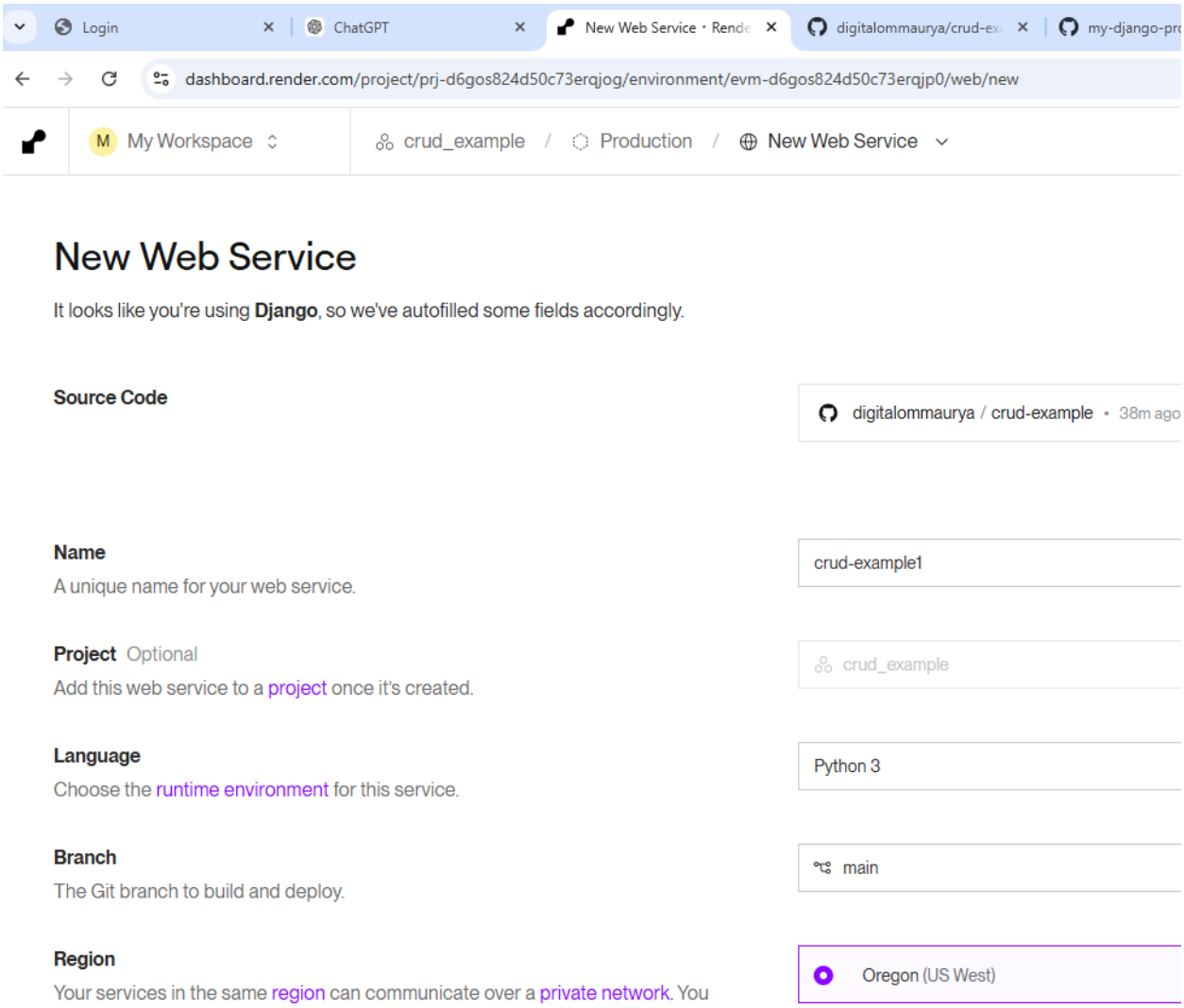
And click on new web services:-



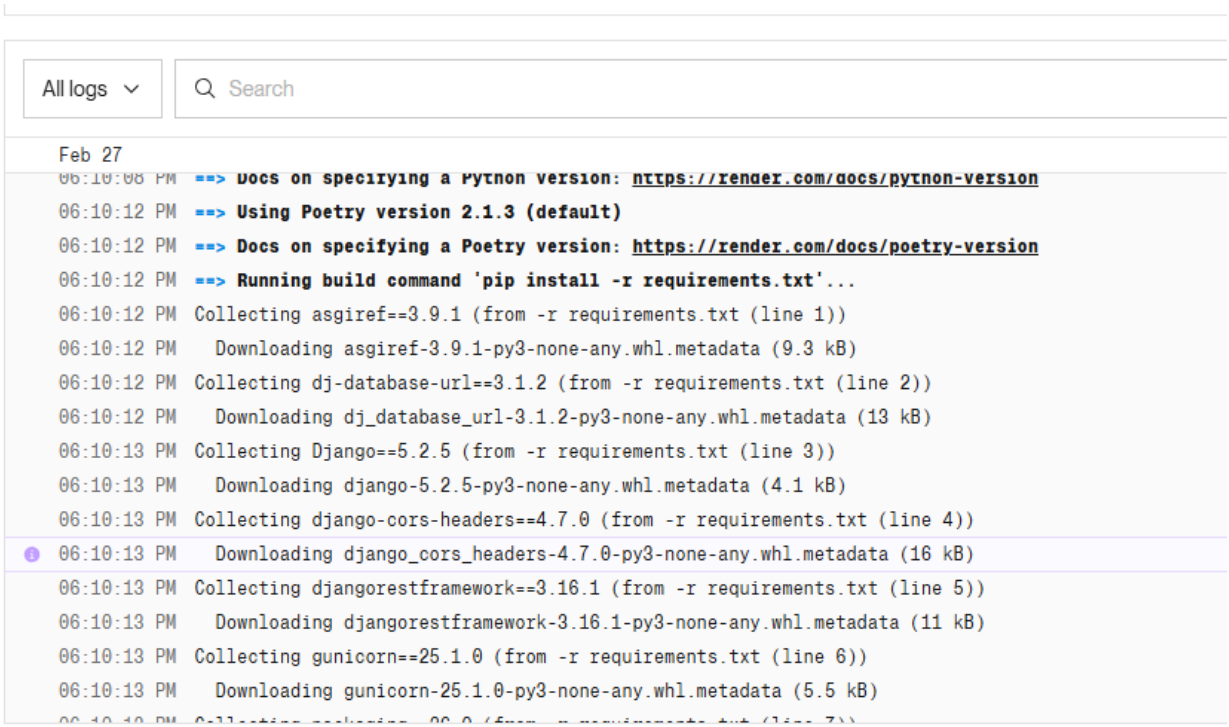
And then connect your github account with all repositories


Login
ChatGPT
New Web Service · Rende
digitalommaurya/crud-ex
my-django-project/projec
my-django-project
dashboard.render.com/project/prj-d6gos824d50c73erqjog/environment/evm-d6gos824d50c73erqjp0/web/new
My Workspace
crud_example
Production
New Web Service
ew Web Service
urce Code
Git Provider
Public Git Repository
Existing Image
Search
digitalommaurya / crud-example 37m ago
digitalommaurya / my-django-project 2h ago
digitalommaurya / php Nov 15, 2025
digitalommaurya / mern-stack-projects Oct 25, 2025
digitalommaurya / omsir Sep 15, 2025

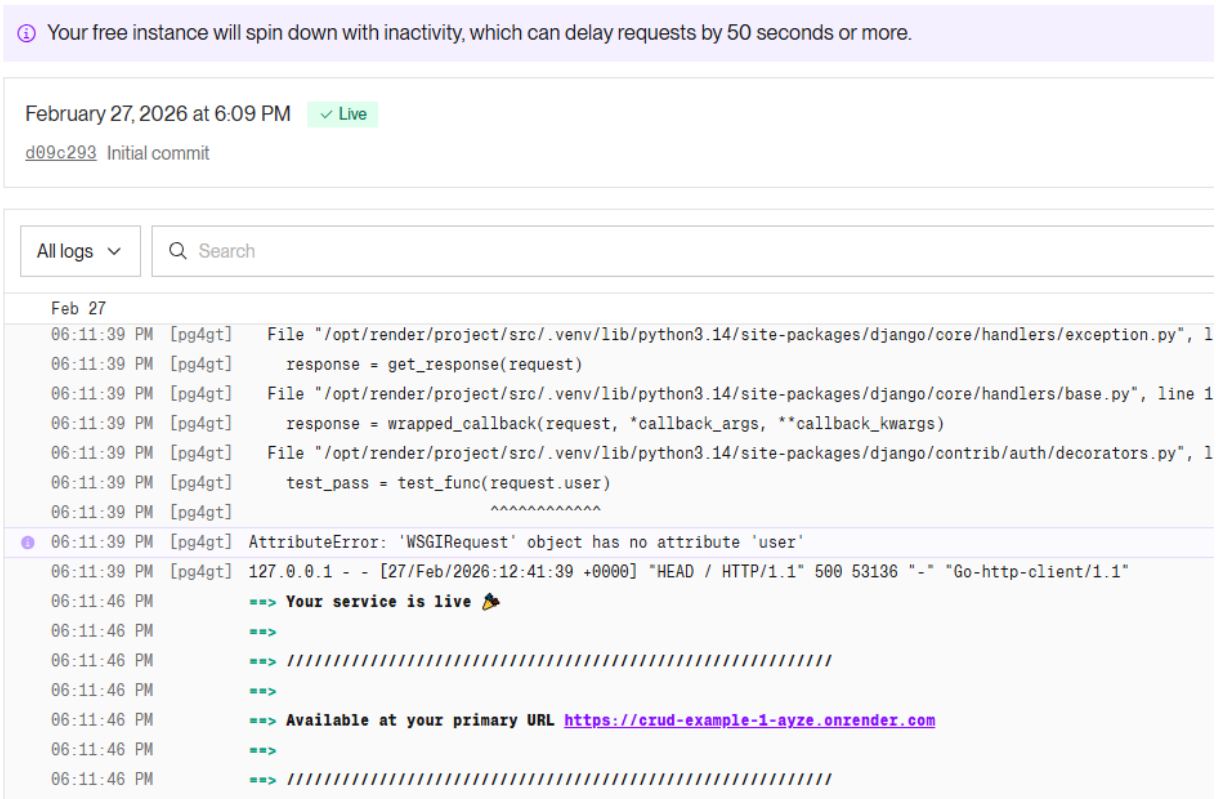
And then

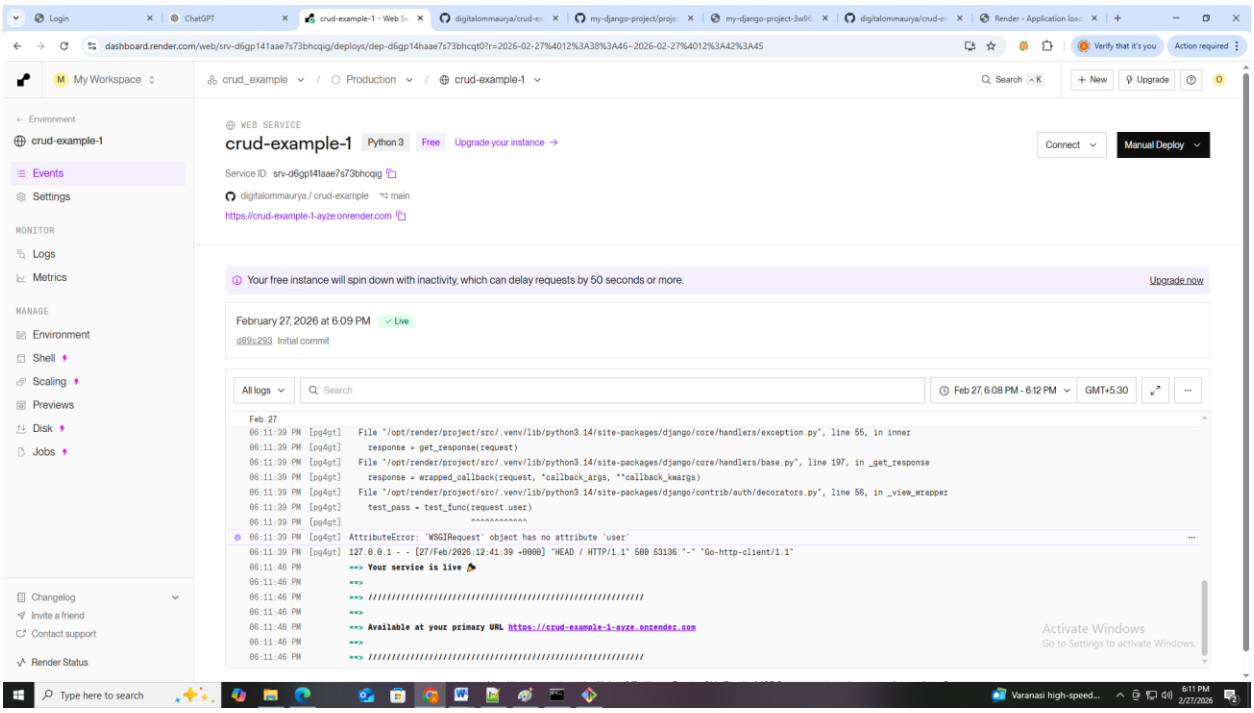


And you will see



And finally your service will be live :-




dashboard.render.com/web/srv-d6gp141aae7s73bhcqig/deploys/dep-d6gp14haae7s73bhcqt0?r=2026-02-27%4012%3A38%3A46~2026-02-27%4012%3A42%3A45
My Workspace
crud_example / Production / crud-example-1
Search
New
Upgrade
Environment
crud-example-1
Events
Settings
MONITOR
Logs
Metrics
MANAGE
Environment
Shell
Scaling
Previews
Disk
Jobs
Changelog
Invite a friend
Contact support
Render Status
WEB SERVICE
crud-example-1
Python 3
Free
Upgrade your instance →
Connect
Manual Deploy
Service ID: srv-d6gp14laae7s73bhcqig
digitalommaurya / crud-example
main
https://crud-example-1-ayze.onrender.com
Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more.
Upgrade now
February 27, 2026 at 6:09 PM
Live
d89c293 Initial commit
All logs
Search
Feb 27, 6:08 PM - 6:12 PM
GMT+5:30
Feb 27
06:11:39 PM [pg4gt] File "/opt/render/project/src/.venv/lib/python3.14/site-packages/django/core/handlers/exception.py", line 55, in inner
06:11:39 PM [pg4gt] response = get_response(request)
06:11:39 PM [pg4gt] File "/opt/render/project/src/.venv/lib/python3.14/site-packages/django/core/handlers/base.py", line 197, in _get_response
06:11:39 PM [pg4gt] response = wrapped_callback(request, *callback_args, **callback_kwargs)
06:11:39 PM [pg4gt] File "/opt/render/project/src/.venv/lib/python3.14/site-packages/django/contrib/auth/decorators.py", line 56, in _view_wrapper
06:11:39 PM [pg4gt] test_pass = test_func(request.user)
06:11:39 PM [pg4gt] ^^^^^^^^^^^^
06:11:39 PM [pg4gt] AttributeError: 'WSGIRequest' object has no attribute 'user'
06:11:39 PM [pg4gt] 127.0.0.1 - - [27/Feb/2026:12:41:39 +0000] "HEAD / HTTP/1.1" 500 53136 "-" "Go-http-client/1.1"
06:11:46 PM ==> Your service is live 🎉
06:11:46 PM ==>
06:11:46 PM ==> ///
06:11:46 PM ==>
06:11:46 PM ==> Available at your primary URL https://crud-example-1-ayze.onrender.com
06:11:46 PM ==>
06:11:46 PM ==> ///
Activate Windows
Go to Settings to activate Windows.
Type here to search
Varanasi high-speed...
6:11 PM
2/27/2026