

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN
MÔN: MẠNG MÁY TÍNH

BÁO CÁO ĐỒ ÁN 2
Đồ án Socket: Xây dựng Web Server đơn giản

Giáo Viên: Nguyễn Thanh Quân
Sinh Viên: 18120322 – Lưu Thiện Đức
18120396 – Ngô Quang Huy

Thành phố Hồ Chí Minh – Tháng 7/2020

MỤC LỤC

MỤC LỤC	2
I. THÀNH VIÊN	3
II. Ý TƯỞNG	4
III. CÁCH THỰC HIỆN	5
IV. CÁCH TEST ỨNG DỤNG	7
1. Khi nhập URL	7
2. Khi đăng nhập	8
V. TỰ ĐÁNH GIÁ	10
VI. THAM KHẢO	11

I. THÀNH VIÊN

STT	Tên	MSSV
1	Lưu Thiện Đức	18120322
2	Ngô Quang Huy	18120396

II. Ý TƯỞNG

- Khi người dùng nhập một đường dẫn trên browser thì Web client lập tức sẽ gửi gói tin request đến Web server. Dựa vào cấu trúc của gói tin ta sẽ lấy ra được 3 trường thông tin quan trọng: method, url (đường dẫn yêu cầu), message body. Từ 3 trường đó Web server có thể xử lý và gửi về gói tin response mong muốn cho Web client với cấu trúc chính như sau:

HTTP/1.1 200 OK => Dòng trạng thái

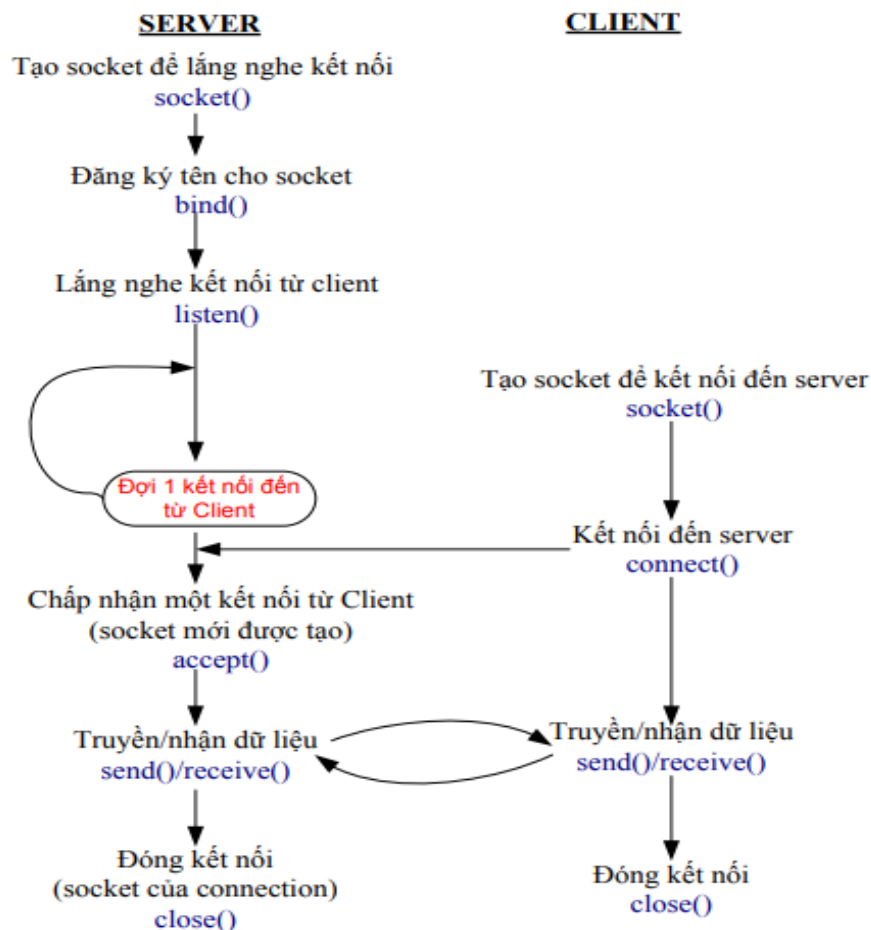
Content-type: text/html => Kiểu nội dung của file

Message body

- Trong đó:
 - Dòng trạng thái: Phiên bản http + mã trạng thái + trạng thái
Mã trạng thái thông báo về kết quả khi nhận được yêu cầu và xử lý
 - Kiểu nội dung của file có thể là:
text/html , text/css, image/png, image/jpeg,
 - Message body là phần nội dung trong url

III. CÁCH THỰC HIỆN

- Trong đồ án lần này, nhóm em sử dụng ngôn ngữ python và đề yêu cầu xây dựng lên 1 ứng dụng Web server thông qua giao thức TCP.
- Tiểu trình chính :
 - Đầu tiên, Ta tạo 1 socket để lắng nghe kết nối
`socket(AF_INET, SOCK_STREAM)`
`SOCK_STREAM`: thể hiện việc đang sử dụng giao thức TCP
 - Tiếp đến, Ta đăng ký tên cho socket qua hàm `bind(Addr)` với `Addr` gồm có host được sử dụng là “localhost” và port là 80. Nếu port 80 đã bị chiếm ta dùng `try....except` và sử dụng port khác (ví dụ là 9000).
 - Tiếp theo, Web server lắng nghe và đợi kết nối từ các client nếu có kết nối thì `accept()`. Đây là tiểu trình con kết nối với nhiều client.
 - Kế đến là quá trình trao đổi gói tin giữa server và client.
 - Cuối cùng là đóng kết nối.

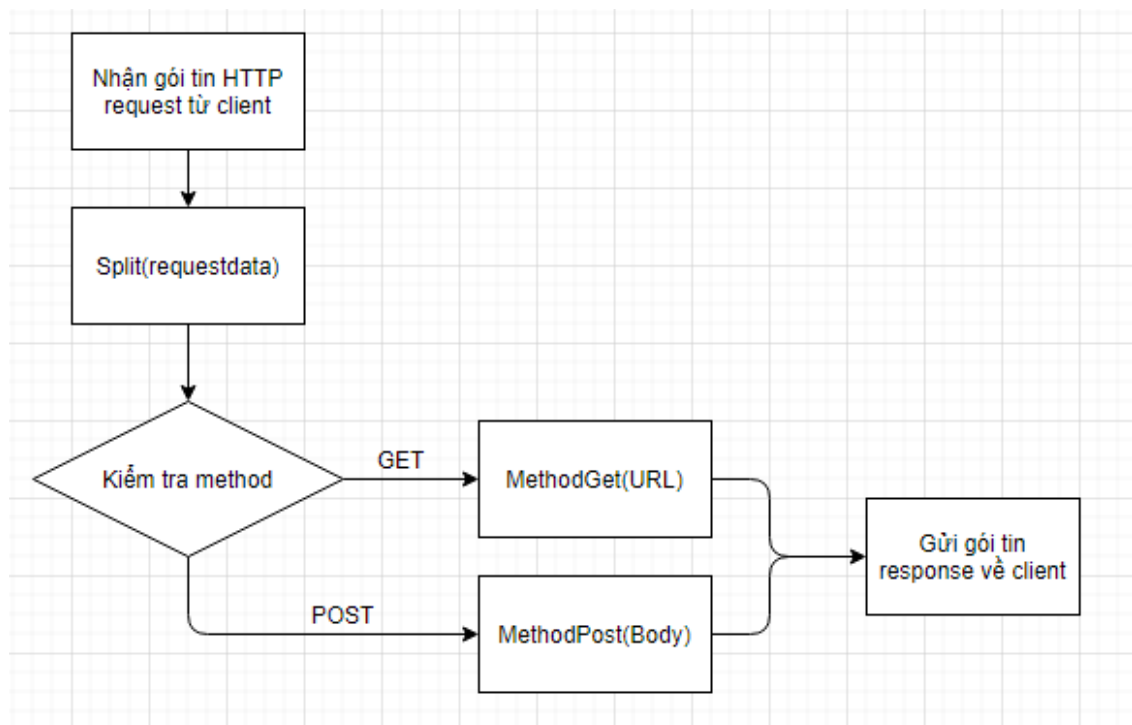


- Trong tiểu trình chính ấy ta có tiểu trình phụ là quá trình trao đổi giữa client và server:
 - Sau khi nhận gói tin từ client, ta sẽ gọi hàm Split để cắt và lấy ra 3 trường thông tin là method, url và bodymessage.
 - Sau đó ta kiểm tra method của gói tin, nếu là GET thì gọi hàm MethodGet() còn nếu là POST sẽ là hàm MethodPost().

Trong đó:

+ MethodGet(URL): sẽ lấy đường dẫn url và trả về nội dung gói tin response nếu đường dẫn đúng thì mã trạng thái là 200(OK) và body message sẽ được đọc lên từ file yêu cầu còn nếu không mã trạng thái sẽ là 404 (NOT FOUND) (Xử lý cho bài 1) .

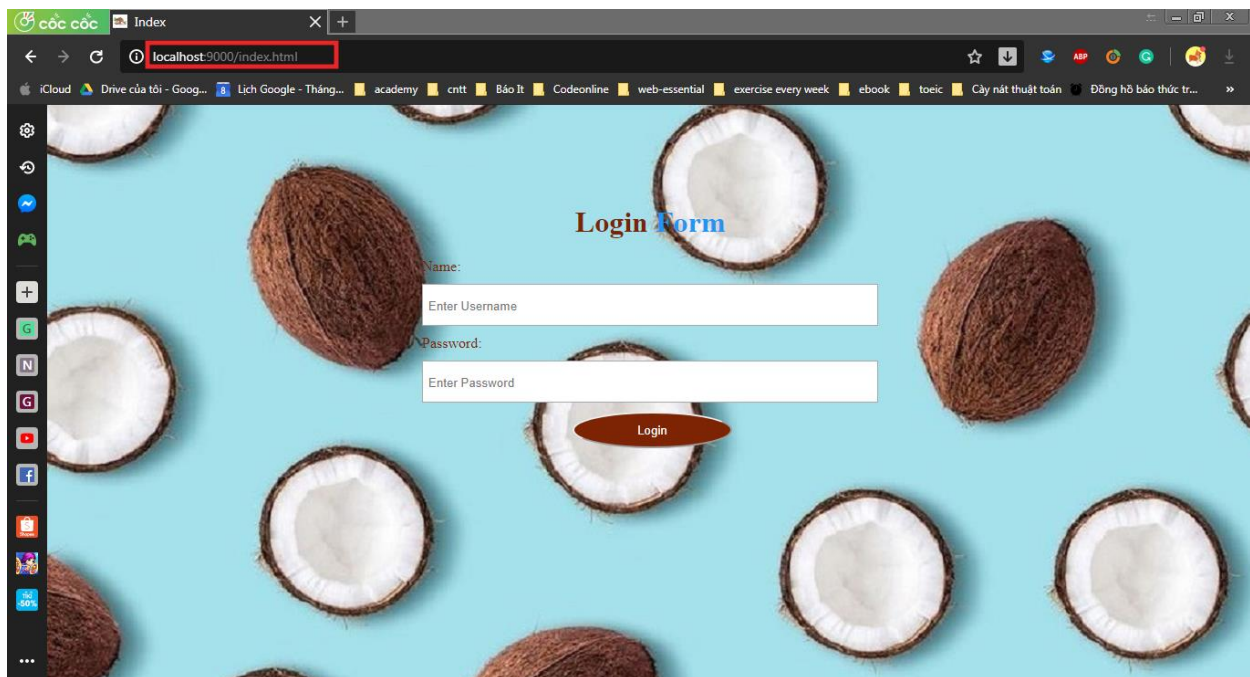
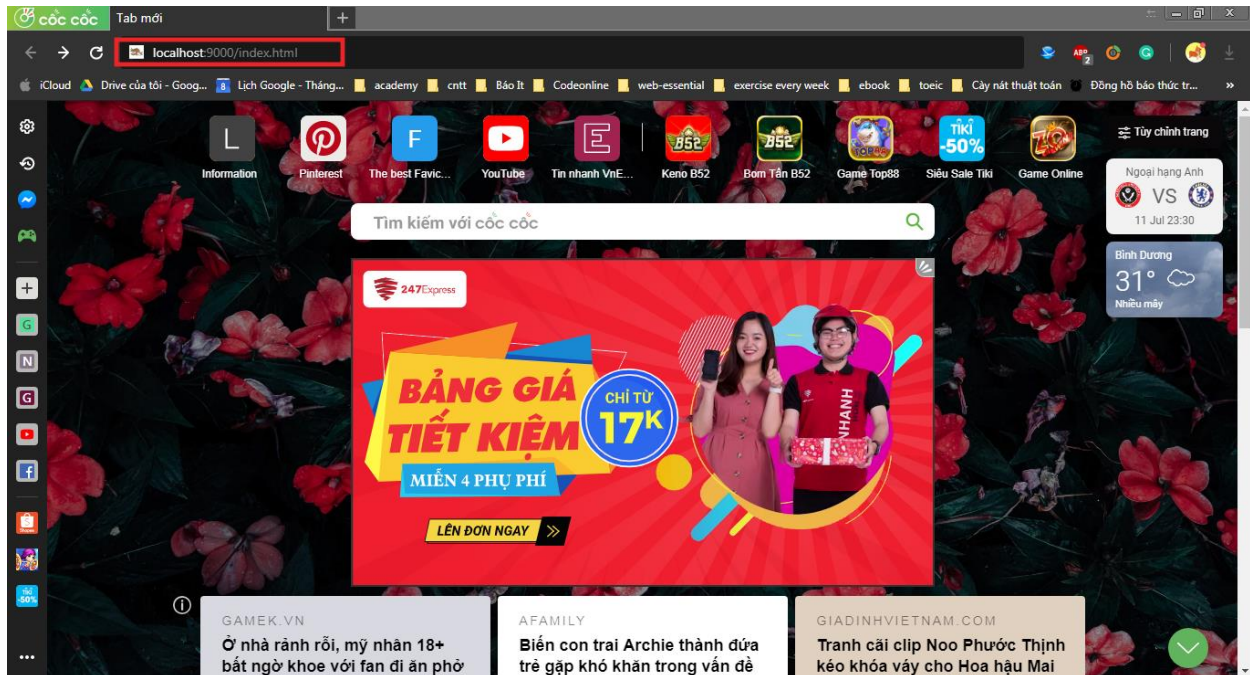
+ MethodPost(Body): sẽ lấy body message sau đó gọi hàm Account(Body) để lấy thông tin từ người dùng đã nhập. Kế tiếp gọi hàm CheckExist(account) xem tài khoản người dùng nhập có tồn tại trong CSDL(account.txt) của mình không. Nội dung gói tin sẽ có mã trạng thái 301(Moved Permanently) giúp điều hướng trang web và đi kèm theo Location: new_url. Nếu tài khoản có tồn tại thì new_url sẽ là trang info.html còn không sẽ là trang 404.html (Xử lý cho bài 2).



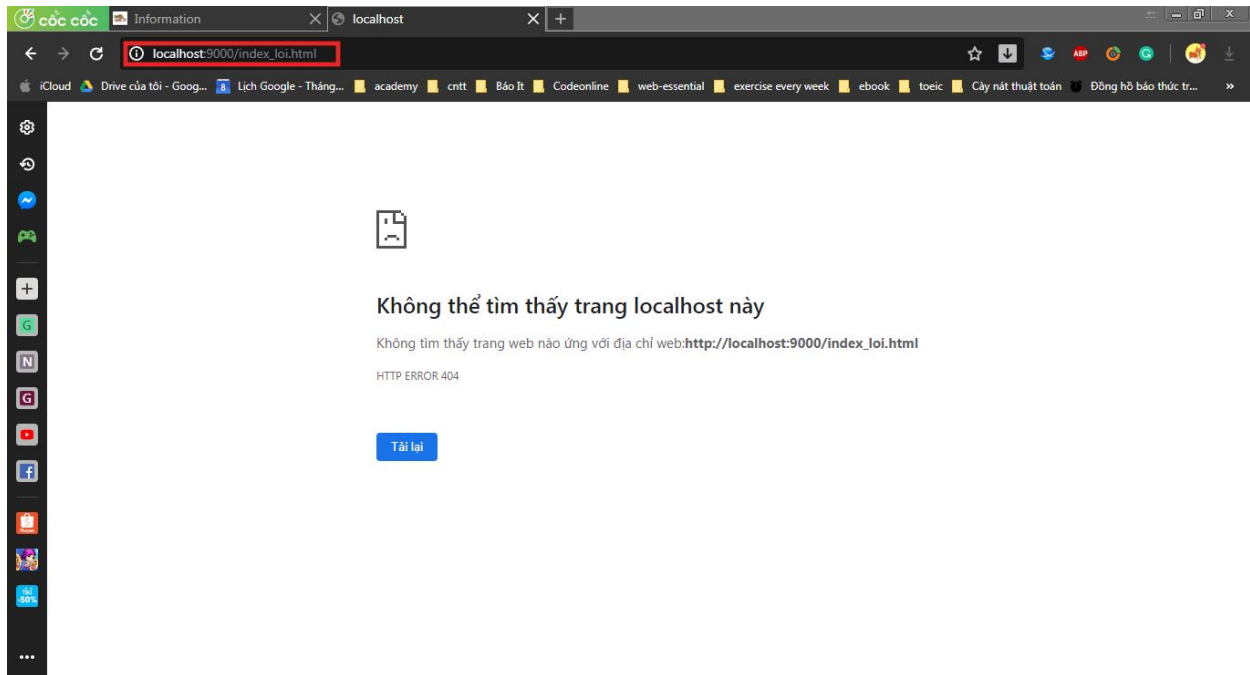
IV. CÁCH TEST ỨNG DỤNG

1. Khi nhập URL

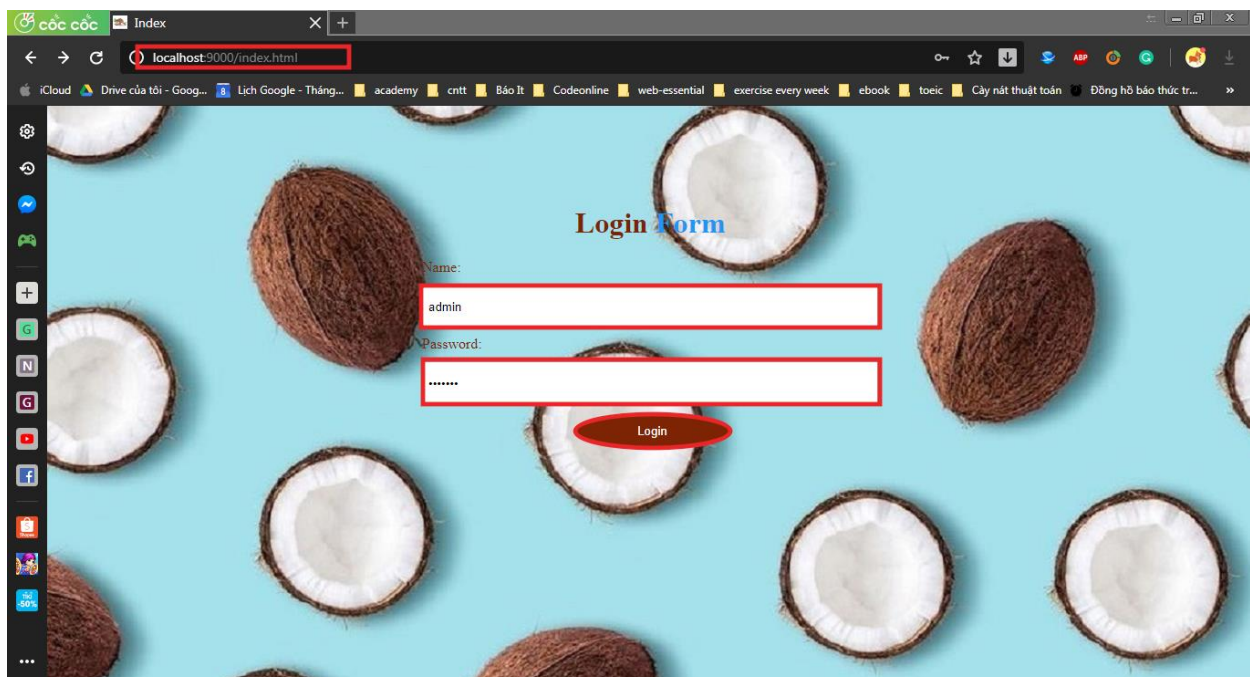
- Đường dẫn url: **host_name[:port_num][[/path]][/file_name]**
- Nếu chúng ta nhập đúng URL: localhost:9000/index.html



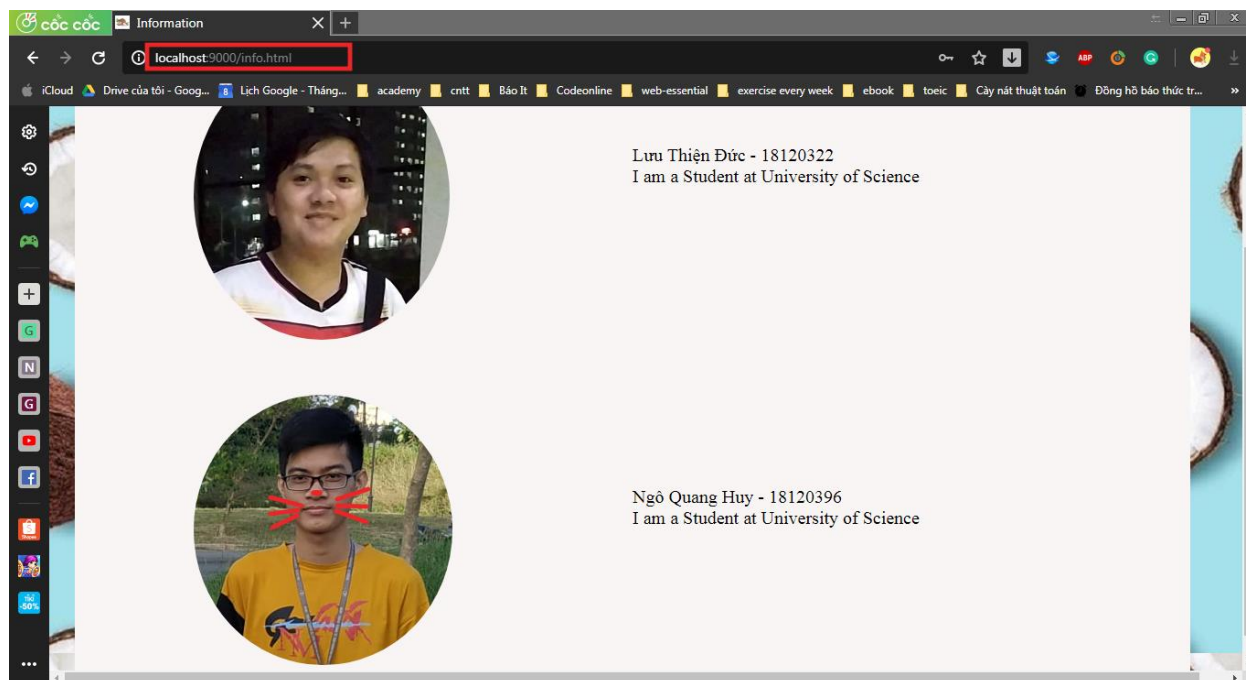
- Nếu nhập sai thì ta đã xử lý try... except nếu không tồn tại trang đấy thì trả về mã trạng thái 404. Trong ví dụ minh họa URL là:
localhost:9000/index_loi.html



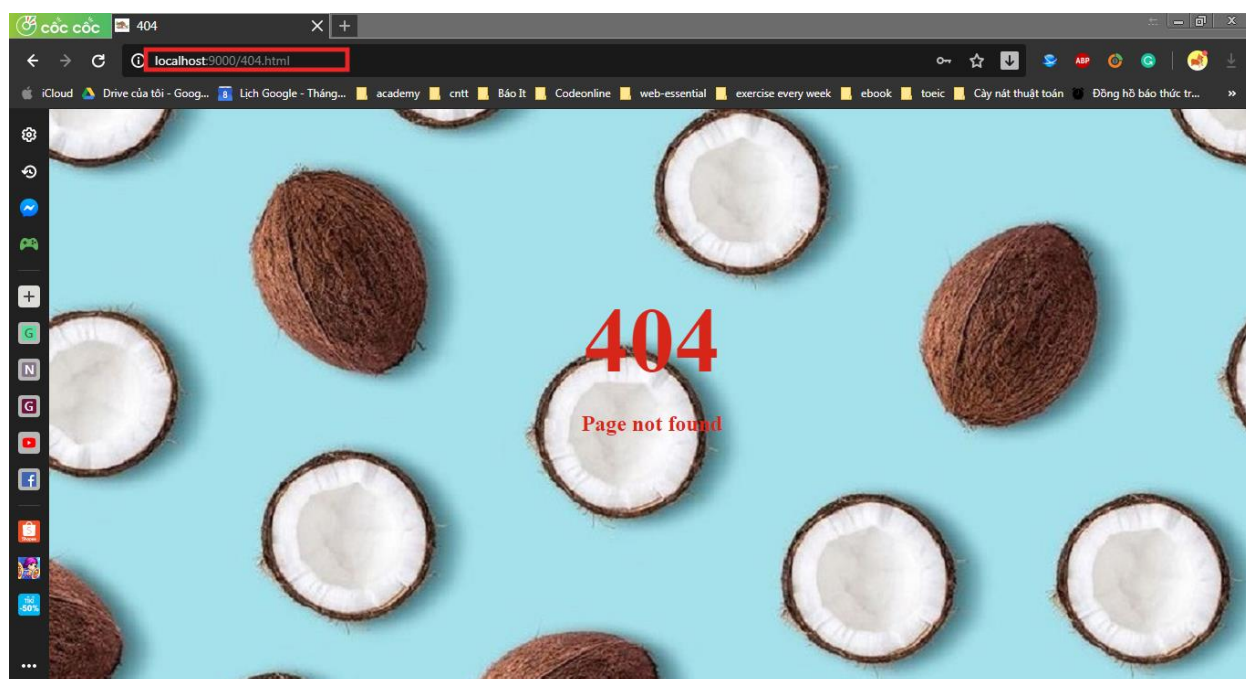
2. Khi đăng nhập



- Nếu ta nhập tài khoản user với password có tồn tại trong file account.txt thì trang index.html sẽ được điều hướng đến trang info.html



- Còn nếu nhập sai thì nó sẽ điều hướng đến trang 404.html



V. TỰ ĐÁNH GIÁ

- Đã hoàn thành đầy đủ yêu cầu trong đề án.
- Các tính năng thêm:
 - Đã thêm tính năng CSDL tài khoản của người dùng (nằm trong file account.txt).
 - Xử lý đồng thời trên nhiều client.

VI. THAM KHẢO

- Slide bài giảng – Môn mạng máy tính
- Tham khảo về redirect
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Redirects>
- Các hàm xử lý chuỗi trong python
<https://toidicode.com/cac-ham-xu-ly-chuoi-trong-python-368.html>