



Bộ môn Công nghệ Phần mềm
Viện CNTT & TT
Trường Đại học Bách Khoa Hà Nội

CÔNG NGHỆ WEB TIỀN TIẾN

Bài 00: Giới thiệu về khóa học



CN WEB TIỀN TIẾN

1. Giới thiệu về khóa học
2. Đề cương chi tiết
3. Tài liệu tham khảo
4. Đánh giá môn học
5. Thảo luận



CN WEB TIỀN TIẾN

1. **Giới thiệu về khóa học**
2. Đề cương chi tiết
3. Tài liệu tham khảo
4. Đánh giá môn học
5. Thảo luận



1. Giới thiệu về khóa học

- Mục đích môn học: Sau khi kết thúc môn học, sinh viên sẽ có thể:
 - Nắm vững các khái niệm liên quan đến công nghệ web: Internet, WWW, HTTP/HTTPS, HTML, CGI, URL, mô hình trang web web, kiến trúc n-tầng;
 - Phát triển trang web phía client sử dụng HTML, CSS, Javascript và Ajax;
 - Hiểu và biết cách sử dụng XML, XML schema và XHTML.
 - So sánh, đánh giá và có khả năng tự lựa chọn một công nghệ phát triển web phía server để phát triển trang web dựa trên kiến trúc MVC/3 lớp;
 - Cải tiến kỹ năng làm việc nhóm và kỹ năng trình bày.



1. Giới thiệu về khóa học

- **Điều kiện tiên quyết:** Sinh viên cần học môn OOP, Cơ sở dữ liệu, hiểu và nắm vững ngôn ngữ CN Javascript.
- **Thời lượng môn học:** 45 tiết lý thuyết và bài tập được phân bổ trong 15 tuần.



1. Giới thiệu về khóa học

■ **Nhiệm vụ của sinh viên:**

- Lên lớp đầy đủ.
- Phong cách CN, phong cách viết báo cáo sẽ được yêu cầu trong bài tập lớn, bài thi
- Hoàn thành các bài tập lớn
- Tự thực hành ở nhà
- Tự tìm hiểu thêm các kiến thức chuyên sâu



1. Giới thiệu về khóa học

■ **BÀI TẬP LỚN cung cấp API cho lớp IT4788:**

- Các API được nhóm hoàn thiện trong bài tập lớn sẽ được sử dụng cho các ứng dụng của lớp IT4788
- Mỗi nhóm của lớp IT4895 được phép cung cấp các API cho nhiều nhóm khác nhau của IT4788
- Chú ý **KHÔNG** cung cấp mã nguồn.
- Nhóm nào được nhiều nhóm của IT4788 sử dụng cũng như được phản hồi tốt sẽ được đánh giá cao.



CN WEB TIỀN TIẾN

1. Giới thiệu về khóa học
2. **Đề cương chi tiết**
3. Tài liệu tham khảo
4. Đánh giá môn học
5. Thảo luận



2. Đề cương chi tiết

- Tuần 1:
 - Giới thiệu chung về khóa học
 - Cơ bản về WWW
 - Cơ bản về HTML
- Tuần 2:
 - Tiếp tục về HTML
 - Cơ sở CSS



2. Đề cương chi tiết

- Tuần 3, 4:
 - Tiếp tục về CSS
 - XML
- Tuần 5:
 - XHTML
 - Javascript
- Tuần 6:
 - Tiếp tục về Javascript



2. Đề cương chi tiết

- Tuần 7:
 - Thực hành với Javascript
- Tuần 8:
 - DHTML
 - Web mobile
- Tuần 9:
 - NodeJS và ExpressJS
- Tuần 10:
 - MongoDB



2. Đề cương chi tiết

- Tuần 11:
 - Web socket
 - SSL
- Tuần 12:
 - Báo cáo tiến độ BTL
- Tuần 13, 14, 15:
 - Bảo vệ BTL
-



4. Đánh giá môn học

- Điểm quá trình
 - Điểm danh
 - Bài tập lớn
- Điểm kết thúc môn học
 - Thi trắc nghiệm + tự luận



4. Đánh giá môn học

- Yêu cầu bài tập lớn: tối đa 4 sinh viên / nhóm
 - Sinh viên đăng ký đề tài Bài tập lớn
 - Nộp báo cáo Bài tập lớn: báo cáo in
 - Link đến github chương trình
 - Bảo vệ bài tập lớn

5. Thảo luận

- Các câu hỏi của sinh viên liên quan đến nội dung môn học “Công nghệ Web tiên tiến”
- Các câu hỏi của sinh viên liên quan đến nội dung thi





Thông tin giảng viên

- Giảng viên: Nguyễn Mạnh Tuấn
- Bộ môn: Công nghệ phần mềm
- Mobile: 0977 045 184
- Email: tuannm@soict.hust.edu.vn



Thông tin giảng viên

- Giảng viên: Nguyễn Tiến Thành
- Bộ môn: Công nghệ phần mềm
- Mobile: 0987 768 886
- Email: nguyenthanh@soict.hust.edu.vn



Bộ môn Công nghệ Phần mềm
Viện CNTT & TT
Trường Đại học Bách Khoa Hà Nội

CN WEB TIỀN TIẾN

Bài 01: Kiến thức cơ bản về công nghệ WWW, HTML, CSS



Mục đích

- Cung cấp cho Sinh viên kiến thức về cơ chế hoạt động của trang web
 - Khái niệm siêu văn bản (hypertext)
 - Máy phục vụ, trình duyệt web, giao thức hoạt động

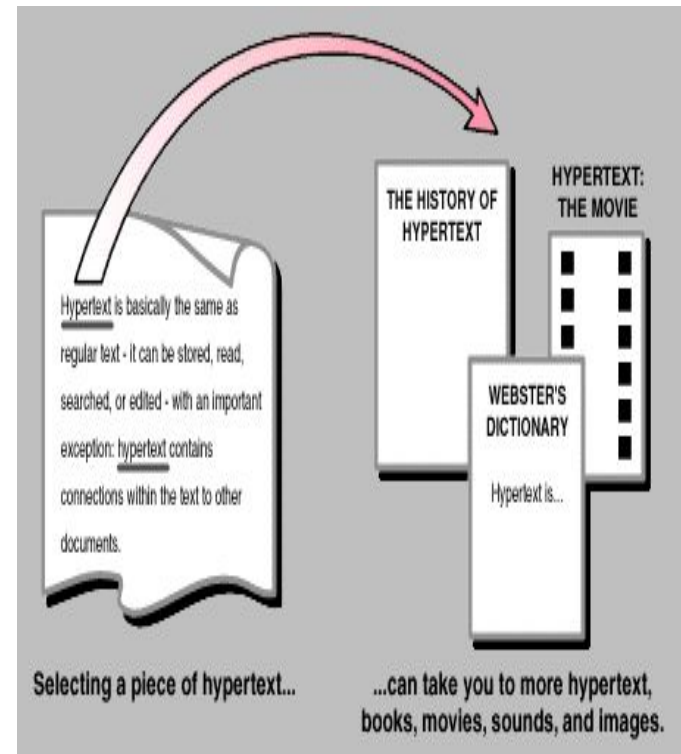


Nội dung

- 1. Khái niệm về siêu văn bản
- 2. Các thành phần của Web
 - 2.1. Máy phục vụ, máy khách
 - 2.2. WebServer, WebBrowser
 - 2.3. Website, WebPage
 - 2.4. URL
- 3. Phân loại trang Web
- 4. Một số điều cần chú ý trong phát triển Web

1. Khái niệm về siêu văn bản

- Hypertext? Hyperlink? Hypermedia?
- **Hypertext:** là văn bản (text) trên máy tính có sự *liên kết* đến một văn bản khác và có thể *truy cập trực tiếp* thông qua click chuột, nhấn phím.
- **Hyperlink:** là một liên kết từ một file hypertext đến một file khác
- **Hypermedia:** là sự mở rộng của hypertext, trong đó đối tượng có thể là văn bản, đồ họa, âm thanh, video...



2. Các thành phần của Web

- Web (World Wide Web)?

Web là hệ thống các tài liệu hypertext liên kết với nhau thông qua mạng Internet

- Các thành phần của Web

- Web Page
- Web Site
- Uniform Resource Locator
- Web Server
- Web Browser



2.1. Máy phục vụ, máy khách

Máy khách (Client)

- Máy khai thác dịch vụ của máy phục vụ
- Với mỗi dịch vụ, thường có các phần mềm chuyên biệt để khai thác
- Một máy tính có thể vừa là client vừa là server



2.1. Máy phục vụ, máy khách

Máy phục vụ (Server)

- Là máy chuyên cung cấp các dịch vụ và tài nguyên cho các máy tính khác
- Thường cài các phần mềm chuyên dụng
- Một máy phục vụ có thể dùng cho một hay nhiều mục đích.
 - *File server*
 - *Application server*
 - *Mail server*
 - *Web server*

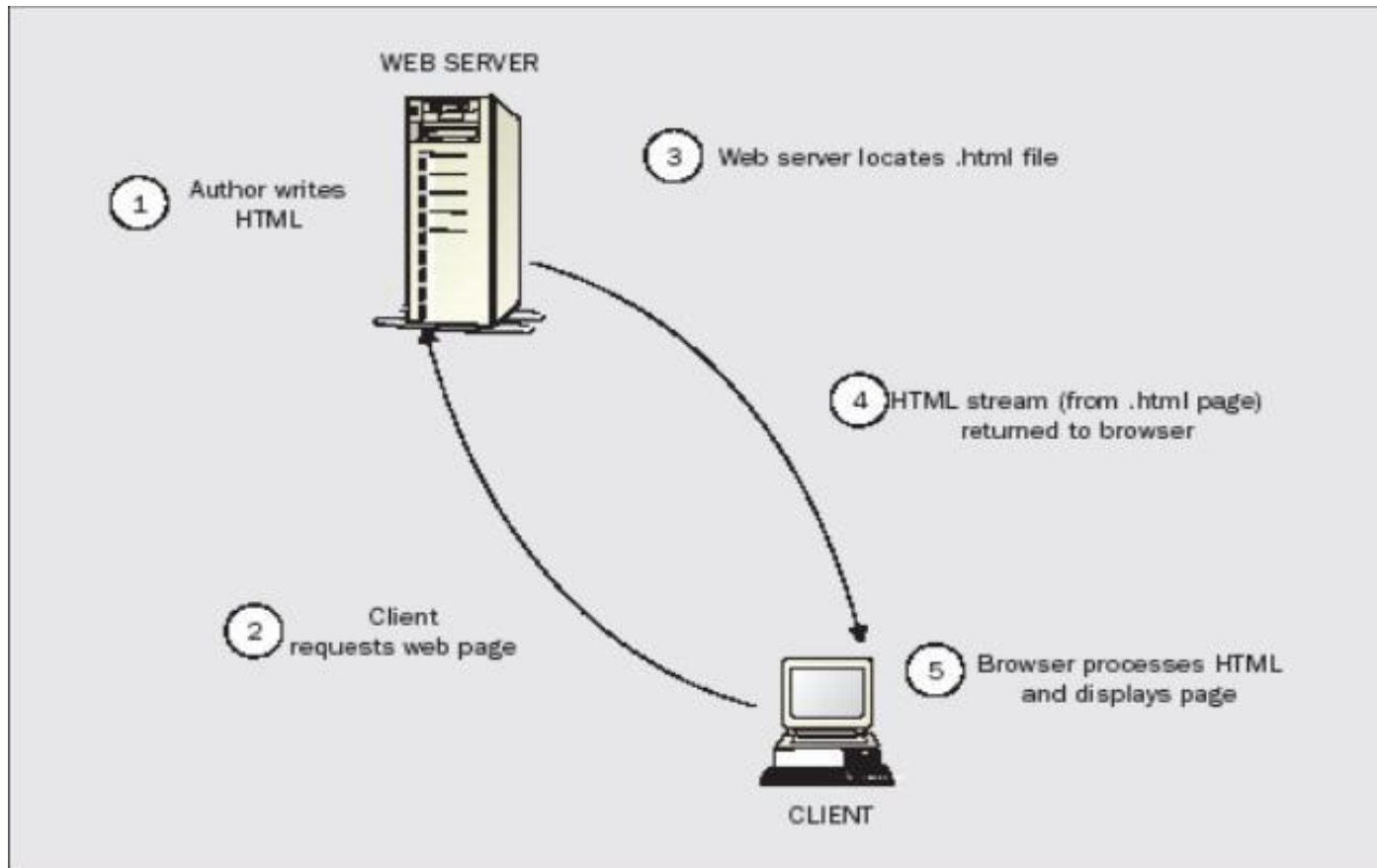


2.2. Web Server, Web Browser

- Web Server
 - Máy phục vụ web
 - Một số phần mềm web server chuyên dụng:
 - Apache, Internet Information Services (IIS)
- Web Browser
 - Là phần mềm chạy trên Client để khai thác dịch vụ Web



2.2. Web Server, Web Browser



Tương tác giữa web Server và web Browser

2.3. WebPage, WebSite

- WebPage:
 - Là một trang Web
 - Ngôn ngữ viết: PHP, ASP, ASP.Net... nhưng kết quả trả về cho Client là HTML
- Website:
 - Là tập hợp các WebPage có nội dung thống nhất



2.3. WebPage, Website

- Phân cấp trang Web



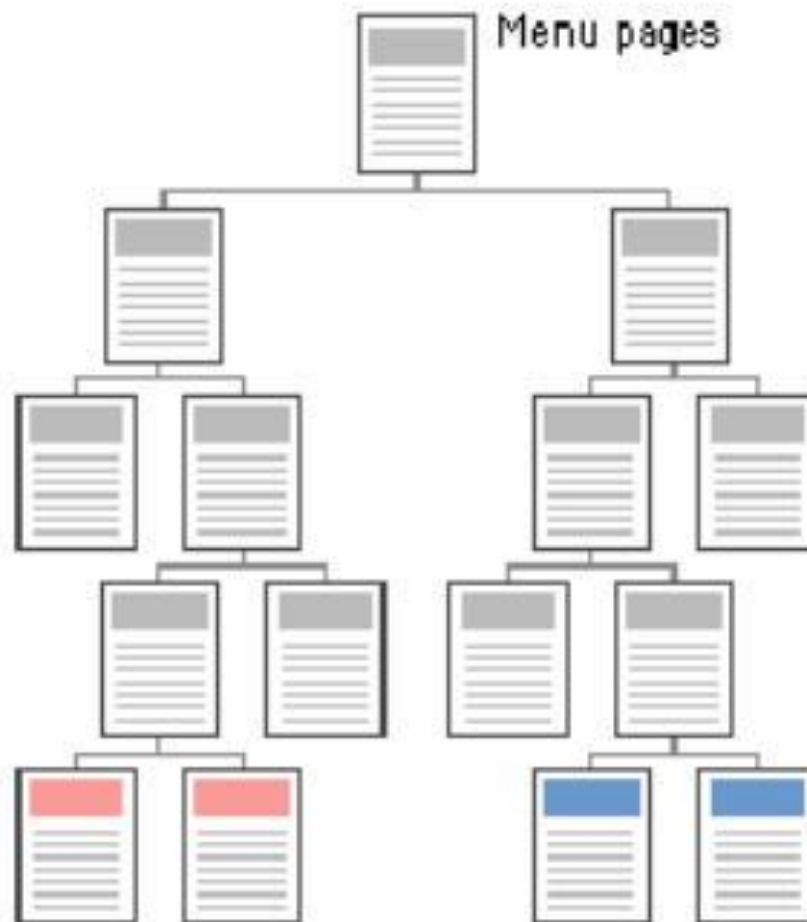
2.3. WebPage, Website

- Phân cấp quá “nông”



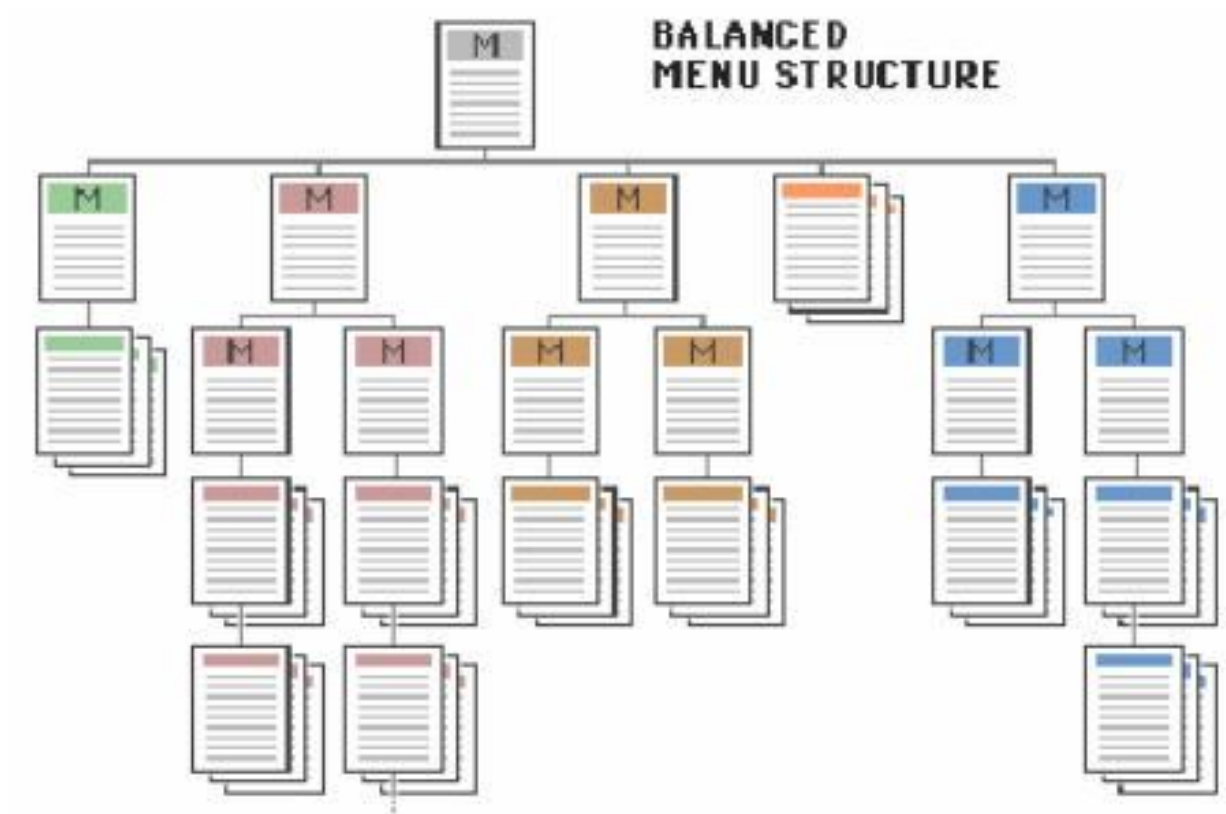
2.3. WebPage, Website

- Phân cấp quá “sâu”



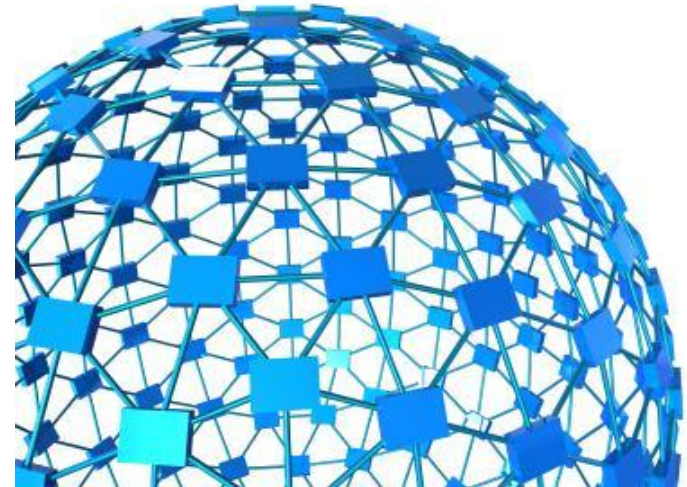
2.3. WebPage, Website

- Phân cấp hợp lý



2.4. URL (Uniform Resource Locator)

- Địa chỉ IP (IP Address)
 - Xác định một máy tính trong mạng dựa trên giao thức TCP/IP. Hai máy tính trong mạng có 2 địa chỉ IP khác nhau
 - Có dạng x.y.z.t ($0 \leq x, y, z, t \leq 255$)
 - Ví dụ: google Việt Nam?
74.125.127.106
 - Địa chỉ Loop Back: 127.0.0.1



2.4. URL (Uniform Resource Locator)

- Tên miền (Domain name)
 - Tại sao lại cần?
 - Là tên gắn liền với một địa chỉ IP
 - Ở dạng văn bản, thân thiện với người dùng
 - Các cấp tên miền?
 - Đánh số cấp từ bên phải, bắt đầu từ 1
 - Ex: www.soict.hut.edu.vn
 - Localhost: 127.0.0.1



2.4. URL (Uniform Resource Locator)

- Cổng dịch vụ (Service Port)
 - Tại sao lại cần?
 - Một Server có thể cung cấp nhiều dịch vụ => cần sử dụng cổng để xác định dịch vụ cung cấp
 - Mỗi dịch vụ thường chiếm những cổng mặc định
 - HTTP: 80, FTP: 21, SMTP: 25, POP3: 110



2.4. URL (Uniform Resource Locator)

- Là chuỗi định vị tài nguyên trên Internet
Ví dụ: `http://is.hut.vn:8080/hpsg/index`



- Cấu trúc

giao thức://địa chỉ máy:cổng/đường dẫn đến tài nguyên

- Nhiều tham số có giá trị mặc định => có thể bỏ qua



3. Phân loại trang Web

- Web tĩnh
 - Sử dụng HTML
 - Tương tác yếu
- Web động
 - Kết hợp mã HTML và mã lệnh
 - Tương tác mạnh
 - Mã lệnh được thực thi trên Server, trả về HTML cho Client



3. Phân loại trang Web

- So sánh
- Web tĩnh
 - Dễ phát triển
 - Nội dung thường cố định trên Browser
- Web động
 - Khó phát triển hơn
 - Có thể thay đổi nội dung trên Browser
 - Các ngôn ngữ sử dụng đa dạng: PHP, ASP, ASP.NET..



4. Một số điều cần chú ý trong phát triển Website

- Phân tích thiết kế HTTT
 - Nhận yêu cầu và tiến hành khảo sát
 - Đặc tả
 - Phân tích
 - Thiết kế
 - CN
 - Kiểm thử

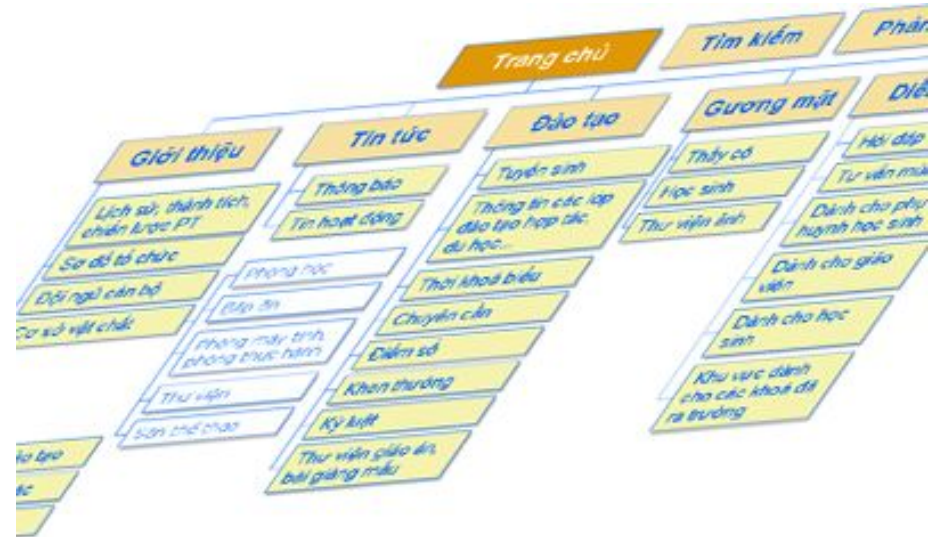
4. Một số điều cần chú ý trong phát triển Website

■ Đặc tả:

- Web để làm gì?
- Ai dùng?
- Trình độ người dùng?
- Nội dung, hình ảnh?

■ Phân tích

- Mỗi liên quan giữa các nội dung?
- Thứ tự các nội dung?



4. Một số điều cần chú ý trong phát triển Website

■ Thiết kế

- Sơ đồ cấu trúc website
- Giao diện
- Tĩnh hay động
- CSDL
- Nội dung từng trang
- Liên kết giữa các trang

■ Xây dựng

- Cấu trúc thư mục
- Các modul dùng chung
- ...

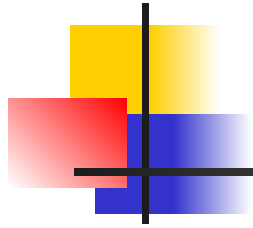


4. Một số điều cần chú ý trong phát triển Website

■ Kiểm thử

- Kiểm tra trên nhiều trình duyệt
- Kiểm tra trên nhiều loại mạng
- Kiểm tra tốc độ
- Kiểm tra các liên kết
- Thử các lỗi bảo mật
- ...





BÀI TẬP TUẦN 1



Mục lục

1. Yêu cầu tổng quan của đề tài
2. Quy trình bảo vệ
3. Yêu cầu chức năng
4. Cơ sở dữ liệu
5. Danh sách API
6. Đăng ký đăng nhập



Mục lục

1. **Yêu cầu tổng quan của đề tài**
2. Quy trình bảo vệ
3. Yêu cầu chức năng
4. Cơ sở dữ liệu
5. Danh sách API
6. Đăng ký đăng nhập



1. Yêu cầu tổng quan của đề tài

- Phát triển một hệ thống có client (trang web đa nền tảng), hệ thống database server, chat server và service server (tùy chọn công nghệ) để xây dựng lại (bắt chước một phần giao diện và một phần chức năng) một trang web phổ biến với người dùng Việt Nam.
- Trong học kỳ này, trang web được lựa chọn là Facebook. SV cần hoàn thành không những một trang web mà còn xây dựng (các) server hỗ trợ
 - Database lưu trữ các bài viết, thông tin của người dùng
 - Chat server phục vụ chức năng chat cho người dùng
 - Service server cung cấp các api để trang web di động truy cập
 - Các push notification được thực hiện bằng Firebase (hoặc SV tự phát triển)



1. Yêu cầu tổng quan của đề tài

Lịch trình giới thiệu API và giao diện của bài tập tuần:

Tuần 1: API Đăng ký/Đăng nhập

Tuần 2:

- API Đăng ký/Đăng nhập (tiếp theo)
- API đăng bài

Tuần 3:

- API truy vấn bài/sửa bài

Tuần 4:

- API xóa/báo cáo/thích/xem BL/đăng BL của bài

Tuần 5:

- API danh sách bài/kiểm tra phần tử mới



1. Yêu cầu tổng quan của đề tài

Lịch trình giới thiệu API và giao diện của bài tập tuần (tiếp):

Tuần 6:

- API tìm kiếm/xem tìm kiếm/xóa tìm kiếm

Tuần 7:

- API yêu cầu kết bạn/lấy DS video và bạn bè
-

Tuần 8:

- API xử lý yêu cầu/xem danh sách gợi ý/yêu cầu kết bạn và xem danh sách chặn

Tuần 9:

- API đổi mật khẩu và đọc/sửa cài đặt thông báo
- API chặn người dùng, kiểm tra phiên bản



Yêu cầu tổng quan của đề tài

Lịch trình giới thiệu API và giao diện của bài tập tuần (tiếp):

Tuần 10:

- API lấy/đọc thông báo
- API gán mã thiết bị
- API lấy/sửa thông tin cá nhân

Tuần 11:

- Các API cuối cùng phục vụ việc nhắn tin



Mục lục

1. Yêu cầu tổng quan của đề tài
2. **Quy trình bảo vệ**
3. Yêu cầu chức năng
4. Cơ sở dữ liệu
5. Danh sách API
6. Đăng ký đăng nhập



2. Quy trình bảo vệ

2.1. Thành viên

2.2. Thời gian bảo vệ

2.3. Hình thức bảo vệ



2. Quy trình bảo vệ

2.1. Thành viên

2.2. Thời gian bảo vệ

2.3. Hình thức bảo vệ



2.1. Thành viên

Mỗi nhóm cần tối đa năm sinh viên để thực hiện các công việc sau (không bắt buộc):

- + Một bạn phát triển Database server, chat server, service server để có các dữ liệu bài viết, thông tin cá nhân, lịch sử hoạt động của người dùng
- + Một bạn phát triển thư viện xử lý JSON trả về từ server
- + Một bạn phát triển giao diện và xử lý sự kiện cho trang web Facebook
- + Một bạn xử lý chat và push notification phía clients
- + Một bạn đảm nhiệm kiểm thử tổng hợp. Các thành viên có thể thực hiện kiểm thử chéo.



2. Quy trình bảo vệ

2.1. Thành viên

2.2. Thời gian bảo vệ

2.3. Hình thức bảo vệ



2.2. Thời gian bảo vệ

Chậm nhất là trước khi vào học **ngày thứ 3**, các sinh viên gửi thông tin về nhóm của mình cùng thông tin liên lạc của nhóm trưởng

Các danh sách bảo vệ sẽ được gửi cho lớp trước khi bảo vệ

Các nhóm đến bảo vệ theo đúng lịch của mình, với **đầy đủ** các thành viên.

Bảo vệ vào các buổi học **thứ 13, 14, 15** của môn học

Điểm bài tập lớn sẽ được tính vào **điểm quá trình** của môn học



Thời gian bảo vệ (2)

Các nhóm sẽ được bảo vệ **thử** vào buổi học thứ 12

Một số nhóm sẽ được chọn lựa để đi bảo vệ thử vào buổi thứ 12.

Các nhóm chỉ lên trình bày công việc mình làm được, nếu chưa làm được gì hoặc chưa hoàn thành xong hết hoặc không đến thì vẫn được chấp nhận.

Các nhóm trưởng khác **nên** có mặt để tham khảo

Các nhóm nên chuẩn bị như thể đây là một buổi bảo vệ chính thức, để có thể có kết quả tốt trong buổi tiếp theo



2. Quy trình bảo vệ

2.1. Thành viên

2.2. Thời gian bảo vệ

2.3. Hình thức bảo vệ



2.3. Hình thức bảo vệ

Sinh viên nộp báo cáo bản cứng vào hôm bảo vệ và bản mềm ba ngày trước khi nhóm bảo vệ. Trong báo cáo ghi rõ phân công công việc, link github của mã nguồn hai phía.

Nhóm sinh viên chuẩn bị đầy đủ ít nhất là 2 **máy tính** (kết nối mạng 3G/4G), server để demo. Không chấp nhận chỉ sử dụng máy ảo.

Khi một nhóm bảo vệ (các thành viên ngồi ở hai bàn đầu, dãy giữa), các nhóm khác ngồi cách xa **ít nhất hai bàn**

Bảo vệ xong, nhóm cần ở lại đến phút cuối của buổi học để có thể trả lời các câu hỏi khác (nếu có)



Mục lục

1. Yêu cầu tổng quan của đề tài
2. Quy trình bảo vệ
3. **Yêu cầu chức năng**
4. Cơ sở dữ liệu
5. Danh sách API
6. Đăng ký đăng nhập



3. Yêu cầu chức năng (1)

Hệ thống cần phát triển có các chức năng sau:

- + Cho phép người dùng đăng ký và đăng nhập vào hệ thống. Tên đăng nhập là số điện thoại, mật khẩu không phải là số điện thoại, không chứa các ký tự đặc biệt
- + Cho phép người dùng đăng bài mới với một chế độ duy nhất là public, số lượng từ giới hạn là 500 từ.
- + Bài viết của người dùng giới hạn tối đa **bốn** hình ảnh HOẶC **một** video.
- + Khi bài được đăng, bất kỳ ai cũng có thể xem, like, comment được. Số lượng từ trong một comment không vượt quá 500 từ.
- + Một tài khoản có thể kết bạn được với tối đa 500 bạn



3. Yêu cầu chức năng (2)

- + Một bài viết bất kỳ đều có chức năng cho phép người khác báo cáo
- + Một người có thể chặn (**block**) và bỏ chặn (**unblock**) bất kỳ ai.
Việc chặn sau khi bỏ chặn không bị giới hạn thời gian như trên facebook
- + Việc chặn một người sẽ khiến người đó không còn là bạn bè của người thực hiện thao tác chặn
- + Việc chặn một người không khiến **bình luận, like** của người đó bị xóa khỏi trang cá nhân
- + Việc chặn một người sẽ khiến hai người không thể trao đổi chat với nhau được nữa
- + Việc chặn một người sẽ khiến hai người không xem được trang cá nhân của nhau.



3. Yêu cầu chức năng (3)

- + Chỉ có một cảm xúc like duy nhất, không có nhiều cảm xúc như phiên bản Facebook hiện nay.
- + Hai người chỉ có thể chat với nhau nếu họ đã là bạn bè, nếu chưa là bạn bè thì người A sẽ không nhận được nút chat để trao đổi với người B.
- + Hiện tại chưa cho phép tiến hành lập nhóm chat.
- + Khi một người đăng nhập thành công vào trang web hệ thống sẽ hiển thị ưu tiên các bài viết của bạn bè trước. Nếu đã hiển thị xong các bài viết của bạn bè thì sẽ hiển thị ngẫu nhiên các bài của những người dùng khác. Ưu tiên sắp xếp theo thời gian. Tất nhiên không hiển thị bài của những người bị chặn
- + Chủ bài viết không xóa bình luận được. Chủ bình luận không sửa được bình luận.



3. Yêu cầu chức năng (4)

- + Không xem được danh sách những người thích một bài viết (nhưng biết được số lượng thích của bài viết).
- + Không bày tỏ cảm xúc cho bình luận được.
- + Bình luận chỉ chứa văn bản, HTTP link và/hoặc emoticon
- + Các thông tin được push: có người thích bài viết của bạn, có người bình luận bài viết của bạn, bạn bè đăng bài viết mới, có người gửi tin nhắn cho bạn, có người muốn kết bạn với bạn, tài khoản đăng nhập trên máy khác... (xem chi tiết bài tập tuần 9)



3. Yêu cầu chức năng (5)

- + Tải ảnh có kích thước tối đa 4MB, không làm việc với ảnh động
- + Tải video có kích thước tối đa 10MB, thời lượng không quá 10s và phải 1s trở lên.
- + Hoàn toàn có thể sửa, xóa bài viết. Nhưng một bài viết không chứa lẫn lộn cả ảnh lẫn video
- + Không có việc tham gia và tạo lập nhóm



3. Yêu cầu chức năng (6)

- + Không cho phép tag thành viên khác vào bài viết hoặc đoạn chat hoặc bình luận
- + Hệ thống cho phép việc tải ảnh và video
- + Hệ thống cho phép chèn các emoticon trong bài viết hoặc bình luận hoặc tin nhắn chat bằng cách nhập các ký tự đặc biệt
- +



3. Yêu cầu chức năng (7)

- + Cho phép tìm kiếm. Chẳng hạn tìm kiếm với từ khóa “đội bóng Nam Việt đá” thì vẫn ra được các kết quả về các bài viết có chứa các từ khóa “đội bóng Việt Nam đá”.
- + KHÔNG cho phép người dùng tự xóa tài khoản của mình
- + Xây dựng trang quản trị để người quản trị có thể dễ dàng thêm/sửa/xóa trực tiếp các dữ liệu cần thiết (về API Web thì các nhóm được quyền share cho nhau)



Mục lục

1. Yêu cầu tổng quan của đề tài
2. Quy trình bảo vệ
3. Yêu cầu chức năng
4. **Cơ sở dữ liệu**
5. Danh sách API
6. Đăng ký đăng nhập



4. Cơ sở dữ liệu

SV cần thiết kế ra được cơ sở dữ liệu phục vụ các dữ liệu liên quan đến (dự kiến):

- + Tài khoản cá nhân: id, tên, mật khẩu, số điện thoại, link avatar, token, danh sách id bị chặn bởi cá nhân
- + Bài viết cá nhân: id, id của chủ bài viết, nội dung bài viết, media, danh sách các id của bình luận, danh sách các id của người thích, thời điểm tạo
- + Bình luận: id, id của người viết bình luận, nội dung bình luận, thời điểm tạo
- + Đoạn chat: id của người A, id của người B, id của tin chat, nội dung chat, thời điểm tạo
- + Một số bảng cơ sở dữ liệu khác (phục vụ cho push)
- + Sinh viên được toàn quyền sử dụng loại CSDL nào hoặc tự đề xuất cấu trúc của bảng



4. Cơ sở dữ liệu

SV cần thiết kế ra được cơ sở dữ liệu phục vụ các dữ liệu liên quan đến (dự kiến):

- + Tài khoản cá nhân: id, tên, mật khẩu, số điện thoại, link avatar, token, danh sách id bị chặn bởi cá nhân
- + Bài viết (id, tiêu đề, nội dung, ảnh, video, thời gian đăng, người đăng, danh sách cá nhân tạo)
- + Bình luận (id, nội dung, người đăng, thời gian đăng, bài viết liên quan)
- + Đoạn clip (id, nội dung, người đăng, thời gian đăng, bài viết liên quan)
- + Một số bảng cơ sở dữ liệu khác (phục vụ cho push)
- + Sinh viên được toàn quyền sử dụng loại CSDL nào hoặc tự đề xuất cấu trúc của bảng
- +

CÂU HỎI 2: SINH VIÊN CÓ ĐỀ XUẤT THIẾT KẾ CƠ SỞ DỮ LIỆU NÀO KHÁC KHÔNG?



Mục lục

1. Yêu cầu tổng quan của đề tài
2. Quy trình bảo vệ
3. Yêu cầu chức năng
4. Cơ sở dữ liệu
- 5. Danh sách API**
6. Đăng ký đăng nhập



5. Danh sách API

Danh sách 41 API cần được phát triển:

Login, logout, signup, get_verify_code, check_verify_code, change_info_after_signup, get_list_posts, get_post, add_post, edit_post, delete_post, get_comment, set_comment, report_post, like, search, get_saved_search, del_saved_search, get_user_friends, get_user_info, set_user_info, get_list_videos, get_list_blocks, set_block, set_accept_friend, get_requested_friends, set_request_friend, get_push_settings, set_push_settings, change_password, check_new_version, set_devtoken, get_conversation, delete_message, get_list_conversation, delete_conversation, get_list_suggested_friends, check_new_item, get_notification, set_read_message, set_read_notification,



5. Đa

CÂU HỎI 3: NẾU trang web CÓ TRUY VẤN DỮ LIỆU TỪ SERVER, THÌ NÓ CÓ ƯU ĐIỂM GÌ?

Danh sách 41 API cần được phát triển:

Login, logout, signup, get_verify_code, check_verify_code, change_info_after_signup, get_list_posts, get_post, add_post, edit_post, delete_post, get_comment, set_comment, report_post, like, search, get_saved_search, del_saved_search, get_user_friends, get_user_info, set_user_info, get_list_videos, get_list_blocks, set_block, set_accept_friend, get_requested_friends, set_request_friend, get_push_settings, set_push_settings, change_password, check_new_version, set_devtoken, get_conversation, delete_message, get_list_conversation, delete_conversation, get_list_suggested_friends, check_new_item, get_notification, set_read_message, set_read_notification,



RESTfull API

Đường dẫn cơ bản:

<https://ABC.def/it4788/>

Tên miền ABC.def sinh viên tự thiết lập,
cho phép local

Với việc sử dụng API Login thì truy cập

<https://ABC.def/it4788/login?>... với các
tham số đi kèm



DANH SÁCH MÃ RESPONSE

| Code | Message | Note |
|------|---|--|
| 1000 | OK | |
| 9992 | Post is not existed | Bài viết không tồn tại |
| 9993 | Code verify is incorrect | Mã xác thực không đúng |
| 9994 | No Data or end of list data | Không có dữ liệu hoặc không còn dữ liệu |
| 9995 | User is not validated | Không có người dùng này |
| 9996 | User existed. | Người dùng đã tồn tại |
| 9997 | Method is invalid | Phương thức không đúng |
| 9998 | Token is invalid. | Sai token |
| 9999 | Exception error. | Lỗi exception |
| 1001 | Can not connect to DB. | Lỗi mất kết nối DB/hoac thuc thi cau SQL |
| 1002 | Parameter is not enough. | Số lượng Paramater không đầy đủ |
| 1003 | Parameter type is invalid. | Kiểu tham số không đúng đắn. |
| 1004 | Parameter value is invalid. | Giá trị của tham số không hợp lệ |
| 1005 | Unknown error. | Unknown error |
| 1006 | File size is too big. | cỡ file vượt mức cho phép |
| 1007 | Upload File Failed!. | upload thất bại |
| 1008 | Maximum number of images. | số lượng images vượt quá quy định |
| 1009 | Not access. | không có quyền truy cập tài nguyên |
| 1010 | action has been done previously by this user. | hành động đã được người dùng thực hiện trước đây |



DANH SÁCH MÃ RESPONSE (2)

1011 | Could not publish this post (Bài đăng vi phạm tiêu chuẩn cộng đồng)

1012 | Limited access (Bài đăng bị giới hạn ở một số quốc gia)



Mục lục

1. Yêu cầu tổng quan của đề tài
2. Quy trình bảo vệ
3. Yêu cầu chức năng
4. Cơ sở dữ liệu
5. Danh sách API
6. **Đăng ký đăng nhập**



6. Đăng ký đăng nhập

6.1. Signup

6.2. Login

6.3. Logout

6.4. Get_verify_code

Signup

O: bắt buộc, X: không bắt buộc

Input

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|---------------|
| 1 | phonenumber | string | O | |
| 2 | password | string | O | password |
| 3 | uuid | string | O | UUID thiết bị |

Output

Loại dữ liệu trả về là JSON

| NO | Tên parameter | Type | Mô tả |
|----|---------------|--------|----------------------------------|
| 1 | code | string | (tham khóa phần response common) |
| 2 | message | string | |
| 3 | data | | |



Mô tả signup

API thực hiện việc cho phép đăng ký một tài khoản mới của người dùng

Request dạng POST

Tham số: **phonenumber** (số điện thoại của người dùng), **password** (mật khẩu của người dùng), **uuid** (mã số định danh của thiết bị)

Kết quả đầu ra: Nếu đăng ký thành công thì trả về thông báo phù hợp đi kèm với việc sinh ra mã xác thực lưu ở server. Sẽ báo lỗi nếu có các trường hợp bất thường xảy ra.



Các test case cho Signup

1. Người dùng nhập số điện thoại vốn là số chưa được đăng ký trên hệ thống. Nhập đúng mật khẩu là xâu không trùng với số điện thoại, không chứa ký tự đặc biệt.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) kèm với mã xác thực được lưu ở phía server

Mã này đảm bảo không trùng nhau với hai số khác nhau, gồm cả số kèm chữ, có 6 ký tự

Chú ý: số điện thoại phải đủ 10 số, có số 0 ở đầu tiên, chấp nhận số cố định

Mật khẩu phải có từ 6 đến 10 ký tự.



Các test case cho Signup (2)

2. Người dùng nhập số điện thoại vốn là số điện thoại đã được đăng ký thành công từ trước. Người dùng cũng nhập mật khẩu đúng quy định

Kết quả mong đợi: 9996 | User existed

3. Người dùng nhập số điện thoại không đúng định dạng (không đủ số hoặc thừa số hoặc không có số 0 ở đầu tiên) và nhập mật khẩu đúng quy định

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của số điện thoại. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.



Các test case cho Signup (3)

4. Người dùng nhập đúng định dạng của số điện thoại nhưng mật khẩu không đúng định dạng (quá ngắn hoặc quá dài hoặc chứa ký tự đặc biệt hoặc trùng với số điện thoại). trang web cần phải kiểm tra ngay trước khi gửi dữ liệu lên server

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của mật khẩu. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.

5. Người dùng bỏ qua không nhập cả số điện thoại và mật khẩu nhưng nhấn vào nút “Đăng ký” (hoặc “Sign up” tùy vào giao diện trang web).

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của số điện thoại. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.

Login

O: bắt buộc, X: không bắt buộc

Input

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|-------|
| 1 | phonenumber | string | 0 | |
| 2 | password | string | 0 | |

Output

| NO | Tên parameter | Loại dữ liệu trả về là JSON | Type | NN | Mô tả |
|----|---------------|-----------------------------|--------|----|------------------------------------|
| 1 | code | | string | 0 | (tham khóa phần response common) |
| 2 | message | | string | 0 | |
| 3 | data | | array | 0 | Thông tin user |
| | id | | string | 0 | |
| | username | | string | 0 | |
| | token | | string | 0 | |
| | avatar | | string | 0 | -1: dời đổi tên và ava; 1: actived |
| | active | | | | |



Mô tả Login

API thực hiện việc cho phép đăng nhập vào một tài khoản của người dùng
Request dạng POST

Tham số: **phonenumber** (số điện thoại của người dùng), **password** (mật khẩu của người dùng). **THÊM TRƯỜNG DEVICE ID** truyền lên nữa

Kết quả đầu ra: Nếu đăng nhập thành công thì (i) nhận thông báo thành công và (ii) chuyển ngay sang màn hình trang chủ. Sẽ báo lỗi nếu có các trường hợp bất thường xảy ra.

Với id (định danh người dùng), username (tên người), token (mã phiên đăng nhập), avatar (link đến avatar), ~~active~~ (hiện tại trường này không sử dụng)



Mô tả Login

API thực hiện việc cho phép đăng nhập vào một tài khoản của người dùng
Request dạng POST

Tham số: **phonenumber** (số điện thoại của người dùng), **password** (mật khẩu của người dùng). **THÊM TRƯỜNG DEVICE ID** truyền lên nữa

Kết quả đầu ra: Nếu đăng nhập thành công thì (i) nhận thông báo thành công
trở về
Với
đăng
dùng)

CÂU HỎI 4: Mã TOKEN sẽ khác nhau nếu một người đăng nhập cùng một tài khoản trên các máy khác nhau. Theo em tại sao lại cần mã token này?



Mô tả Login

API thực hiện việc cho phép đăng nhập vào một tài khoản của người dùng

Request dạng POST

Tham số: **phonenumber** (số điện thoại của người dùng), **password** (mật khẩu)

Kết quả trả về: **CÂU HỎI 5: Sẽ gửi về kết quả gì cho trang web nếu như người dùng vẫn sử dụng avatar mặc định?**

nếu có các trường hợp bất thường xảy ra.

Với id (định danh người dùng), username (tên người), token (mã phiên đăng nhập), avatar (link đến avatar), ~~active~~ (hiện tại trường này không sử dụng)



Các test case cho Login

1. Người dùng nhập số điện thoại vốn là số đã được đăng ký trên hệ thống. Nhập đúng mật khẩu và xâu không trùng với số điện thoại, không chứa ký tự đặc biệt.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và trang web sẽ chuyển sang màn hình trang chủ. Chú ý, nếu ĐÃ đăng nhập trên máy khác thì (nếu chức năng push hoàn thành và máy khác có mạng) máy khác sẽ bị đăng xuất.

Mã token được lưu trữ ở cả client và server

2. Người dùng nhập số điện thoại vốn là số CHƯA được đăng ký trên hệ thống. Nhập mật khẩu đúng định dạng



Các test case cho Login (2)

Kết quả mong đợi:

**CÂU HỎI 6: THEO EM KẾT QUẢ
MONG ĐỢI LÀ GÌ?**

3. Người dùng nhập số điện thoại không đúng định dạng (không đủ số hoặc thừa số hoặc không có số 0 ở đầu tiên) và nhập mật khẩu đúng quy định

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của số điện thoại. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.



Các test case cho Login (3)

4. Người dùng nhập đúng định dạng của số điện thoại nhưng mật khẩu không đúng định dạng (quá ngắn hoặc quá dài hoặc chứa ký tự đặc biệt hoặc trùng với số điện thoại). trang web cần phải kiểm tra ngay trước khi gửi dữ liệu lên server

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của mật khẩu. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.

5. Người dùng bỏ qua không nhập cả số điện thoại và mật khẩu nhưng nhấn vào nút “Đăng nhập” (hoặc “Log in” tùy vào giao diện trang web).

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo sai định dạng của số điện thoại. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.



Các test case cho Login (4)

6. Người dùng nhập đúng định dạng của số điện thoại và mật khẩu nhưng không có kết nối mạng.

Kết quả mong đợi: trang web cần phải đưa ra thông báo càng sớm càng tốt về việc “Không thể kết nối Internet”. Mẹo: Kể cả khi có kết nối mạng Internet nhưng server không trả về kết quả hoặc kết quả không đúng định dạng JSON thì vẫn thông báo là “Không thể kết nối Internet”.

7. Người dùng nhập số điện thoại và mật khẩu có nội dung giống số điện thoại rồi nhấn vào nút “Đăng nhập” (hoặc “Log in” tùy vào giao diện trang web).

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo “Mật khẩu sai định dạng”. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.

Các test case cho Login (4)

6. Người dùng nhập đúng định dạng của số điện thoại và mật khẩu nhưng không có kết nối mạng.

Kết quả mong đợi: trang web cần phải đưa ra thông báo càng sớm càng tốt về việc “Không thể kết nối Internet”. Mẹo: Kể cả khi có kết quả kiểm tra kết nối mạng, trang web vẫn phải kiểm tra kết nối Internet trước khi cho phép đăng nhập.

CÂU HỎI 7: THEO EM VIỆC LIÊN TỤC KIỂM TRA DỮ LIỆU TRƯỚC Ở PHÍA trang web SẼ CÓ ƯU ĐIỂM GÌ?

7. Người dùng nhập số điện thoại và mật khẩu có nội dung giống số điện thoại rồi nhấn vào nút “Đăng nhập” (hoặc “Log in” tùy vào giao diện trang web).

Kết quả mong đợi: trang web kiểm tra ngay và đưa ra thông báo “Mật khẩu sai định dạng”. Kể cả khi không có mạng Internet thì trang web vẫn kiểm tra và đưa thông báo.



Các test case cho Login (5)

8. Người dùng đăng nhập trên máy A. Rồi đăng nhập trên máy B.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và token cũ (của lần đăng nhập trên máy A) sẽ bị xóa đi bên phía server.



Logout

Input

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|----------------|
| 1 | token | string | 0 | token từ sever |

Output

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|----------------------------------|
| 1 | code | string | 0 | (tham khóa phần response common) |
| 2 | message | string | 0 | |



Mô tả Logout

API thực hiện việc cho phép đăng xuất khỏi một tài khoản của người dùng

Request dạng POST

Tham số: **token** (mã phiên đăng nhập của người dùng)

Kết quả đầu ra: Nếu đăng xuất thành công thì (i) nhận thông báo thành công và (ii) chuyển ngay sang màn hình đăng nhập. Sẽ báo lỗi nếu có các trường hợp bất thường xảy ra.



Các test case cho Logout

1. Người dùng chỉ đăng nhập trên một thiết bị và sau đó đăng xuất.
Kết quả mong đợi: 1000 | OK (Thông báo thành công) và trang web sẽ chuyển sang màn hình đăng nhập. Token cũ sẽ bị xóa đi
2. Người dùng đăng nhập trên máy A, sau đó tắt mạng. Rồi đăng nhập trên máy B. Tiếp theo mở mạng của máy A ra và (push chưa kịp chuyển về máy A hoặc nhóm không làm push) đăng xuất.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và trang web sẽ chuyển sang màn hình đăng nhập. Tất cả token sẽ bị xóa hết



Các test case cho Logout (2)

3. Người dùng đăng nhập trên máy A, sau đó tắt mạng. Rồi đăng nhập trên máy B, sau đó đăng xuất trên máy B. Tiếp theo mở mạng của máy A ra và (push chưa kịp chuyển về máy A hoặc nhóm không làm push) đăng xuất.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và trang web ở máy A sẽ chuyển sang màn hình đăng nhập. Tất cả token sẽ bị xóa hết ở cả 2 phía

4. Người dùng đăng nhập trên máy A và ngắt kết nối mạng. Rồi đăng nhập trên máy B, rồi đăng xuất trên máy B.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và không còn bất kỳ trường token nào dành cho người dùng này ở cả hai phía



get_verify_code

Input

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|-------|
| 1 | phonenumber | string | 0 | |
| | | | | |

Output

| NO | Tên parameter | Type | NN | Mô tả |
|----|---------------|--------|----|----------------------------------|
| 1 | code | string | 0 | (tham khóa phần response common) |
| 2 | message | string | 0 | |



Mô tả get_verify_code

API thực hiện việc nhận mã xác thực cho một tài khoản của người dùng

Request dạng POST

Tham số: **phonenumber**

Kết quả đầu ra: nếu nhận thành công thì sẽ có mã xác thực được tạo ra phía server. Server sẽ lưu trữ mã này. Nếu không thành công thì sẽ có các thông báo lỗi tương ứng



Các test case cho get_verify_code

1. Người dùng gọi request này lại khi mà chưa nhận được hoặc làm mất mã xác thực.

Kết quả mong đợi: 1000 | OK (Thông báo thành công) và mã xác thực cũ sẽ vẫn được tồn tại

2. Người dùng liên tục gửi request này lại trong thời gian quá ngắn.

Kết quả mong đợi: Nếu dưới 120 giây sau khi đã gửi request này thì trang web sẽ làm mờ nút gửi mã xác thực. Nếu một lý do nào đó request này đến được server thì sẽ có thông báo trả về mã 1010 hoặc 1009



Các test case cho get_verify_code (2)

3. Người dùng truyền tham số với số điện thoại đã được hoàn tất quy trình đăng ký (tức việc xác thực cũng đã xong).

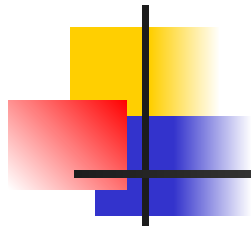
Kết quả mong đợi: mã lỗi 1010

4. Người dùng truyền tham số với số điện thoại chưa được đăng ký.

Kết quả mong đợi: trả về mã lỗi 1004 hoặc 9995

5. Người dùng truyền tham số với số điện thoại sai định dạng.

Kết quả mong đợi: trang web sẽ phải kiểm tra từ trước để thông báo lỗi không đúng định dạng. Nếu như được gửi lên server sẽ được nhận về mã 1004



HẾT TUẦN 1