

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN - ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN



# BÁO CÁO ĐỒ ÁN

## SOCKET

**LỚP : MMT 18CQ/4**

**HỌ TÊN - MSSV : Phạm Hoàng Long-18120452**

**Dương Thành Long-18120444**

**Du Chí Nhân-18120492**

*TP. Hồ Chí Minh – /7/2020*

# 1. Ý tưởng

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
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)
  - Nếu đúng thông tin đăng nhập (admin/admin) thì redirect đến trang info.html. Trang này bao gồm hình ảnh và thông tin của các thành viên trong nhóm sinh viên thực hiện đồ án.
  - Nếu sai thông tin đăng nhập, thì trả về lỗi 404 cho trình duyệt.

## 2. Môi trường lập trình

Sublime Text 3 với ngôn ngữ Python 3.8

## 3. Cách tạo ứng dụng

Import thư viện socket vào python và sử dụng các chức năng được cung cấp khởi tạo 1 server

Do là một Web Server nên sẽ sử dụng port 80 để giao tiếp, ta gán địa chỉ của máy hiện tại với port 80 bằng lên s.bind(('',80))

Thiết lập số client có thể kết nối tới bằng 1 s.listen(1)

Cho Server chạy một vòng lặp vô tận để luôn có thể nhận kết nối đến từ client

Khi kết nối được thiết lập ta mã hóa và phân tách dữ liệu nhận được để biết yêu cầu từ client

```
header_part = request.split('\r\n\r\n')[0]
request_line = header_part.split('\r\n')[0]
method = request_line.split(' ')[0]
link = request_line.split(' ')[1]
```

Hình 3.1: Đoạn mã sử dụng để phân tách và lấy thông tin header

Lấy phần header bằng cách tách dữ liệu nhận về với chuỗi kí tự phân cách là xuống dòng

Trong header sẽ có phương thức gửi (method) cũng như yêu cầu truy cập 1 liên kết (link) nào đó. Từ đó, ta xử lý và điều hướng truy cập của client thông qua các phản hồi được gửi lại

```
if link != '/':
    if link.split('.')[1] == "html":
        link = link.replace("/", "")
        code = '200 OK\n'
        file = codecs.open(link, "r", "utf8")
        response = file.read()
    else:
        code = '204 No content'
else:
    code = '301 Moved permanently\nLocation: index.html'
```

Hình 3.2: Xử lý yêu cầu truy cập liên kết

- Nếu vừa kết nối chưa có yêu cầu truy xuất liên kết cụ thể client sẽ được điều hướng đến trang index.html
- Các yêu cầu không nhận dạng được sẽ được hồi đáp với http message 204 tức không có nội dung về yêu cầu này
- Các yêu cầu còn lại sẽ được chấp thuận và hồi đáp do không sử dụng session nên khi truy cập trực tiếp vào trang info.html mà không thông qua đăng nhập vẫn chưa thể xử lý

Do phương thức gửi là post để yêu cầu truy cập vào info.html với dữ liệu là username và password ở phần data nên ta phải đọc data và xét xem có đúng tài khoản hay không sau đó mới cho phép kết nối, nếu không khớp Server sẽ điều hướng đến trang 404.html

```
if method == 'POST':
    data_part = request.split('\r\n\r\n')[1]
    user, pwd = data_part.split('&')
    user = user.split('=')[1]
    pwd = pwd.split('=')[1]
    if user == 'admin' and pwd == 'admin':
        code = '200 OK\n'
        file = codecs.open(link, "r", "utf8")
        response = file.read()
    else:
        code = '301 Moved permanently\nLocation: 404.html'
```

Hình 3.3: Phân giải dữ liệu đi kèm khi client gửi theo phương thức POST

- Kiểm tra tên đăng nhập nếu đúng thì chấp thuận yêu cầu truy cập. Ngược lại, điều hướng đến 404.html

Sau khi lấy đủ thông tin để có thể phản hồi, ta mã hóa dữ liệu và gửi hồi đáp yêu cầu cho client

```
conn.sendall(bytes('HTTP/1.1 ', "utf8"))
conn.sendall(bytes(code, "utf8"))
if code.split(' ')[0] == '200':
    conn.sendall(bytes('Content-Type: text/html\n', "utf8"))
    conn.sendall(bytes('\n', "utf8"))
    conn.sendall(bytes(response, "utf8"))
```

Hình 3.4: Mã hóa và gửi lại dữ liệu cho client

- Nếu yêu cầu được chấp thuận (http message code = 200), Server sẽ có dữ liệu để hồi đáp cho client

Cuối cùng là đóng kết nối và tiếp tục chờ nhận yêu cầu kết nối

## 4. Hướng dẫn cách test ứng dụng

Đầu tiên chúng ta cài phần mềm Sublime Text 3 để có thể chỉnh sửa và chạy code.

Link: <https://www.sublimetext.com/3>

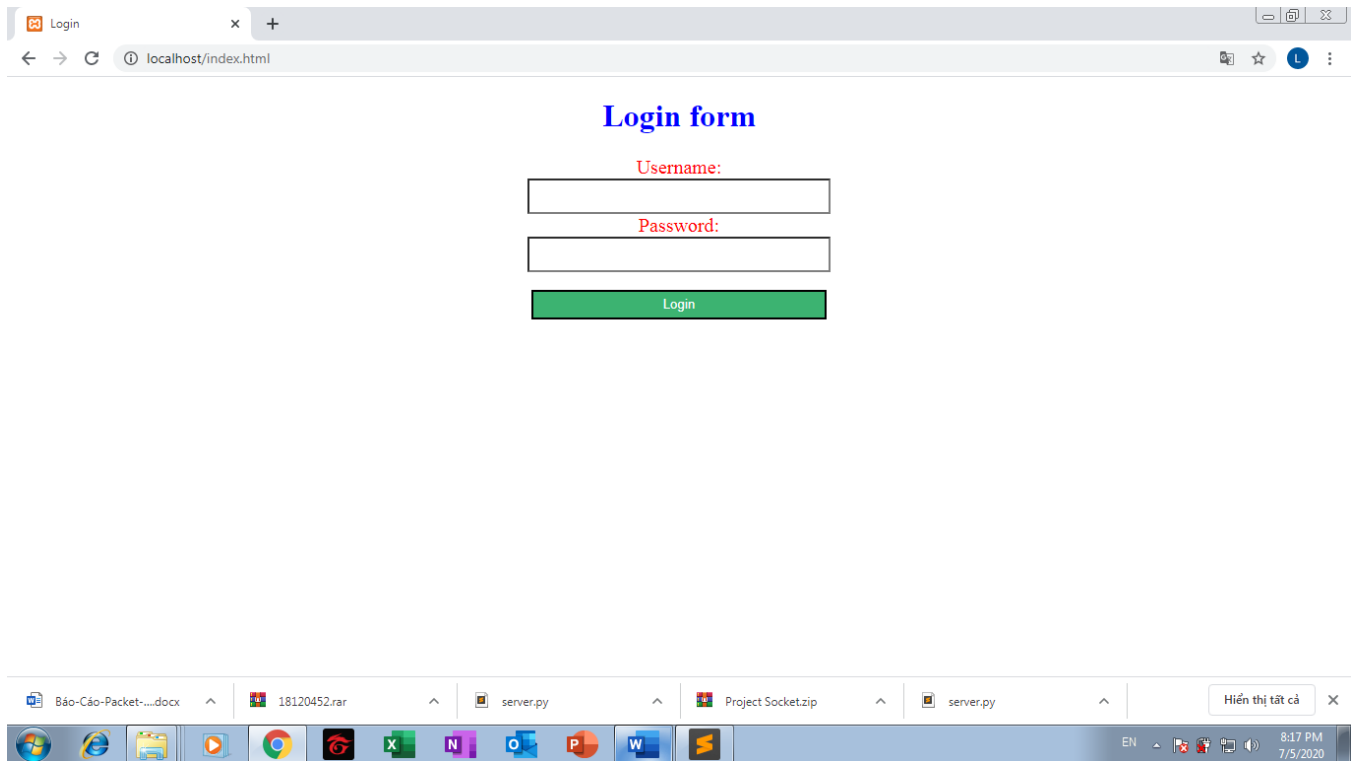
Download bản python mới nhất ở link: <https://www.python.org/downloads/>

Ta cài đặt môi trường python cho sublime text 3 theo hướng dẫn link:

<https://www.youtube.com/watch?v=NRab8bTkI1g>

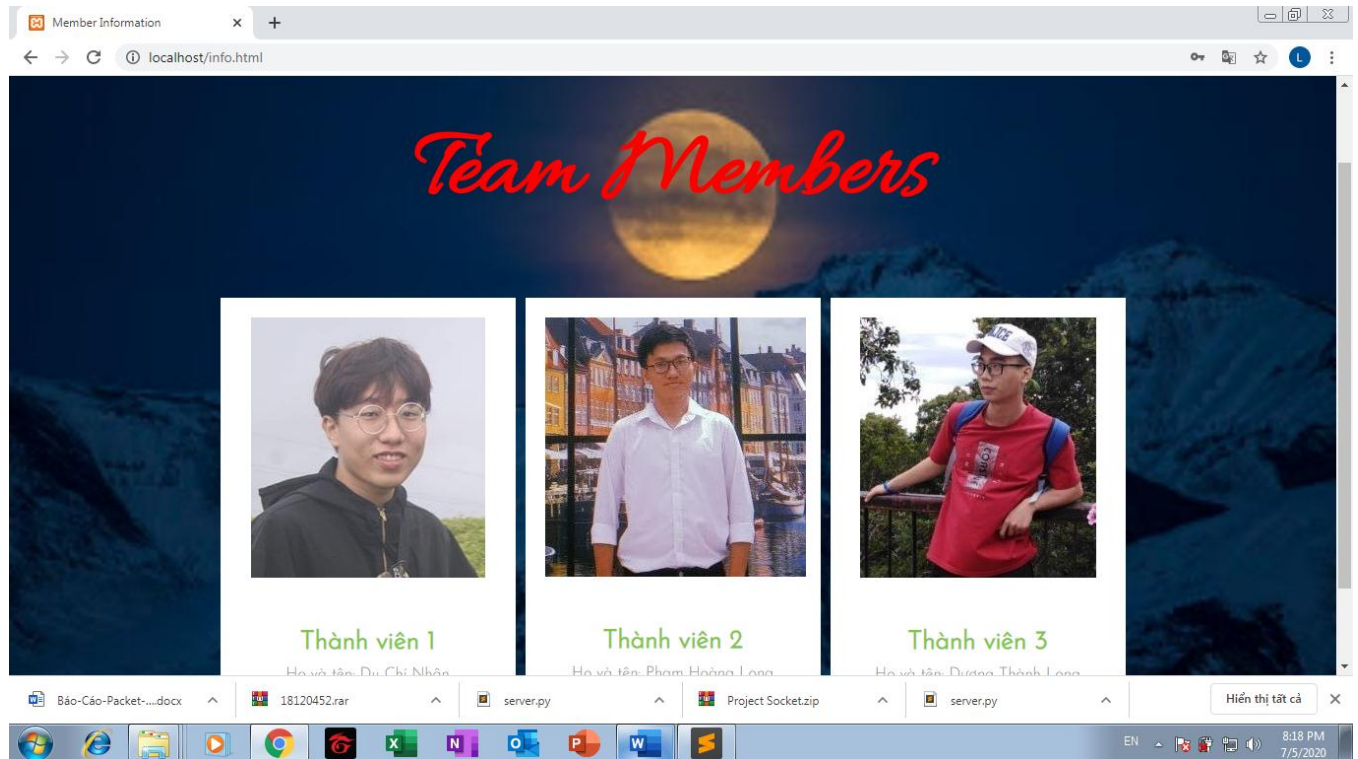
Mở file server.py và build để khởi động server.

Mở bất kì trình duyệt nào để truy cập localhost:80

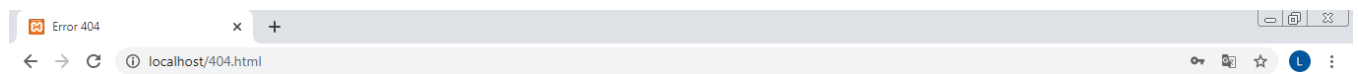


Nhập username với password là admin/admin và Login

Đây là màn hình info.html gồm các thành viên của nhóm



Nếu đăng nhập sai sẽ xuất hiện thông báo lỗi từ trình duyệt



## Error 404

We're sorry but the page you're looking for could not be found!!!!



**Hoàn thành yêu cầu đồ án 100%**