

Panduan Menjalankan Aplikasi (Python Flask + PostgreSQL + Machine Learning + Ollama)

1 Clone Repository

Buka **Terminal / PowerShell**, lalu jalankan:

- git clone https://github.com/hialtta/machine_learning_project_2025.git
- cd machine_learning_project_2025

```
PS D:\Ara folder\BION\SKRIPSI\Project\test repository> git clone https://github.com/hialtta/machine_learning_project_2025.git
Cloning into 'machine_learning_project_2025'...
remote: Enumerating objects: 158, done.
remote: Counting objects: 100% (158/158), done.
remote: Compressing objects: 100% (139/139), done.
remote: Total 158 (delta 10), reused 158 (delta 10), pack-reused 0 (from 0)R
Receiving objects: 100% (158/158), 21.95 MiB | 359.00 KiB/s, done.
Resolving deltas: 100% (10/10), done.
PS D:\Ara folder\BION\SKRIPSI\Project\test repository> cd machine_learning_project_2025
PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025> |
```

Pastikan struktur proyek terlihat (contoh):

machine_learning_project_2025/

```
├── app/
├── config.py
├── requirements.txt
├── run.py / wsgi.py
└── README.md
```

 .vscode	1/17/2026 6:01 PM	File folder
 app	1/17/2026 6:01 PM	File folder
 .gitignore	1/17/2026 6:01 PM	Git Ignore Source File 1 KB
 config.py	1/17/2026 6:01 PM	Python Source File 1 KB
 README.md	1/17/2026 6:01 PM	Markdown Source File 1 KB
 requirements.txt	1/17/2026 6:01 PM	Text Document 1 KB
 run.py	1/17/2026 6:01 PM	Python Source File 1 KB

2 Membuat & Mengaktifkan Virtual Environment (venv)

Bukalah folder machine_learning_project_2025 pada Visual Studio Code IDE, lalu buka *New Terminal*.

Buat virtual environment:

- python -m venv venv
- venv\Scripts\activate

Jika berhasil, prompt akan berubah menjadi:

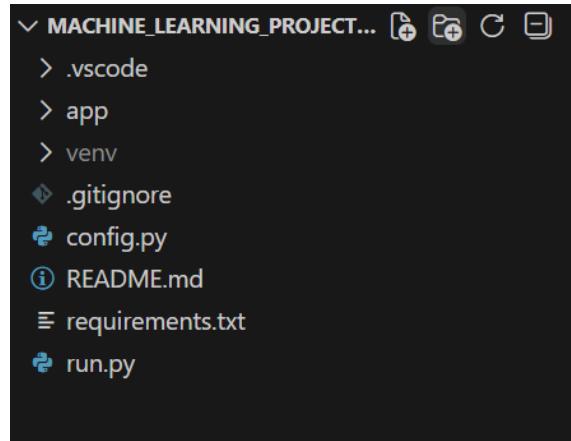
- (venv) PS D:\...

```
● PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025> python -m venv venv
● PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025> venv\Scripts\activate
○ (venv) PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025>
```

3 Install Dependency

Pastikan requirements.txt ada, lalu jalankan:

- pip install -r requirements.txt



```
○ (venv) PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025> pip install -r requirements.txt
Collecting blinker==1.9.0 (from -r requirements.txt (line 1))
  Using cached blinker-1.9.0-py3-none-any.whl.metadata (1.6 kB)
Collecting click==8.3.0 (from -r requirements.txt (line 2))
Successfully installed Flask-3.1.2 Jinja2-3.1.6 MarkupSafe-3.0.3 Pillow-12.1.0 Werkzeug-3.1.3 asn1crypto-1.5.1 blinker-1.9.0 certifi-2026.1.4 cffi-2.0.0 charset-normalizer-3.4.4 click-8.3.0 colorama-0.4.6 cryptography-46.0.3 flask-sqlalchemy-3.1.1 greenlet-3.3.0 idna-3.11 itsdangerous-2.2.0 joblib-1.5.3 nltk-3.9.2 numpy-2.4.1 pandas-2.3.3 pdfminer.six-20251230 pdfplumber-0.11.9 pg8000-1.31.5 pycparser-2.23 pypdfium2-5.3.0 python-dateutil-2.9.0.post0 python-dotenv-1.2.1 pytz-2025.2 regex-2026.1.15 requests-2.32.5 scikit-learn-1.6.1 scipy-1.17.0 scramble-1.4.8 six-1.17.0 sqlalchemy-2.0.45 threadpoolctl-3.6.0 tqdm-4.67.1 typing-extensions-4.15.0 tzdata-2025.3 urllib3-2.6.3 xgboost-3.1.3
[notice] A new release of pip is available: 24.0 -> 25.3
[notice] To update, run: python.exe -m pip install --upgrade pip
○ (venv) PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025>
```

4 Menyiapkan File .env

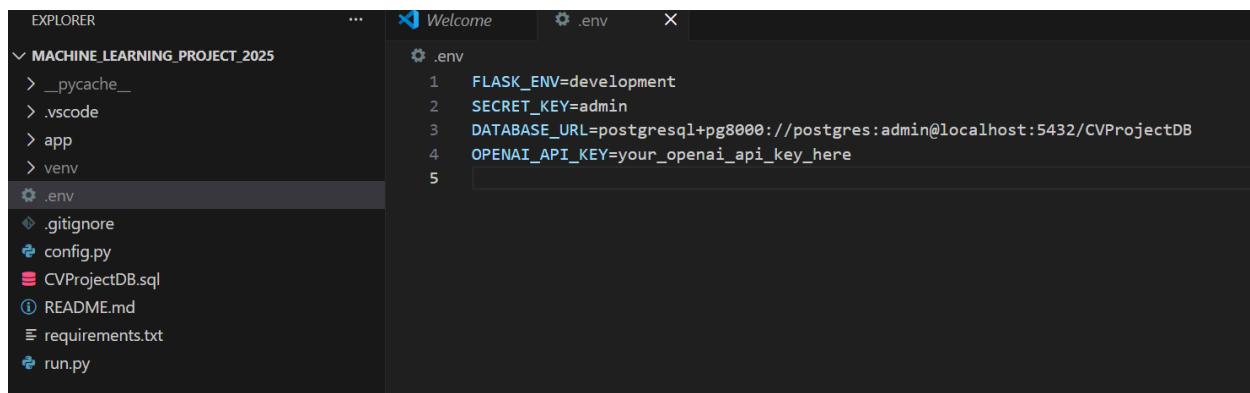
Buat file .env di **root project**, isinya:

FLASK_ENV=development

SECRET_KEY=admin

DATABASE_URL=postgresql+pg8000://postgres:admin@localhost:5432/CVProjectDB

OPENAI_API_KEY=your_openai_api_key_here



The screenshot shows the VS Code interface. On the left, the Explorer sidebar displays a project structure under 'MACHINE_LEARNING_PROJECT_2025'. It includes folders like '_pycache_', '.vscode', 'app', 'venv', and files like '.env', '.gitignore', 'config.py', 'CVProjectDB.sql', 'README.md', 'requirements.txt', and 'run.py'. The '.env' file is selected and shown in the main editor area. The content of the .env file is as follows:

```
1 FLASK_ENV=development
2 SECRET_KEY=admin
3 DATABASE_URL=postgresql+pg8000://postgres:admin@localhost:5432/CVProjectDB
4 OPENAI_API_KEY=your_openai_api_key_here
5
```

⚠ Catatan penting

- Jangan pernah commit .env
- .env harus terdaftar di .gitignore

5 Menjalankan Database (PostgreSQL)

Pastikan:

- PostgreSQL **sudah aktif**
- **Import** database CVProjectDB dengan menggunakan **pgAdmin 4**.
- Database CVProjectDB.sql didapatkan setelah clone repository machine_learning_project_2025, berada di direktori /.

config.py	1/17/2026 6:01 PM	Python Source File	1 KB
CVProjectDB.sql	1/17/2026 6:19 PM	SQL Source File	71,310 KB

6 Menjalankan Server Ollama (LLM Lokal)

Install Ollama

Download dari:

<https://ollama.com/download>

Jalankan Ollama Server

Buka **terminal baru** (jangan ditutup), lalu:

➤ ollama serve

Jika berhasil, akan muncul log seperti:

➤ Listening on <http://localhost:11434>

Pull model

➤ ollama pull llama3

run model

➤ ollama run llama3

```
C:\Users\hialt>ollama serve
Error: listen tcp 127.0.0.1:11434: bind: Only one usage of each socket address (protocol/network address/port) is normally permitted.

C:\Users\hialt>ollama pull llama3
pulling manifest
pulling 6a0746a1ec1a: 100% 4.7 GB
pulling 4fa551d4f938: 100% 12 KB
pulling 8ab4849b038c: 100% 254 B
pulling 577073ffcc6c: 100% 110 B
pulling 3f8eb4da87fa: 100% 485 B
verifying sha256 digest
writing manifest
success

C:\Users\hialt>
```

```
C:\Users\hialt>ollama run llama3
>>> |Send a message (/? for help)
```

➔ Ollama HARUS AKTIF sebelum Flask dijalankan jika aplikasi memanggil LLM.

7 Menjalankan Aplikasi Flask

Opsi A — menggunakan flask run

- flask run

Jika berhasil, akan muncul:

- Running on <http://127.0.0.1:5000>

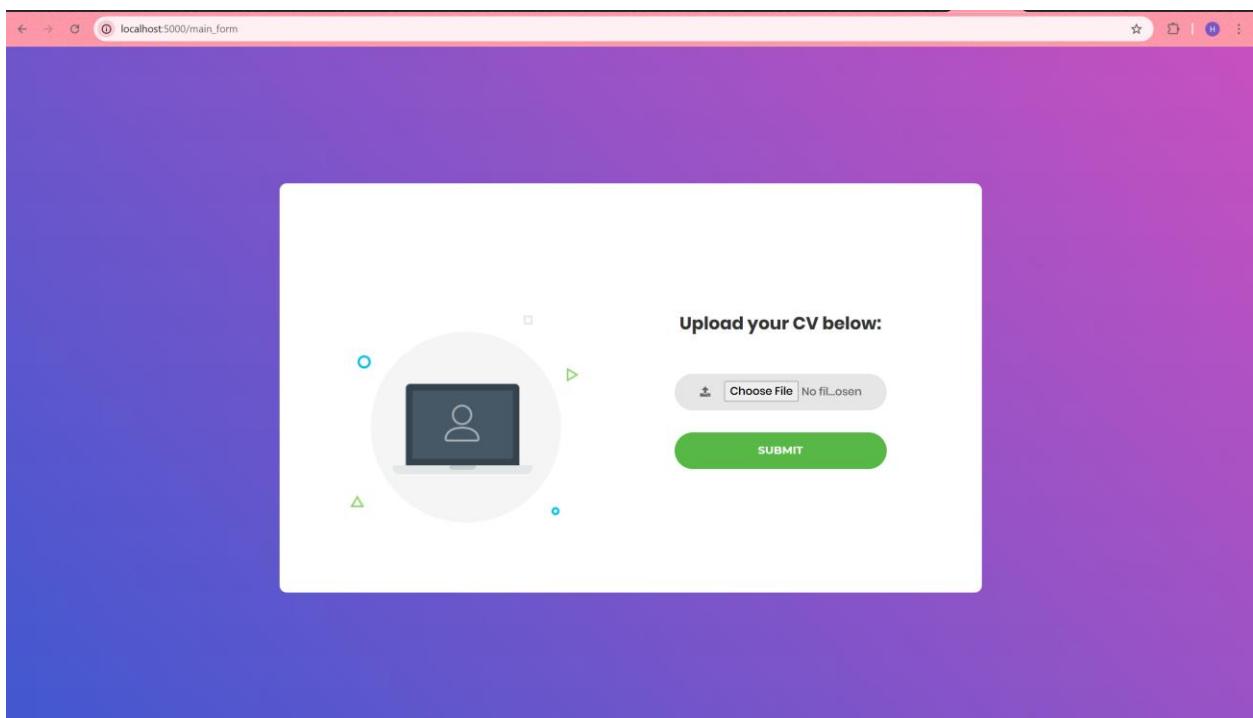
```
(venv) PS D:\Ara folder\BION\SKRIPSI\Project\test repository\machine_learning_project_2025> flask run
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

8 Membuka Aplikasi di Browser

Buka browser (Chrome/Edge/Firefox), lalu akses:

<http://127.0.0.1:5000>

<http://localhost:5000>



🎉 Aplikasi berhasil dijalankan