

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



MẠNG MÁY TÍNH
ĐỒ ÁN LẬP TRÌNH SOCKET

Lớp: MMT CQ2018/2

Nhóm thực hiện:

18120078 – Ngô Phù Hữu Đại Sơn

18120259 – Lê Hoàng Mộng Tuyền

18120263 – Nguyễn Quang Vinh

MỤC LỤC

A. Thông tin khái quát.....	2
I. Thông tin nhóm	2
II. Bảng phân công công việc	2
B. Nội dung.....	3
I. Môi trường phát triển	3
II. Mục tiêu của đồ án:	3
III. Ý tưởng	3
IV. Triển khai.....	4
1. Thiết kế HTTP server	4
2. Thiết kế trang đăng nhập	6
3. Thiết kế trang thông tin.....	6
V. Kiểm thử	7
1. Khởi chạy server.....	7
2. Mở browser tạo kết nối tới server	7
3. Đăng nhập thất bại.....	8
4. Đăng nhập thành công	8
VII. Đánh giá đồ án.....	9
1. Mức độ hoàn thành của các thành viên	9
3. Mức độ hoàn thành đồ án:	9
VII. Nguồn tham khảo.....	9

A. THÔNG TIN KHÁI QUÁT

I. Thông tin nhóm

MSSV	Họ tên	Vai trò
18120078	Ngô Phù Hữu Đại Sơn	Nhóm trưởng
18120259	Lê Hoàng Mộng Tuyền	Thành viên
18120263	Nguyễn Quang Vinh	Thành viên

II. Bảng phân công công việc

MSSV	Công việc phụ trách
18120078	Viết code socket + Viết báo cáo
18120078	Viết báo cáo
18120259	Viết code info.html
18120259	Viết code styles_infor.css
18120263	Viết code index.html
18120263	Viết code styles_index.css

B. NỘI DUNG

I. Môi trường phát triển

- Ngôn ngữ lập trình: Python, HTML, CSS
- Code editor: Visual Studio Code 2019

II. Mục tiêu của đồ án:

Xây dựng web server đơn giản để trả về các nội dung trang web:

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
2. Từ trang index.html trên, người dùng nhập thông tin username là "admin" và password "admin".
 - 2.1. Nếu đúng thông tin đăng nhập (admin/admin) thì redirect đến trang info.html chứa hình ảnh và thông tin các sinh viên trong nhóm thực hiện đồ án
 - 2.2. Nếu sai thông tin đăng nhập thì trả về lỗi 404 cho trình duyệt.

III. Ý tưởng

- Sử dụng ngôn ngữ python thiết kế HTTP server nhận kết nối với các clients, nhận request xử lý và response về clients
- Sử dụng ngôn ngữ html thiết kế trang đăng nhập cho người dùng, trang thông tin của các thành viên sau khi đăng nhập thành công, và trang lỗi "404 Not Found"
- Sử dụng ngôn ngữ css để tùy chỉnh giao diện(vị trí, màu sắc, kiểu chữ,...) cho các trang
- Nguyên nhân lựa chọn Python để lập trình socket:
 - Cú pháp lệnh đơn giản, dễ sử dụng
 - Có hỗ trợ đầy đủ các thư viện và hàm cho lập trình Socket

IV. Triển khai

1. Thiết kế HTTP server

- Khởi tạo địa chỉ các giá trị mặc định

```
import socket
import threading

SERVER = socket.gethostname(socket.gethostname())
PORT = 1501
ADDR = (SERVER, PORT)
USERNAME = 'admin'
PASSWORD = 'admin'
```

- Tạo socket

```
httpserver = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
httpserver.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1) #re-use the port if server crack
httpserver.bind(ADDR)
```

- Hàm chờ kết nối từ client. Khi có kết nối sẽ tạo 1 tiểu trình để giao tiếp với client

```
def start():
    httpserver.listen(10)
    print(f"[LISTENING] Server is listening on {ADDR}...")
    try:
        while True:
            conn, addr = httpserver.accept()
            thread = threading.Thread(target = handle_client, args = (conn, addr))
            thread.start()
    except Exception:
        httpserver.close()
    finally:
        httpserver.close()
```

- Hàm luận lý xử lý khi nhận GET request. Trả về đúng nếu request hợp lệ.

```
def GET_request(filename):
    if(filename == 'info.html'):
        return False
    return True
```

- Hàm luận lý xử lý khi nhận POST request. Trả về đúng nếu request hợp lệ.

```
def POST_request(filename, request_body):
    username = request_body.split('&')[0].split('=')[1]
    password = request_body.split('&')[1].split('=')[1]
    if(username == USERNAME and password == PASSWORD):
        return True
    return False
```

- Hàm nhận request từ client xử lý và response.

```
def handle_client(conn,addr):  
    print(f"[NEW CONNECTION] {addr} connected")  
    request = conn.recv(1024).decode('utf8')  
    request_detail = request.split(' ')  
    method = request_detail[0]  
    requesting_file = request_detail[1][1:]  
    if(requesting_file == ''):  
        requesting_file = 'index.html'  
    flag = True  
    if(method == 'GET'):  
        flag = GET_request(requesting_file)  
    else:  
        flag = POST_request(requesting_file, request.split('\n')[-1])  
    if(flag == True):  
        send_response(conn, requesting_file, '200 OK')  
    else:  
        send_response(conn, '404error.html', '404 Not Found')  
    conn.close()  
    print(f"[DISCONNECT] {addr} has disconnected")
```

- Hàm response 1 file về cho client.

```
def send_response(client, filename, status):  
    with open(filename,'rb') as f:  
        body = f.read()  
    header = 'HTTP/1.1' + status + '\n'  
    if(filename.endswith(".jpg")):  
        mimetype = 'image/jpg'  
    elif(filename.endswith(".css")):  
        mimetype = 'text/css'  
    else:  
        mimetype = 'text/html'  
    header += 'Content-Type: '+str(mimetype)+'\n\n'  
    response_msg = header.encode('utf8') + body  
    client.send(response_msg)
```

2. Thiết kế trang đăng nhập

Gửi POST request với thông tin ở phần body là username và password mà người dùng nhập

```
<h2>Login Form</h2>
<form action="info.html" method="post">
  <div class="imgcontainer">
    
  </div>
  <div class="container">
    <label for="uname"><b>Username</b></label>
    <input type="text" placeholder="Username" name="uname" required>

    <label for="psw"><b>Password</b></label>
    <input type="password" placeholder="Password" name="psw" required>

    <button type="submit">Login</button>
    <label><input type="checkbox" name="remember"> Remember me</label>
  </div>
</form>
```

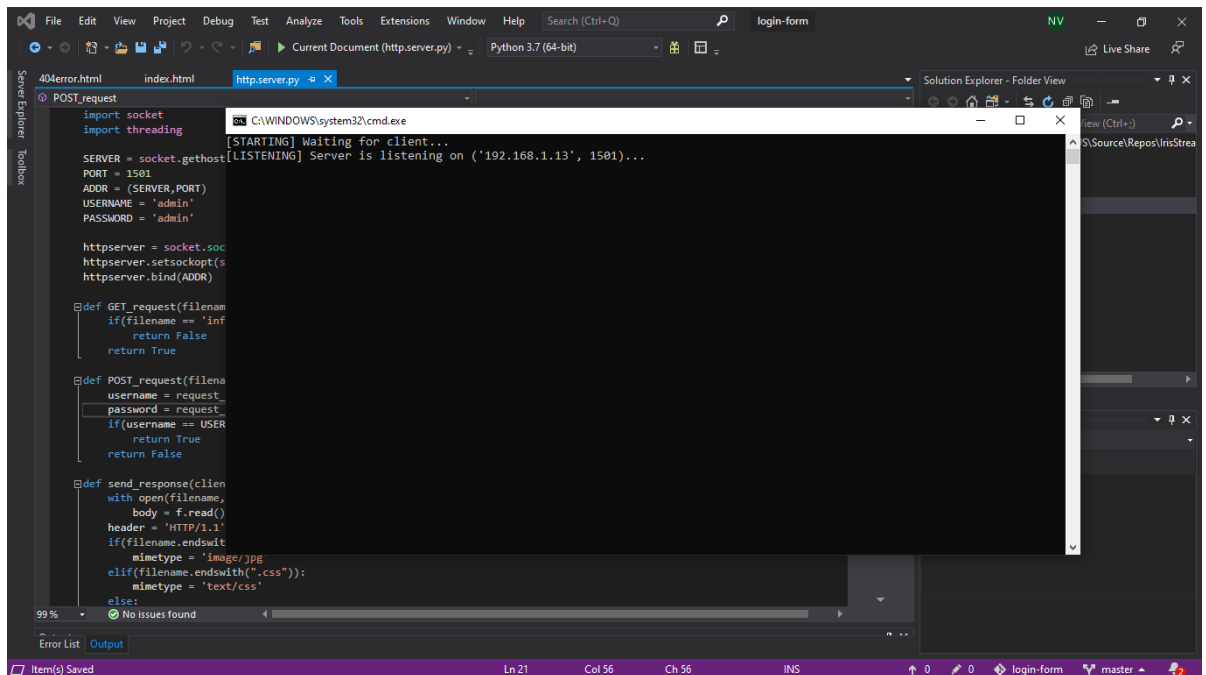
3. Thiết kế trang thông tin

Thông tin của mỗi thành viên được gói trong các thẻ. Tạo 1 nút dẫn đến facebook của thành viên.

```
<div class="column">
  <div class="card">
    <center></center>
    <div class="container">
      <h2>Ngô Phù Hữu Đại Sơn</h2>
      <p class="title">Nhóm trưởng</p>
      <div>MSSV: 18120078</div>
      <div>Lớp: 18CTT1</div>
      <div>Email: ngodaison@gmail.com</div>
      <form action="https://www.facebook.com/dai.son7544/" method="GET">
        <p><button class="button">Contact</button></p>
      </form>
    </div>
  </div>
</div>
```

V. Kiểm thử

1. Khởi chạy server



The screenshot shows a Visual Studio Code editor with a Python file named `http.server.py`. The code defines a simple HTTP server using `socket` and `threading` modules. It listens on `192.168.1.13` at port `1501`. The server handles `GET` and `POST` requests. A terminal window in the foreground shows the command `C:\WINDOWS\system32\cmd.exe` and the output: `[STARTING] Waiting for client...` and `[LISTENING] Server is listening on ('192.168.1.13', 1501)...`.

```

import socket
import threading

SERVER = socket.gethostbyname('localhost')
PORT = 1501
ADDR = (SERVER, PORT)
USERNAME = 'admin'
PASSWORD = 'admin'

httpserver = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
httpserver.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
httpserver.bind(ADDR)

def GET_request(filename):
    if(filename == 'inf'):
        return False
    return True

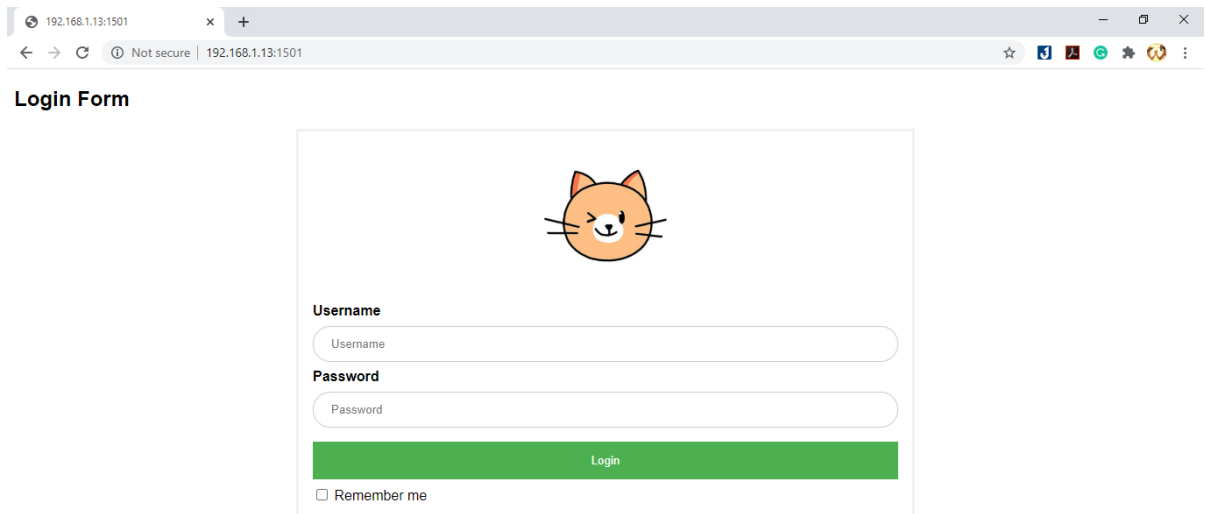
def POST_request(filename, username, password):
    if(username == USERNAME and password == PASSWORD):
        return True
    return False

def send_response(client_socket, filename):
    with open(filename) as f:
        body = f.read()
    header = 'HTTP/1.1'
    if(filename.endswith('.jpg')):
        mimetype = 'image/jpeg'
    elif(filename.endswith('.css')):
        mimetype = 'text/css'
    else:
        mimetype = 'text/html'
    response = header + '\n' + mimetype + '\n' + body
    client_socket.send(response)

httpserver.listen(1)

while True:
    client_socket, address = httpserver.accept()
    thread = threading.Thread(target=send_response, args=(client_socket, 'index.html'))
    thread.start()
    
```

2. Mở browser tạo kết nối tới server



The screenshot shows a web browser window with the address `192.168.1.13:1501`. The page displays a login form titled "Login Form" with a cat icon. The form includes fields for "Username" and "Password", a "Login" button, and a "Remember me" checkbox.

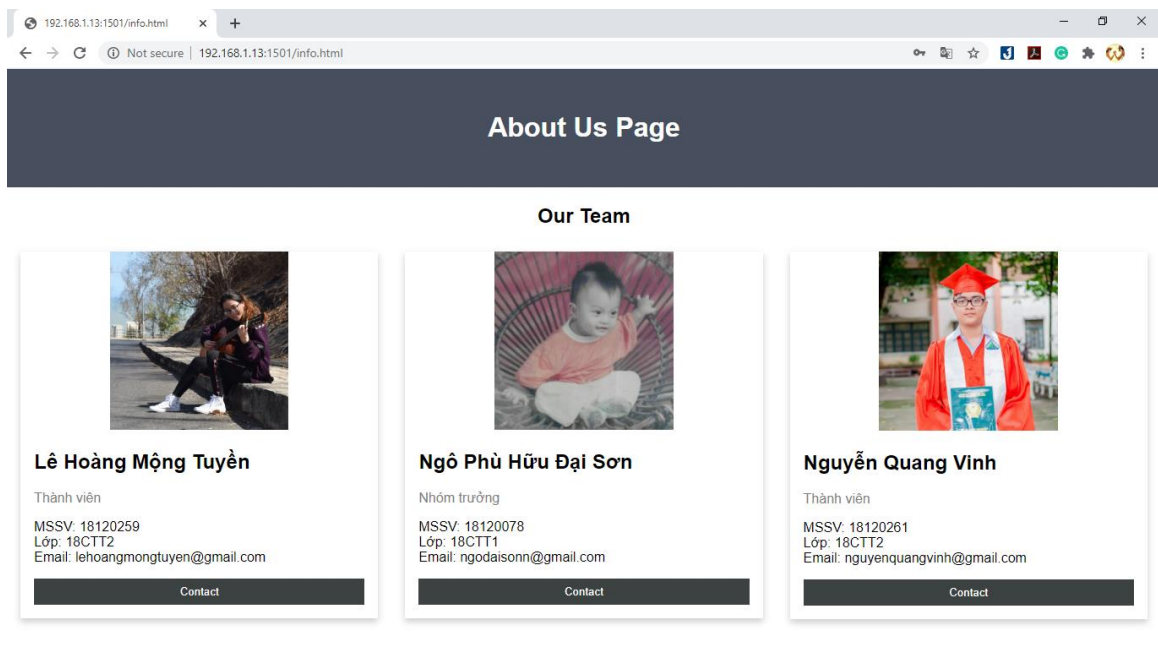
```

<div>
  <div>
    <img alt="Cat icon" data-bbox="510 570 590 610"/>
  </div>
  <div>
    <div>
      Username
      <input type="text" value="Username"/>
    </div>
    <div>
      Password
      <input type="password" value="Password"/>
    </div>
    <div>
      <button type="button" value="Login"/>
    </div>
    <div>
      <input type="checkbox" value="Remember me"/> Remember me
    </div>
  </div>
</div>
  
```


3. Đăng nhập thất bại



4. Đăng nhập thành công



VII. Đánh giá đồ án

1. Mức độ hoàn thành của các thành viên

MSSV	Mức độ hoàn thành công việc	Đóng góp
18120078	100%	40%
18120259	100%	30%
18120263	100%	30%

3. Mức độ hoàn thành đồ án:

- Các chức năng đã hoàn thành
 - Xây dựng được server thực hiện đúng theo kịch bản của phần mềm
 - Xây dựng được trang đăng nhập
 - Xây dựng được trang thông tin thành viên
 - Xây dựng được trang báo lỗi “404 Not Found”
- Các hạn chế
 - Giao diện còn đơn giản. Không có các chức năng nâng cao.

Đánh giá: 100/100

VII. Nguồn tham khảo

<https://stackoverflow.com/>

<https://www.w3schools.com/html/default.asp>

<https://www.w3schools.com/css/default.asp>

<https://www.w3schools.com/python/default.asp>