

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN #2

LẬP TRÌNH SOCKET

XÂY DỰNG WEB SERVER ĐƠN GIẢN



Lớp 18TN – Môn Mạng Máy Tính

Thành viên nhóm:

18120160 – Hứa Huy Cường

18120593 – Trần Quang Tiến

Thành phố Hồ Chí Minh, ngày 26, tháng 06, năm 2020

MỤC LỤC

I. GIỚI THIỆU ĐỒ ÁN	3
1. Thông tin đồ án	3
1.1. Yêu cầu đồ án	3
1.2. Thời gian thực hiện đồ án	3
2. Thông tin nhóm	3
2.1. Thành viên nhóm	3
2.2. Phân công công việc	3
II. Thực hiện đồ án	4
1. Ngôn ngữ lập trình	4
2. Quá trình xây dựng Web Server:	4
2.1. Tạo socket, gán địa chỉ, port cho server và cho server lắng nghe request từ trình duyệt: 4	
2.2. Nhận request và xử lý:	4
2.2.1. Xử lý khi request method là GET	5
2.2.2. Xử lý khi request method là POST	5
2.2.3. Hậu xử lý	5
2.3. Xử lý Exception và đóng kết nối	6
3. Quá trình xây dựng các file html:	7
3.1.1. Index.html	7
3.1.2. Info.html	7
3.1.3. Files.html	8
3.1.4. 404.html	9
III. Kiểm thử đồ án	10
1. Chạy thử - kiểm tra	10
IV. Đánh giá kết quả	13
1. Đánh giá mức độ hoàn thành đồ án	13
2. Đánh giá các thành viên nhóm	13
TÀI LIỆU THAM KHẢO	14

I. GIỚI THIỆU ĐỒ ÁN

1. Thông tin đồ án

1.1. Yêu cầu đồ án

Sinh viên sử dụng lập trình Socket để viết 1 ứng dụng Web Server để trả về các nội dung trang web sau:

1. Trả về nội dung trang index.html để trình duyệt web có thể hiển thị, nội dung trang này yêu cầu người dùng nhập username và password để đăng nhập theo mẫu tham khảo.
2. Từ trang index.html trên, người dùng nhập thông tin username là “admin” và password “admin” (sử dụng HTTP method là POST để gửi lên Web Server)
 - a. Nếu đúng thông tin đăng nhập (admin/admin) thì redirect đến trang info.html (Tham khảo: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Redirections>). Trang này bao gồm hình ảnh và thông tin của các thành viên trong nhóm sinh viên thực hiện đồ án.
 - b. Nếu sai thông tin đăng nhập, thì trả về lỗi 404 cho trình duyệt.
3. Tạo trang web files.html, hiển thị danh sách các files cho phép người dùng download các files này thông qua trình duyệt Web.
- Cho phép trình duyệt Web có thể sử dụng các HTTP methods GET và POST khi gửi yêu cầu (request) đến Web Server.
- Cho phép tải và hiển thị được hình ảnh của trang Web.

Lưu ý:

- Không yêu cầu quản lý login session và quản lý cookies.
- Không sử dụng thư viện HTTP server có sẵn

1.2. Thời gian thực hiện đồ án

Thời gian thực hiện: 2 tuần (từ 16/06/2020 đến 30/06/2020)

2. Thông tin nhóm

2.1. Thành viên nhóm

MSSV	Họ và tên	Vai trò
18120160	Hứa Huy Cường	Leader
18120593	Trần Quang Tiến	Contributor

Bảng 1: Giới thiệu thành viên nhóm

2.2. Phân công công việc

Họ và tên	Công việc	Mô tả
Hứa Huy Cường	Xây dựng source code Web server	Hoàn thành
	Xây dựng các file html	Hoàn thành
	Test server	Hoàn thành
Trần Quang Tiến	Xây dựng source code Web server	Hoàn thành
	Xây dựng các file html	Hoàn thành
	Test server	Hoàn thành

Bảng 2: Phân công công việc

II. Thực hiện đồ án

1. Ngôn ngữ lập trình

Source Code được viết trên ngôn ngữ Python.

2. Quá trình xây dựng Web Server:

2.1. Tạo socket, gán địa chỉ, port cho server và cho server lắng nghe request từ trình duyệt:

```
1 import socket
2 import os
3 dirname = os.path.dirname(__file__) # Get dir of present file
4
5 HOST, PORT = '127.0.0.1', 8082      # Declare HOST address, using port
6
7 my_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8 my_socket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
9 my_socket.bind((HOST, PORT))
10 my_socket.listen(2)
11 print('Server is on! Serving at port ', PORT)
```

2.2. Nhận request và xử lý:

```
13 while True:
14     connection, address = my_socket.accept()
15     request = connection.recv(1024).decode('utf-8') #Get request
16     print(request)
17     string_list = request.split(' ') # Split request from spaces
18
```

Hàm accept() để chấp nhận kết nối, trả về kết nối socket và địa chỉ client
recv(1024) để nhận tối đa 1024 bytes trong chi tiết yêu cầu từ client, và định dạng sang utf-8
ví dụ một yêu cầu từ client

```
GET / HTTP/1.1
Host: 127.0.0.1:8082
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:49.0) Gecko/20100101 Firefox/49.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

ta cần tách ra phương thức (GET/POST) gán vào **method** và địa chỉ mà client yêu cầu gán vào **myfile**, như ví dụ ở trên '/' là root

```
19     try:
20         method = string_list[0]      # Method = (GET,POST)
21         requesting_file = string_list[1]
22         myfile = requesting_file.split('?')[0] # After the "?" symbol not relevant
23             here
24         myfile = myfile.lstrip('/') #Delete the '/' symbol
25
```

2.2.1. Xử lý khi request method là GET

```
25         header = 'HTTP/1.1 200 OK\n'
26         if (method == "GET"):          #when the method is GET
27             if (myfile == ''):        #if the address have no request file
28                 myfile = os.path.join(dirname, 'index.html') #Load index.html file
29                     as default
30             header = 'HTTP/1.1 301 Moved Permanently\nLocation:
31                 http://127.0.0.1:8082/index.html\n' #send header to redirect to
32                 index.html
33         else:
34             myfile = os.path.join(dirname, myfile)
35
36
37
38
39
40
41
42
43
44
45
46
47
```

Nếu như client yêu cầu đến địa chỉ root, ta mặc định chuyển hướng sang index.html theo như yêu cầu của đồ án với code 301

2.2.2. Xử lý khi request method là POST

```
33     elif (method == "POST"):      #when the method is POST
34         if (request.find("uname") > 0):      #find uname on POST request
35             x = request.split("uname=")
36             name = x[1].split('&')
37             user = name[0]                  #get username on POST request
38             pwd = name[1].split('=')[1]      #get password on POST request
39             if (user == 'admin' and pwd == 'admin'):    #check if username and
39                 password are correct?
40             header = 'HTTP/1.1 302 Found\nLocation:
41                 http://127.0.0.1:8082/info.html\n' #redirect to info.html
42             myfile = os.path.join(dirname, 'info.html')
43         else:
44             header = 'HTTP/1.1 404 Not Found\n'    #redirect to 404.html
45             myfile = os.path.join(dirname, '404.html')
46         else:
47             myfile = os.path.join(dirname, myfile)
```

Khi tìm thấy username và password trong POST request từ client, nếu đúng là admin/admin thì redirect đến trang info.html với code 302, sai thông tin đăng nhập thì trả về lỗi 404 và hiện nội dung 404.html

2.2.3. Hậu xử lý

```
47
48     file = open(myfile, 'rb')  # open file , r => read , b => byte format
49     response = file.read()
50     file.close()
51
52     if (myfile.endswith(".jpg")):
53         mimetype = 'image/jpg'
54     elif (myfile.endswith(".css")):
55         mimetype = 'text/css'
56     elif (myfile.endswith(".png")):
57         mimetype = 'image/png'
58     else:
59         mimetype = 'text/html'
60
61     header += 'Content-Type: ' + str(mimetype) + '\n\n'
```

Đọc file và truyền dữ liệu vào response. Nhận diện và thêm vào header các định dạng phổ biến như ảnh, text,...

2.3. Xử lí Exception và đóng kết nối

```
62
63     except Exception as e:
64         header = 'HTTP/1.1 404 Not Found\n'      #handle exception: redirect to
65         404.html
66         myfile = os.path.join(dirname,'404.html')
67         file = open(myfile, 'rb')    # open file , r => read , b => byte format
68         response = file.read()
69         file.close()
70         mimetype = 'text/html'
71         header += 'Content-Type: ' + str(mimetype) + '\n\n'
72
73         final_response = header.encode('utf-8')
74         final_response += response
75         connection.send(final_response)
76         connection.close()
```

Nếu người dùng truy cập đến địa chỉ không tồn tại trong server thì trả về lỗi 404, hiển thị nội dung 404.html

Lúc này đã có header và content (response), ta gộp lại và truyền đến client để hiển thị. Sau đó đóng kết nối và tiếp tục vòng lặp lắng nghe

3. Quá trình xây dựng các file html:

- * Các thiết kế CSS được tham khảo từ Internet.
- * Các demo chạy thử file html sẽ được báo cáo trong phần Kiểm Thủ Ứng Dụng

3.1.1. Index.html

- Ở trang index, ta cần thiết kế một login form để người dùng đăng nhập vào và sử dụng method POST để gửi request đến server

```
62 <div class="login">
63   <h1>Login</h1>
64   <form method="post">
65     <input type="text" name="uname" placeholder="Username" required="required"
66       />
67     <input type="password" name="psw" placeholder="Password"
68       required="required" />
69     <button type="submit" class="btn btn-primary btn-block btn-large">Let me
70       in.</button>
71   </form>
72 </div>
```

- Dòng thứ 56 khai báo một form điền thông tin và method sử dụng ở đây là post

3.1.2. Info.html

- Ở file info, ta thiết kế một trang hiện thông tin cá nhân của các thành viên tham gia thực hiện đồ án, và một nút chuyển hướng đến Files.html để lấy các files yêu cầu ở câu 3.

```
35 <body>
36
37 <h1>SOCKET PROJECT – WEB SERVER HTTP – PYTHON BASED</h1>
38 <h2>MEMBERS' INFO:</h2>
39
40 <div>
41   <p style="float: left; padding: 0 3em 2em 0;"></p>
43   <br>
44   <p>18120160 – HUÁ (LINHTÂM) HUY CƯỜNG </p>
45   <p>Student at University of Science – VNU – HCMC, class 18TN, Information
46     Technology major </p>
47   <p>Hoa đã có chậu, cá đã có hồ, chim đã có tổ. </p>
48 </div>
49 <div style="clear: left;">
50   <p style="float: left; padding: 0 3em 2em 0;"></p>
52   <br>
53   <p>18120593 – TRẦN (LONELY) QUANG TIẾN </p>
54   <p>Student at University of Science – VNU – HCMC, class 18TN, Information
55     Technology major </p>
56   <p>Trái tim nhân hậu, rộng rãi, thoáng mát. </p>
57 </div>
```

- Phần body chứa nội dung cần hiển thị như trên.

```

60 <p align=right>
61 <button onclick="location.href='files.html'" class="button">Go to
   files.html</button>
62 </p>

```

- Thêm một nút chuyển hướng sang Files.html tại cuối trang

3.1.3. *Files.html*

- Để thiết kế hiển thị Files, ta cần thiết kế kiểu dữ liệu bảng (gồm 4 cột: Name, Date Modified, Size, Preview) cho dễ nhìn hơn:

```

4  <head>
5      <title>files.html</title>
6      <style>
7          table {
8              width: 100%;
9              border-spacing: 0;
10             border-collapse: collapse;
11
12         }
13         table td {
14
15     }
16         table td,
17         table th {
18             border: 1px solid #dadada;
19             padding: 5px;
20         }
21         table thead th {
22             text-align: left;
23         }
24         table thead th:nth-child(0),
25         table thead th:nth-child(1){
26             width: 25%;
27         }
28
29         table thead th:nth-child(2){
30             width: 20%;
31         }
32         table thead th:nth-child(3){
33             width: 20%;
34         }
35         img {
36             max-width: 100%;
37             height: auto;
38         }
39     </style>

```

- Nội dung hiển thị gồm 4 cột, phần Name có liên kết link để có thể download khi nhấp vào, phần Preview sẽ hiển thị ảnh xem trước cho người dùng.
- Ngoài ra, còn có một nút Home ở đầu trang liên kết đến trang index.html
- Ở trang files, nhóm chúng em để săn 3 ảnh dạng png.

```

44 <pre><A HREF="/">[Home]</A><br><br>
45   <table>
46     <thead>
47       <tr>
48         <th>Name</th>
49         <th>Last Modified</th>
50         <th>Size</th>
51         <th>Preview</th>
52       </tr>
53     </thead>
54     <tbody>
55       <tr>
56         <td><a href="avatar.png" download="avatar.png"> avatar.png </a></td>
57         <td>3/12/2020 9:47 AM </td>
58         <td>8229 bytes </td>
59         <td> </td>
60       </tr>
61       <tr>
62         <td><a href="img_avatar2.png" download="avatar.png">img_avatar2.png</a></td>
63         <td>3/12/2020 9:49 AM </td>
64         <td>8314 bytes </td>
65         <td> </td>
66       </tr>
67       <tr>
68         <td><a href="avatar.png" download="avatar.png"> avatar.png </a></td>
69         <td>3/12/2020 9:50 AM </td>
70         <td>8229 bytes </td>
71         <td> </td>
72       </tr>
73     </tbody>
74   </table>
75

```

3.1.4. 404.html

```

<div class="mainbox">
  <div class="err">4</div>
  <i class="far fa-question-circle fa-spin"></i>
  <div class="err2">4</div>
  <div class="msg">Maybe this page moved? Got deleted? Is hiding out in
    quarantine? Never existed in the first place?<p>Let's go <a
    href="/">home</a> and try from there.</p></div>
</div>

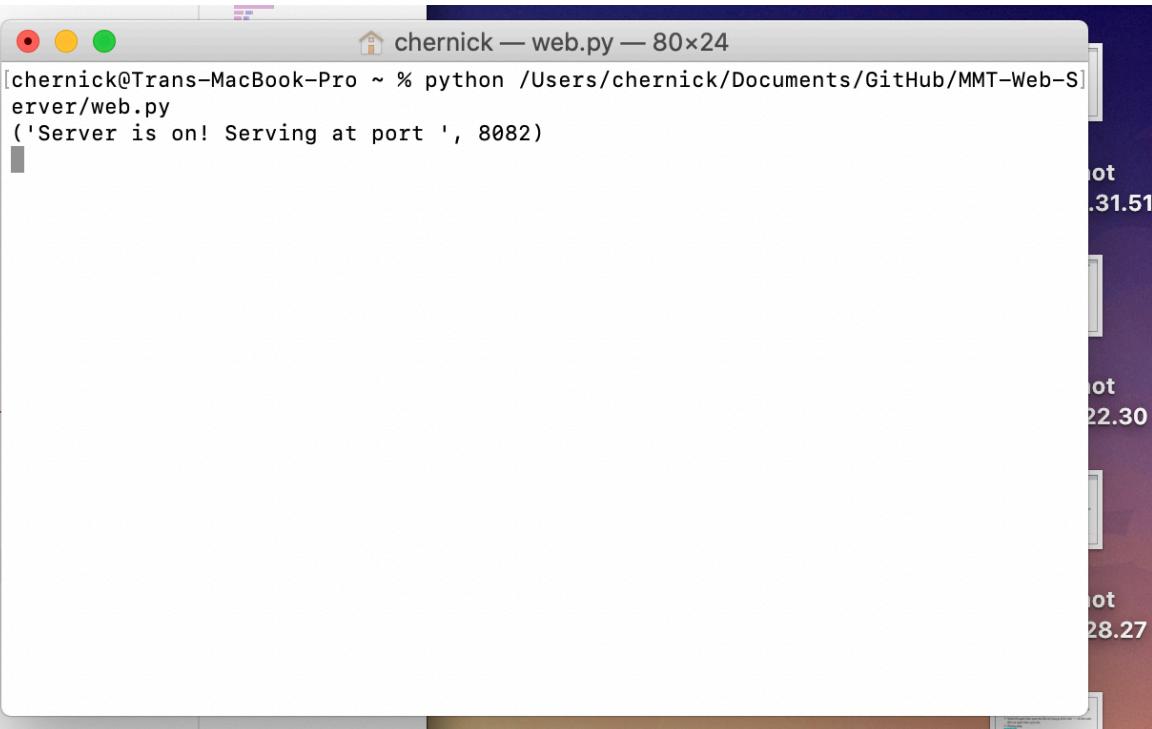
```

- File 404.html được sử dụng khi người dùng request một trang không tồn tại trong server, hoặc người dùng nhập sai username và password khi login.
- Nhấn home để quay trở về trang trước đó.

III. Kiểm thử đồ án

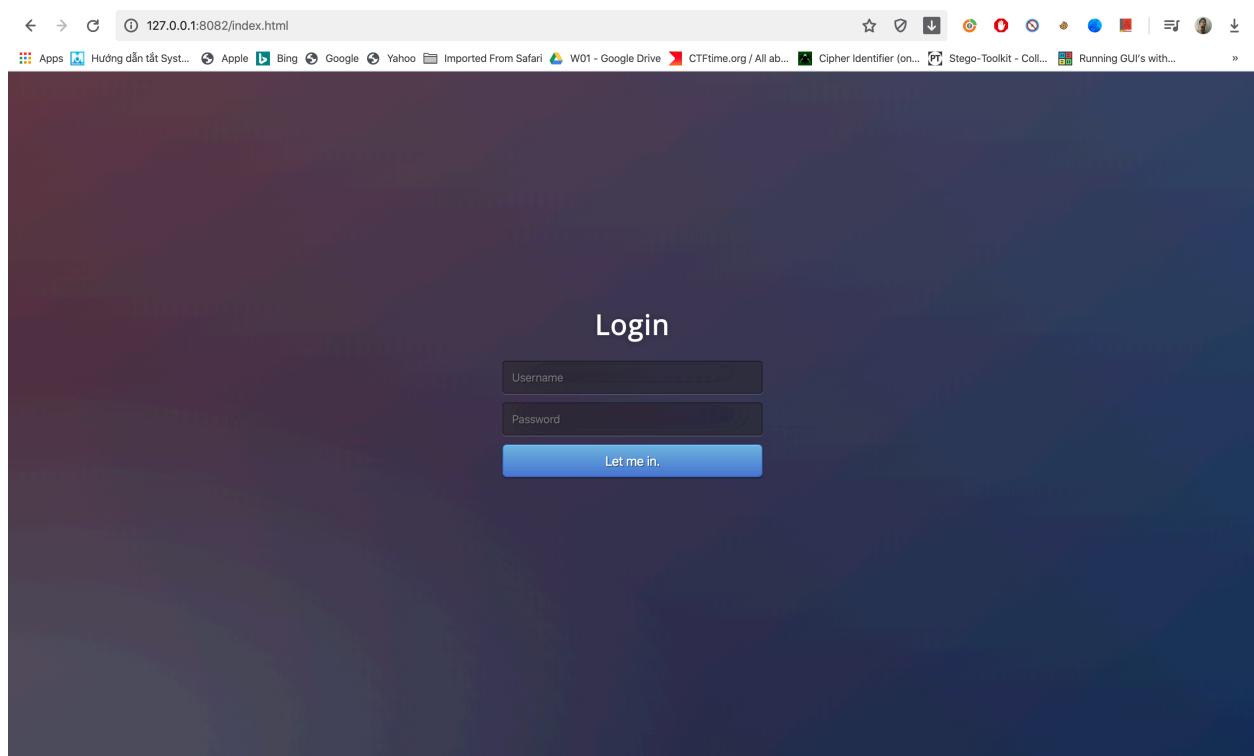
1. Chạy thử - kiểm tra

- Để chạy thử, ta cần chạy file web.py trên Command Prompt (Ở đây em sử dụng Terminal trên hệ điều hành MacOS)
- Sau khi chạy, sẽ có thông báo Server đã hoạt động, ở port 8082.



```
chernick — web.py — 80x24
[chernick@Trans-MacBook-Pro ~ % python /Users/chernick/Documents/GitHub/MMT-Web-SERVER/web.py
('Server is on! Serving at port ', 8082)
```

- Sau khi Server đã hoạt động, ta mở trình duyệt (ở đây em sử dụng Chrome) để test Ứng dụng, truy cập vào địa chỉ 127.0.0.1:8082, ta được trang web sau:



- Đăng nhập vào trang web với Username/Password: admin/admin, ta sẽ được redirect tới trang sau:

SOCKET PROJECT - WEB SERVER HTTP - PYTHON BASED

MEMBERS' INFO:

18120160 - HÚA (LINHTÂM) HUY CUỐNG
 Student at University of Science - VNU - HCMC, class 18TN, Information Technology major
 Hoa dã có chậu, cá dã có hồ, chim dã có tổ.

18120593 - TRẦN (LONELY) QUANG TIỀN
 Student at University of Science - VNU - HCMC, class 18TN, Information Technology major
 Trái tim nhân hậu, rông rãi, thoảng mát.

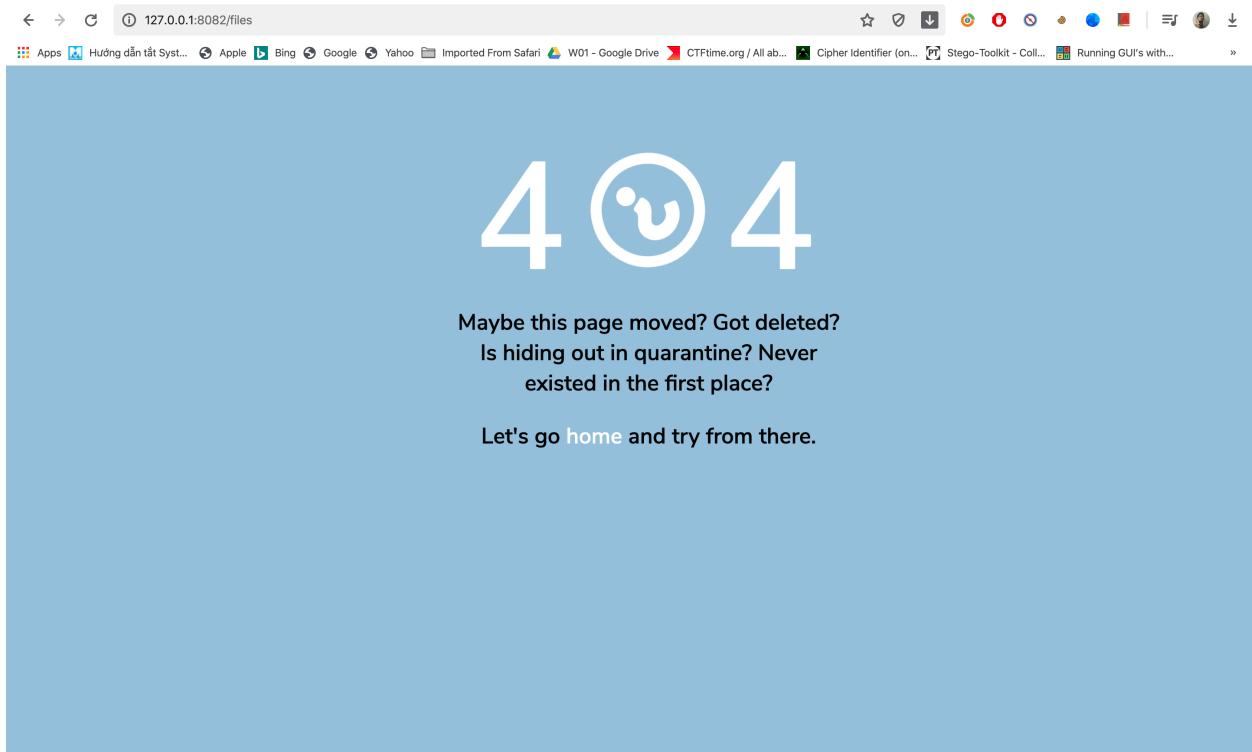
Go to files.html

- Nhấn vào Go to files.html để đến trang get files:

Name	Last Modified	Size	Preview
avatar.png	3/12/2020 9:47 AM	8229 bytes	
img_avatar2.png	3/12/2020 9:49 AM	8314 bytes	
avatar.png	3/12/2020 9:50 AM	8229 bytes	

- Nhấn vào tên hình ảnh để tải về hoặc nhấn home để quay lại trang index.html

- Nếu ta đăng nhập sai, hoặc request đến 1 đường link không tồn tại, trang 404.html sẽ xuất hiện:



IV. Đánh giá kết quả

1. Đánh giá mức độ hoàn thành đồ án

YÊU CẦU		MỨC ĐỘ HOÀN THÀNH
Lập trình Web Server	Tạo HTTP Server (Host, Port)	100%
	Redirect	100%
	Báo lỗi 404 Not Found	100%
	GET Method	100%
	POST Method	100%
Xây dựng html file	File index có login form và sử dụng POST method	100%
	File info chứa thông tin	100%
	File files chứa các file hình ảnh (có hiển thị và cho phép tải về)	100%
	File 404 trả về thông tin lỗi	100%
TỔNG CỘNG		100%

Bảng 3: Đánh giá mức độ hoàn thành

2. Đánh giá các thành viên nhóm

MSSV	HỌ TÊN	KHỐI LƯỢNG CÔNG VIỆC	MỨC ĐỘ HOÀN THÀNH
18120160	Hứa Huy Cường	55%	100%
18120593	Trần Quang Tiến	45%	100%

Bảng 4: Đánh giá mức độ đóng góp

TÀI LIỆU THAM KHẢO

- [1]. <https://emalsha.wordpress.com/2016/11/22/how-create-http-server-using-python-socket/>
- [2]. <https://developer.mozilla.org/en-US/docs/Web/HTTP/Redirections>
- [3]. <https://codepen.io/Lewitje/pen/BNNJjo> (login form template)
- [4]. <https://codepen.io/purplexmoss/pen/PoPyzMW> (404 Not Found html template)