**Bài tập**

**Chủ đề:** Xây dựng chương trình mã, giải mã file dữ liệu bằng sử dụng thuật toán AES

**Yêu cầu:**

+ phân tích ý tưởng

+ phân tích sơ đồ hoạt động

+ hoạt động web

+ có ô nhập mã khóa

+ có chức năng upload và download

1. **Phân tích ý tưởng**

Một chương trình với giao diện web đơn giản với chức năng mã hóa hoặc giải mã file sử dụng thuật toán AES Cypher. Sử dụng thư viện flask và pycryptodome để tạo môi trường hoạt động và giao diện cho web

1. **Phân tích sơ đồ hoạt động**

Người dùng truy cập - > index.html hoạt động hiển thị khóa, file và button -> nhập dữ liệu (khóa, file upload, action gồm encrypt hoặc decrypt) -> xử lý khóa nhập (đệm thành 16 bytes)

+ Encrypt:

* Tạo AES Cypher, IV mới với pad dữ liệu
* Mã hóa bằng AES CBC (cypher = IP + data)
* Gửi file cho người dùng

+ Decrypt:

* Tạo IV từ 16 bytes đầu
* Tạo AES cipher từ IV
* Giải mã và unpad dữ liệu
* Gửi file về client

1. **Hoạt động web**
2. <!DOCTYPE html>
3. <html lang="vi">
4. <head>
5. <meta charset="UTF-8">
6. <title>AES File Encrypt/Decrypt</title>
7. <style>
8. body {
9. font-family: 'Segoe UI', sans-serif;
10. background: #f0f2f5;
11. padding: 40px;
12. display: flex;
13. justify-content: center;
14. }
15. .container {
16. background: white;
17. padding: 30px 40px;
18. border-radius: 12px;
19. box-shadow: 0 4px 12px rgba(0,0,0,0.1);
20. width: 100%;
21. max-width: 500px;
22. }
23. h2 {
24. text-align: center;
25. color: #333;
26. margin-bottom: 20px;
27. }
28. label {
29. font-weight: 600;
30. display: block;
31. margin: 15px 0 5px;
32. }
33. input[type="text"],
34. input[type="file"] {
35. width: 100%;
36. padding: 10px;
37. border: 1px solid #ccc;
38. border-radius: 8px;
39. margin-bottom: 10px;
40. }
41. .buttons {
42. display: flex;
43. justify-content: space-between;
44. margin-top: 20px;
45. }
46. button {
47. flex: 1;
48. padding: 12px;
49. border: none;
50. border-radius: 8px;
51. font-size: 16px;
52. font-weight: bold;
53. color: white;
54. cursor: pointer;
55. transition: 0.3s;
56. }
57. button:first-child {
58. background-color: #4CAF50;
59. margin-right: 10px;
60. }
61. button:first-child:hover {
62. background-color: #45a049;
63. }
64. button:last-child {
65. background-color: #f44336;
66. }
67. button:last-child:hover {
68. background-color: #e53935;
69. }
70. .footer {
71. text-align: center;
72. margin-top: 25px;
73. font-size: 13px;
74. color: #888;
75. }
76. </style>
77. </head>
78. <body>
79. <div class="container">
80. <h2> AES-128 File Encrypt/Decrypt</h2>
81. <form method="POST" enctype="multipart/form-data">
82. <label for="key">Khóa (16 ký tự):</label>
83. <input type="text" name="key" id="key" maxlength="16" required>
84. <label for="file">Chọn file:</label>
85. <input type="file" name="file" id="file" required>
86. <div class="buttons">
87. <button type="submit" name="action" value="encrypt">Mã Hóa</button>
88. <button type="submit" name="action" value="decrypt">Giải Mã</button>
89. </div>
90. </form>
91. <div class="footer">
92. cntt1707 practice AES cypher
93. </div>
94. </div>
95. </body>
96. </html>

**4. Code**

from flask import Flask, render\_template, request, send\_file

from Crypto.Cipher import AES

from Crypto.Util.Padding import pad, unpad

import os

import io

app = Flask(\_\_name\_\_)

UPLOAD\_FOLDER = "uploads"

os.makedirs(UPLOAD\_FOLDER, exist\_ok=True)

@app.route("/", methods=["GET", "POST"])

def index():

    if request.method == "POST":

        key\_text = request.form.get("key")

        action = request.form.get("action")

        file = request.files.get("file")

        if not file or not key\_text:

            return "Vui lòng chọn file và nhập khóa.", 400

        key = key\_text.encode("utf-8")

        if len(key) < 16:

            key = key.ljust(16, b"\0")

        elif len(key) > 16:

            key = key[:16]

        file\_data = file.read()

        filename = file.filename

        output = io.BytesIO()

        if action == "encrypt":

            cipher = AES.new(key, AES.MODE\_CBC)

            ciphertext = cipher.encrypt(pad(file\_data, AES.block\_size))

            output.write(cipher.iv + ciphertext)

            output.seek(0)

            return send\_file(output, as\_attachment=True, download\_name=filename + ".aes")

        elif action == "decrypt":

            iv = file\_data[:16]

            ciphertext = file\_data[16:]

            cipher = AES.new(key, AES.MODE\_CBC, iv)

            try:

                plaintext = unpad(cipher.decrypt(ciphertext), AES.block\_size)

            except ValueError:

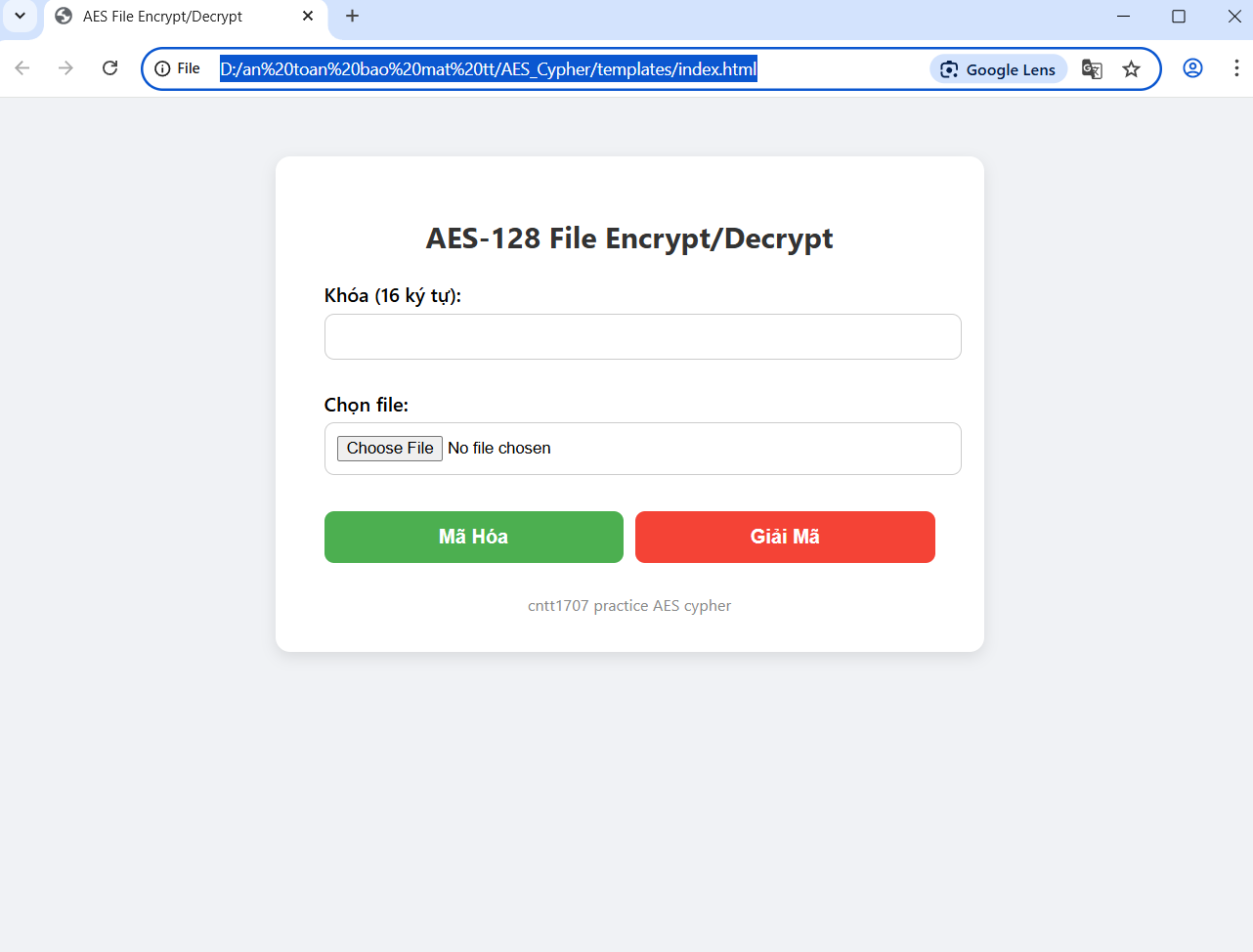
                return "Sai khóa hoặc file không hợp lệ.", 400

            output.write(plaintext)

            output.seek(0)

            return send\_file(output, as\_attachment=True, download\_name="decrypted\_" + filename)

    return render\_template("index.html")



*Kết quả*