Report: Creating a Secure File Sharing Web Portal with AES Encryption on Windows

Author: *Bame Duncan Koko*

Date: *15 July 2025*

Technology Stack: Python 3, Flask, PyCharm Community Edition, Windows CMD

Objective

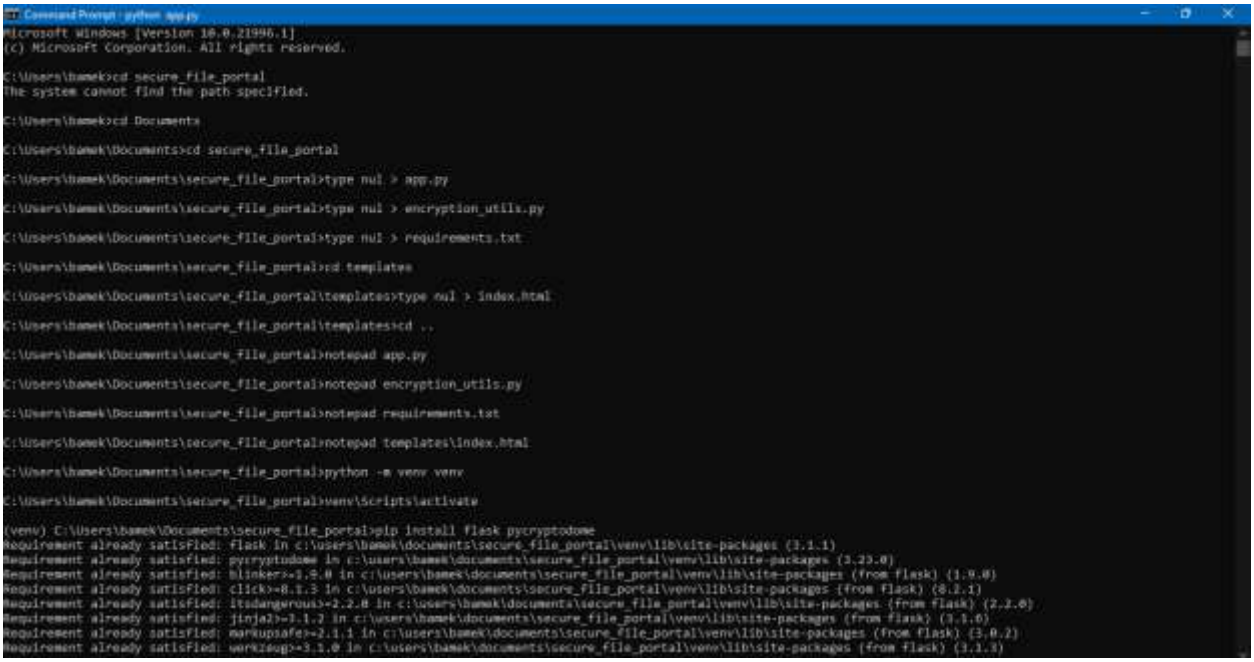
To develop and document a secure file-sharing web portal that allows users to upload and download files with AES encryption implemented before storage and decryption on retrieval.

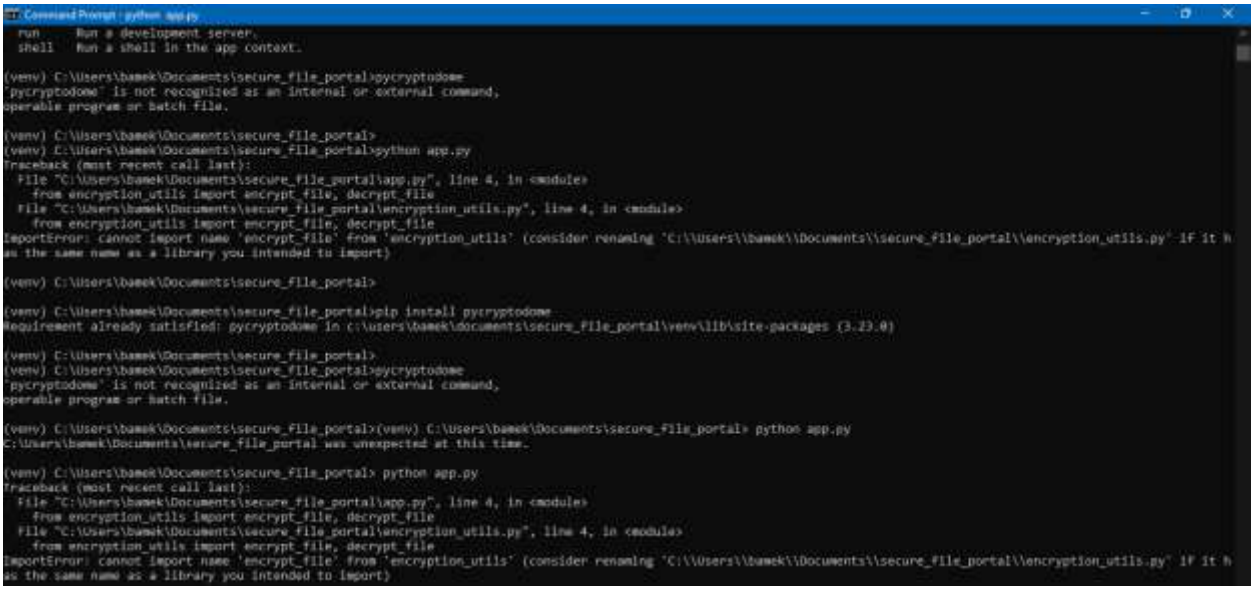
Tools and Technologies

- **Python 3.x** – Programming language
- **Flask** – Web framework
- **PyCryptodome** – AES encryption
- **Windows Command Prompt** – For project setup
- **PyCharm Community Edition** – For editing, running, and managing the project
- **Notepad (optional)** – For quick edits

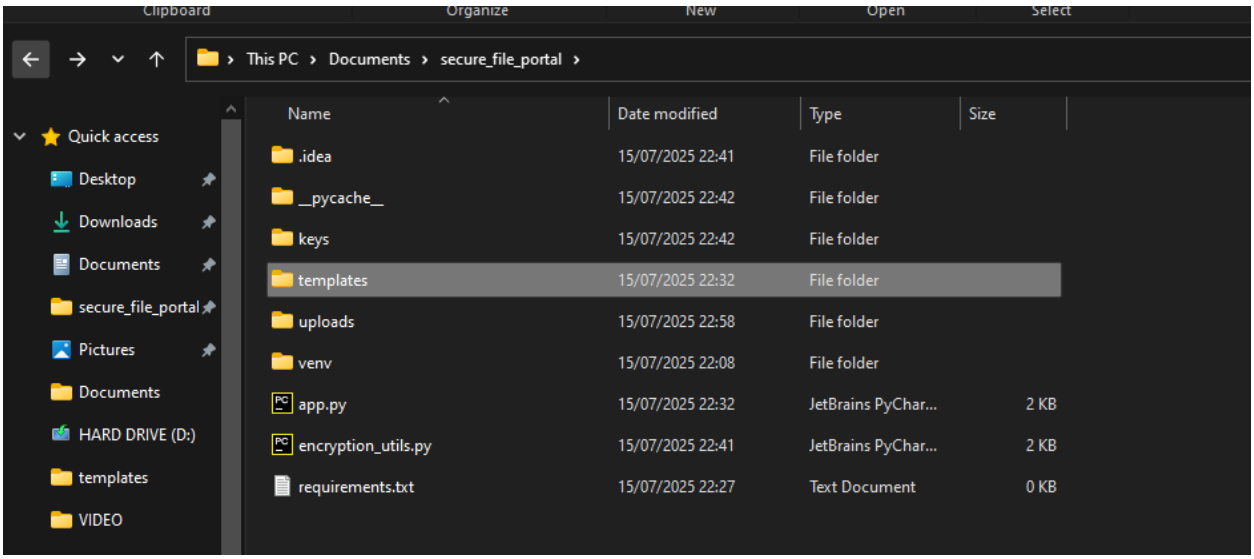
Steps Taken

I installed Python 3 and added it to PATH, then installed PyCharm Community Edition for development. Using Command Prompt, I navigated to C:\Users\bamek\Documents, created a project folder named secure_file_portal, and entered it. I created and activated a virtual environment with python -m venv venv followed by venv\Scripts\activate. I then created a requirements.txt file containing flask and pycryptodome and installed these dependencies using pip install flask pycryptodome. Next, I built the project structure by creating the templates, uploads, and keys folders, and creating empty files app.py, encryption_utils.py, and requirements.txt, then within templates I created index.html. The final structure was:
C:\Users\bamek\Documents\secure_file_portal\ containing app.py, encryption_utils.py, requirements.txt, a templates folder with index.html, and uploads and keys directories.



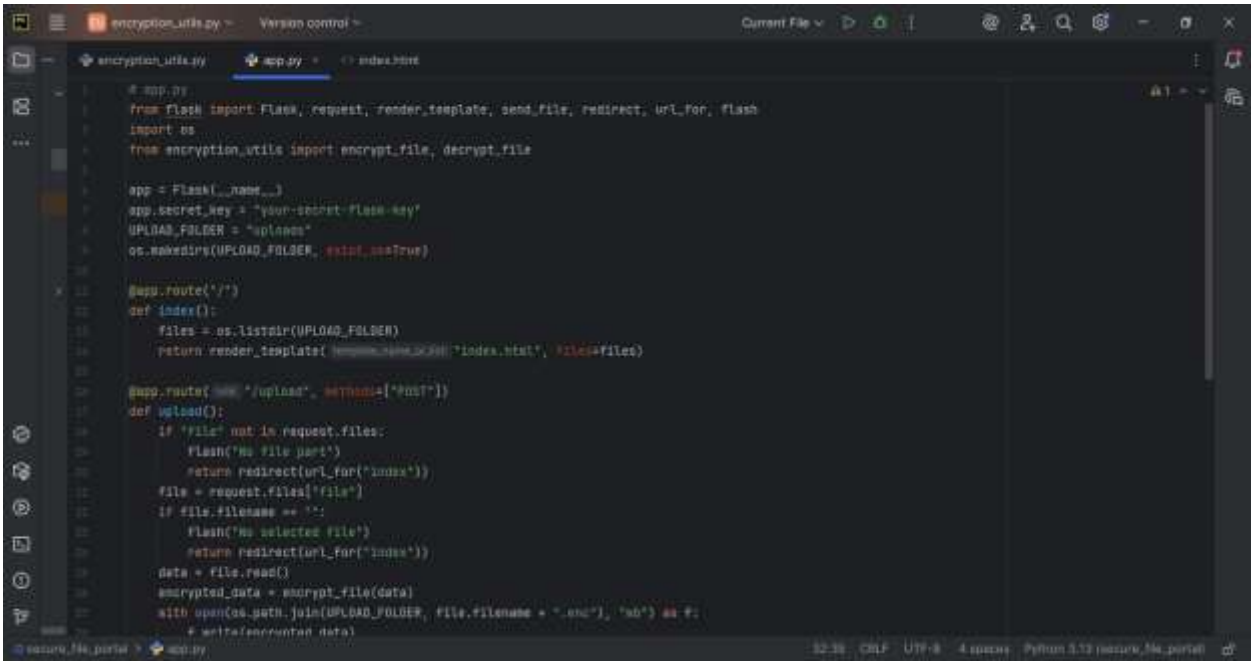

run Run a development server.
shell Run a shell in the app context.
(venv) C:\Users\bamek\Documents\secure_file_portal>pycryptodome
'pycryptodome' is not recognized as an internal or external command,
operable program or batch file.
(venv) C:\Users\bamek\Documents\secure_file_portal>
(venv) C:\Users\bamek\Documents\secure_file_portal>python app.py
Traceback (most recent call last):
File "C:\Users\bamek\Documents\secure_file_portal\app.py", line 4, in <module>
from encryption_utils import encrypt_file, decrypt_file
File "C:\Users\bamek\Documents\secure_file_portal\encryption_utils.py", line 4, in <module>
from encryption_utils import encrypt_file, decrypt_file
ImportError: cannot import name 'encrypt_file' from 'encryption_utils' (consider renaming 'C:\\Users\\bamek\\Documents\\secure_file_portal\\encryption_utils.py' if it h
as the same name as a library you intended to import)
(venv) C:\Users\bamek\Documents\secure_file_portal>
(venv) C:\Users\bamek\Documents\secure_file_portal>pip install pycryptodome
Requirement already satisfied: pycryptodome in c:\users\bamek\documents\secure_file_portal\venv\lib\site-packages (3.23.0)
(venv) C:\Users\bamek\Documents\secure_file_portal>
(venv) C:\Users\bamek\Documents\secure_file_portal>pycryptodome
'pycryptodome' is not recognized as an internal or external command,
operable program or batch file.
(venv) C:\Users\bamek\Documents\secure_file_portal>(venv) C:\Users\bamek\Documents\secure_file_portal> python app.py
C:\Users\bamek\Documents\secure_file_portal was unexpected at this time.
(venv) C:\Users\bamek\Documents\secure_file_portal> python app.py
Traceback (most recent call last):
File "C:\Users\bamek\Documents\secure_file_portal\app.py", line 4, in <module>
from encryption_utils import encrypt_file, decrypt_file
File "C:\Users\bamek\Documents\secure_file_portal\encryption_utils.py", line 4, in <module>
from encryption_utils import encrypt_file, decrypt_file
ImportError: cannot import name 'encrypt_file' from 'encryption_utils' (consider renaming 'C:\\Users\\bamek\\Documents\\secure_file_portal\\encryption_utils.py' if it h
as the same name as a library you intended to import)

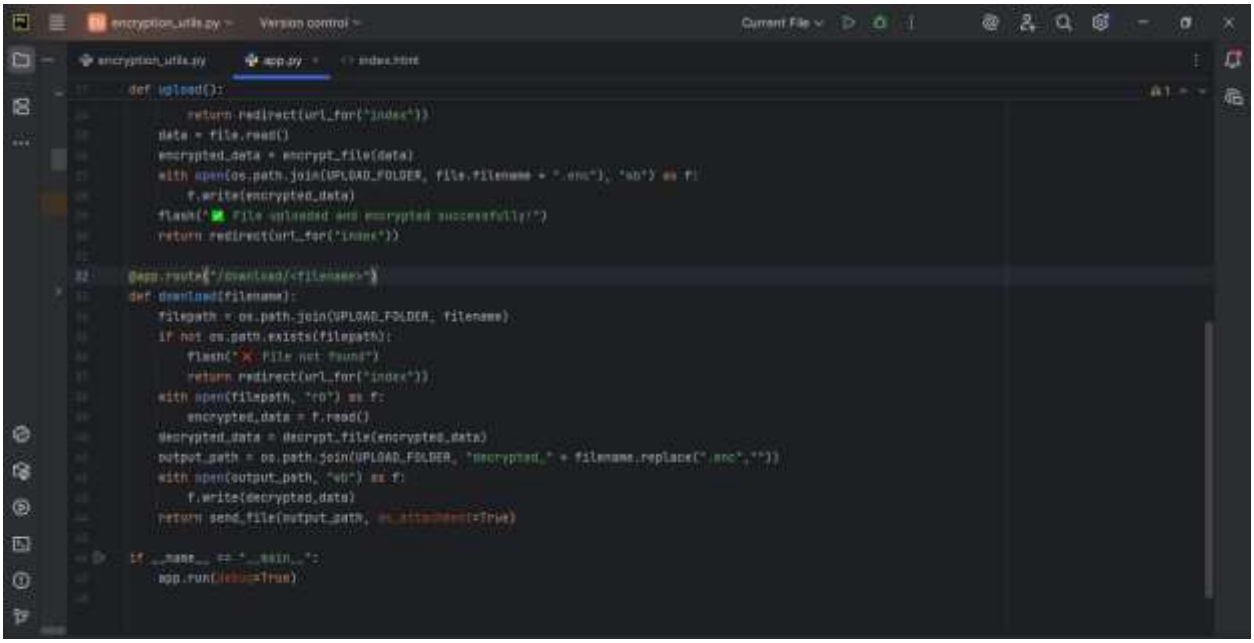

(venv) C:\Users\bamek\Documents\secure_file_portal>python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 283-408-297
127.0.0.1 - - [15/Jul/2025 22:43:25] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [15/Jul/2025 22:43:25] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [15/Jul/2025 22:43:36] "POST /upload HTTP/1.1" 302 -
127.0.0.1 - - [15/Jul/2025 22:43:36] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [15/Jul/2025 22:58:18] "GET /download/Database!!.png.enc HTTP/1.1" 200 -


Clipboard
Organize
New
Open
Select
This PC › Documents › secure_file_portal ›
Quick access
Desktop
Downloads
Documents
secure_file_portal
Pictures
Documents
HARD DRIVE (D:)
templates
VIDEO
Name
Date modified
Type
Size
.idea 15/07/2025 22:41 File folder
__pycache__ 15/07/2025 22:42 File folder
keys 15/07/2025 22:42 File folder
templates 15/07/2025 22:32 File folder
uploads 15/07/2025 22:58 File folder
venv 15/07/2025 22:08 File folder
app.py 15/07/2025 22:32 JetBrains PyChar... 2 KB
encryption_utils.py 15/07/2025 22:41 JetBrains PyChar... 2 KB
requirements.txt 15/07/2025 22:27 Text Document 0 KB

Coding in PyCharm

1. Opened the folder in **PyCharm Community Edition**:
   - File → Open → secure_file_portal
2. Edited app.py, encryption_utils.py, and templates/index.html with the provided code.
3. PyCharm's code highlighting helped ensure syntax correctness.

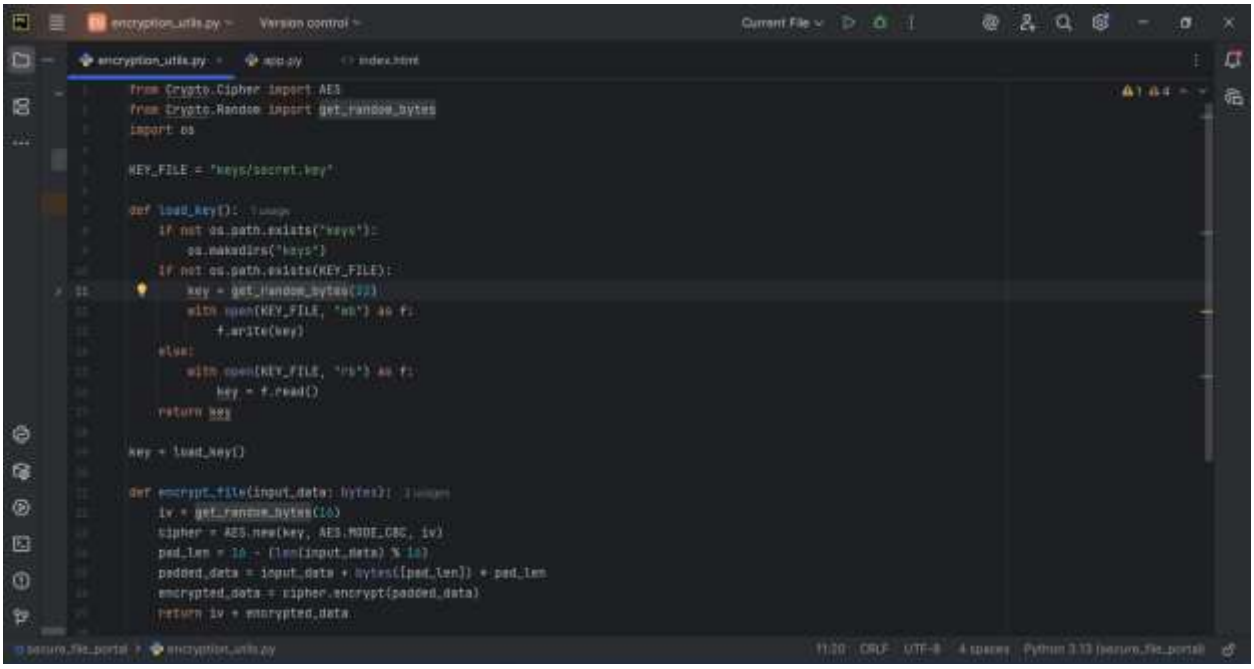


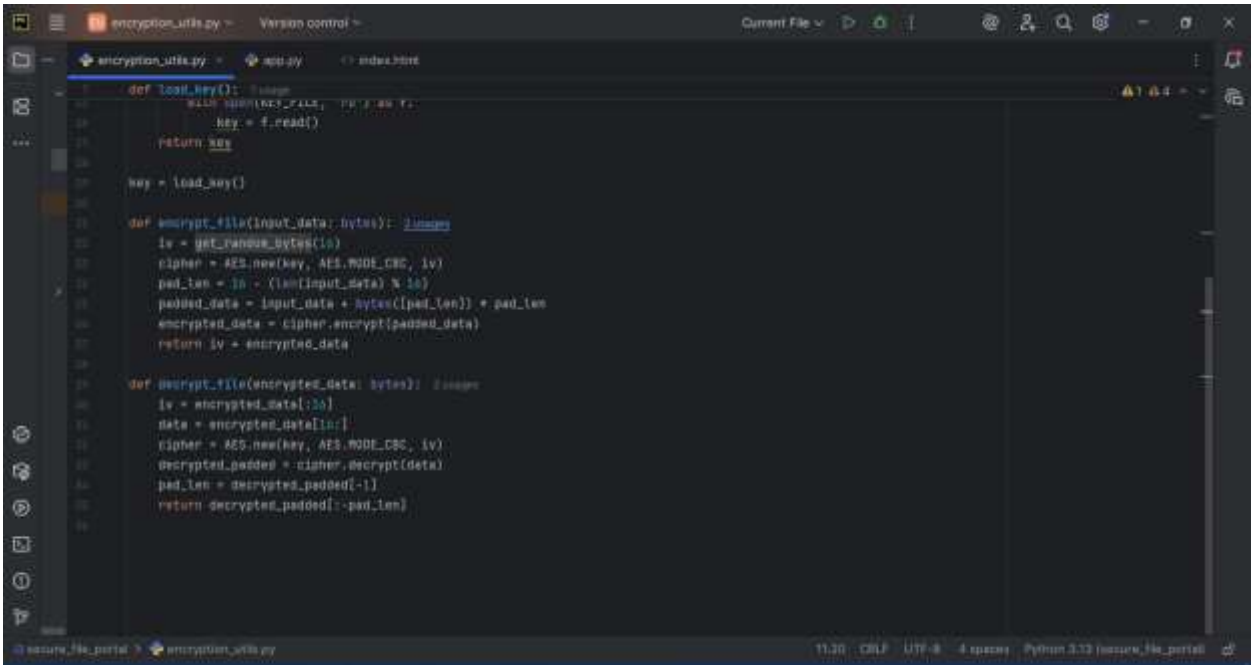


Implementing AES Encryption

In encryption_utils.py:

- Loaded/stored 256-bit key in keys/secret.key.
- Created encrypt_file() and decrypt_file() functions using AES-256 CBC.

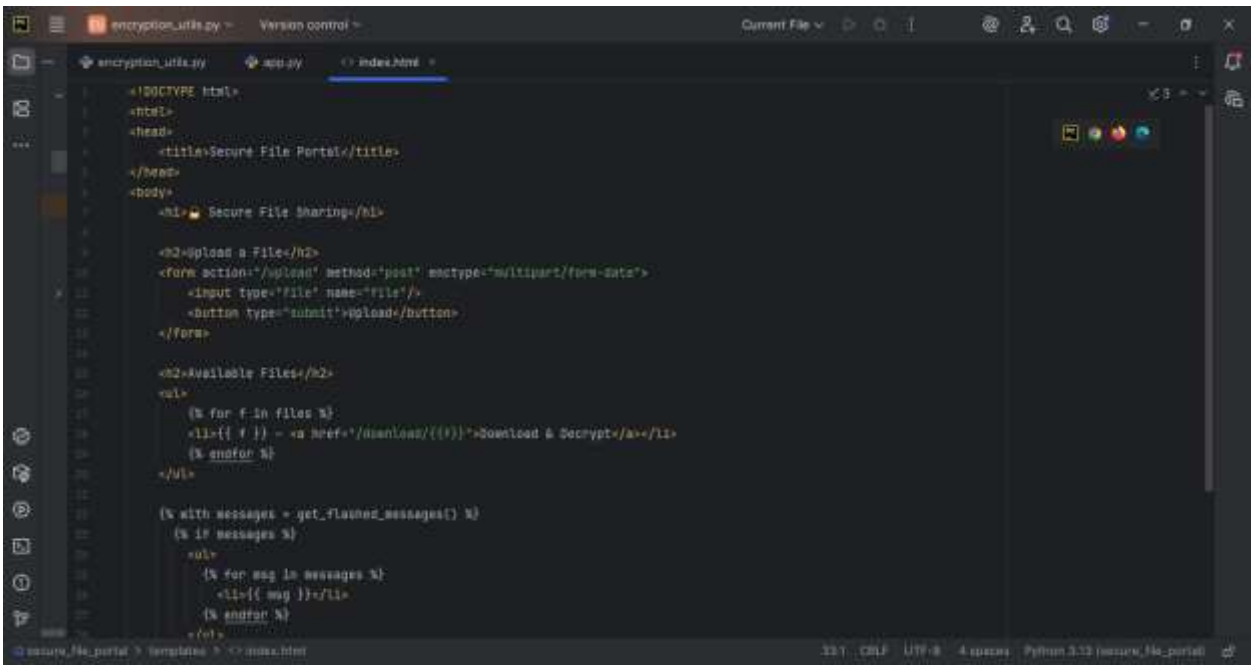




User Interface

Created a minimal HTML page in templates/index.html with:

- File upload form
- Links to download encrypted files



▶ Running the Application

In PyCharm:

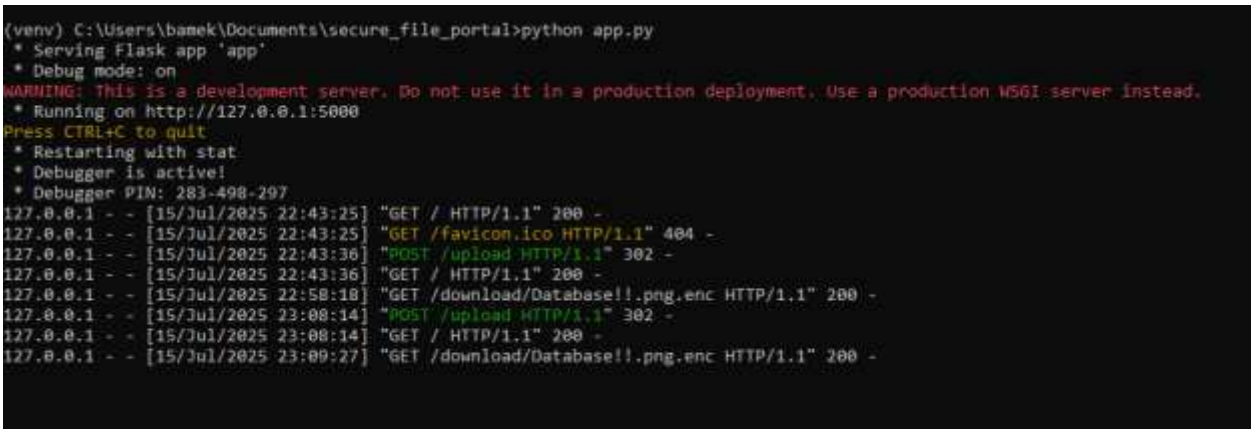
- Right-clicked app.py → **Run 'app'**
  or via Command Prompt:

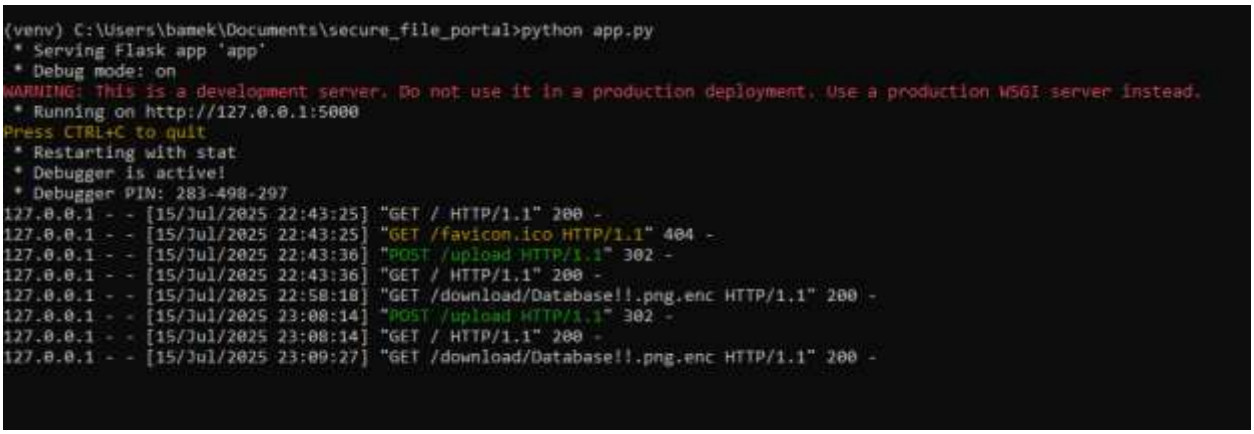
cmd
Copy code
python app.py

Output:

csharp
Copy code
 * Running on http://127.0.0.1:5000/

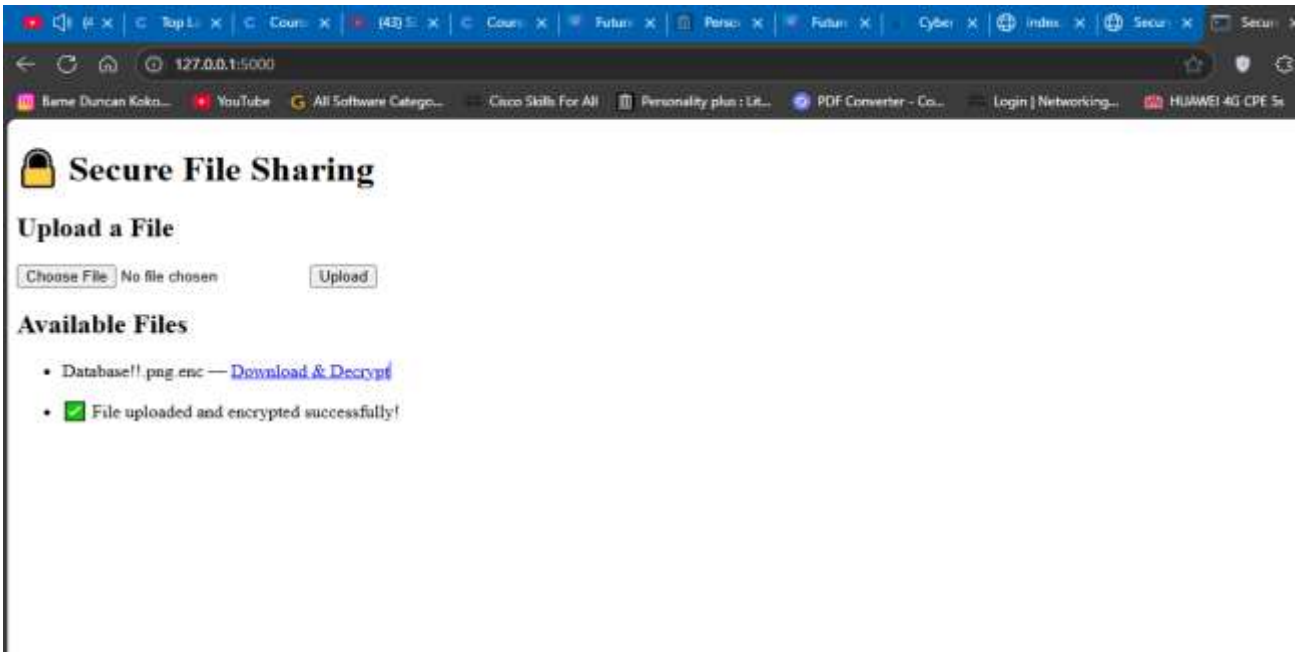
Accessed in browser: http://127.0.0.1:5000

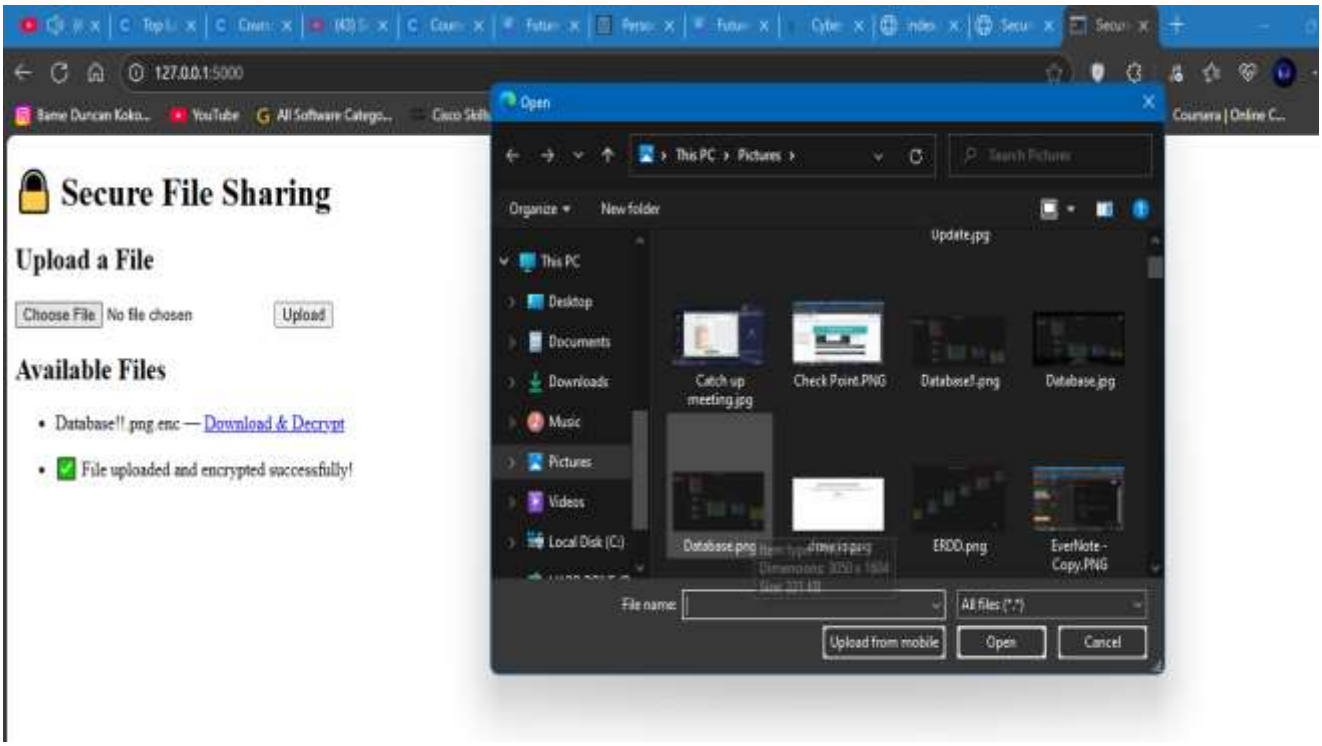


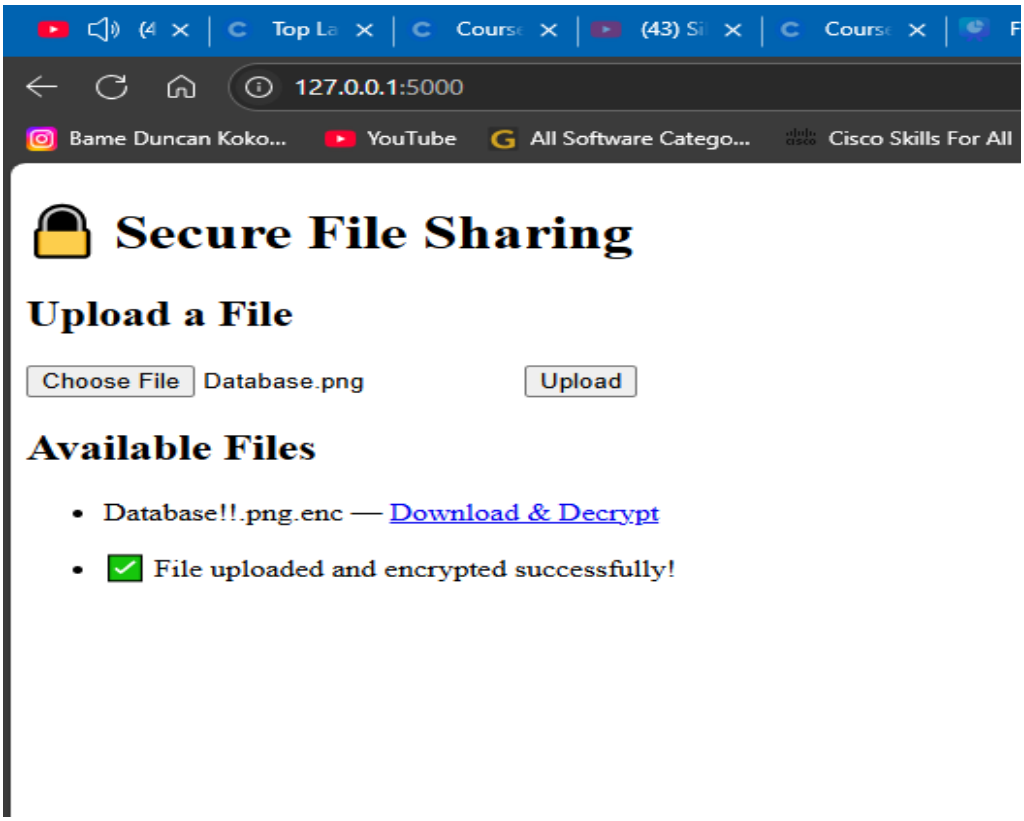


✓ Testing

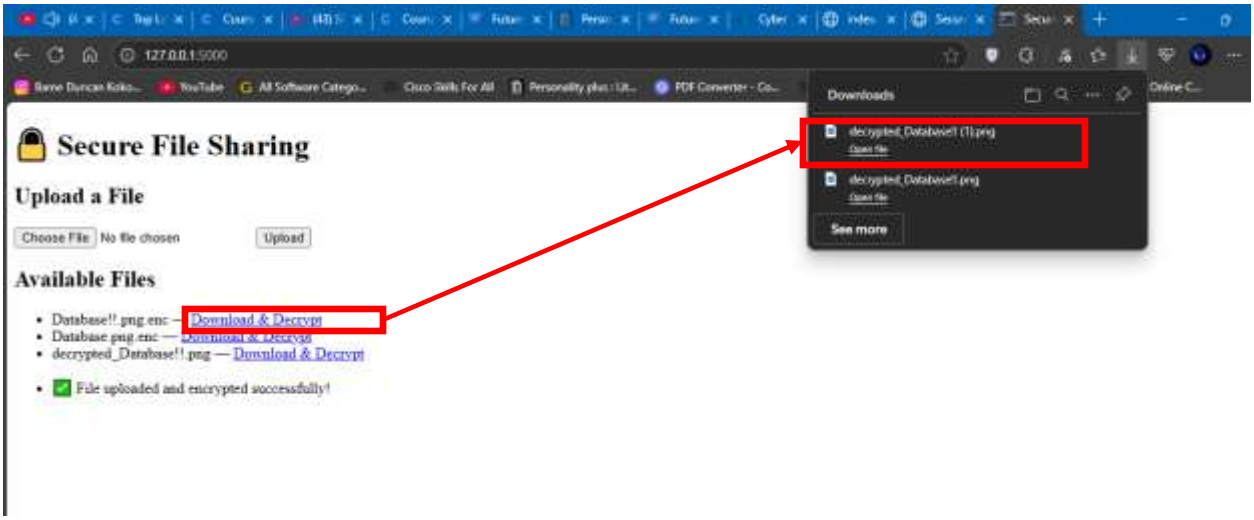
- Uploaded a sample file, confirmed .enc file created.
- Downloaded the file, verified it decrypted correctly.






127.0.0.1:5000
Bame Duncan Koko...
YouTube
All Software Catego...
Cisco Skills For All
Secure File Sharing
Upload a File
Choose File
Database.png
Upload
Available Files
Database!!.png.enc — Download & Decrypt
File uploaded and encrypted successfully!


Secure File Sharing
Upload a File
Choose File
No file chosen
Upload
Available Files
Database!!.png.enc — Download & Decrypt
Database.png.enc — Download & Decrypt
decrypted_Database!!.png — Download & Decrypt
File uploaded and encrypted successfully!



🔒 Security Measures

- AES-256 CBC encryption implemented.
- Key stored in keys/secret.key.
- Encrypted files stored in uploads.
- Flask secret_key set for sessions.
- Recommendation: use HTTPS and stronger key management in production.

🚀 Conclusion

The secure file sharing portal was successfully developed and run locally on Windows using **PyCharm Community Edition**.

✅ **Next Steps / Future Improvements:**

- Add authentication and access control.
- Deploy to a cloud server with HTTPS.
- Use a secrets manager for key storage.