**ĐẠI HỌC ĐÀ NẴNG**

**ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN HỆ ĐIỀU HÀNH VÀ MẠNG MÁY TÍNH**

**Đề tài :**

**XÂY DỰNG ỨNG DỤNG DI DỘNG GỬI ẢNH LÊN SERVER ĐỂ QUẢN LÍ**

**SINH VIÊN THỰC HIỆN:**

**Trần Huy Tùng LỚP: 18TCLC\_DT1 NHÓM: 18N13B-N10**

**Nguyễn Quang Huy LỚP: 18TCLC\_DT1 NHÓM: 18N13B-N10**

**Nguyễn Duy Thương LỚP: 18TCLC\_DT1 NHÓM: 18N13B-N10**

**GVHD: PGS.TS Nguyễn Tấn Khôi**

**Đà Nẵng, Tháng 12-2020**

**LỜI MỞ ĐẦU**

Lập trình mạng là môn học cung cấp những kiến thức như cơ cấu tổ chức, cách trao đổi dữ liệu giữa các máy tính với nhau và lập trình mạng nhằm giúp sinh viên nâng cao hiểu biết cũng như khả năng lập trình về lĩnh vực trên. Việc nghiên cứu và hoàn thành đồ án cơ sở ngành mạng giúp sinh viên tích lũy được những kiến thức sâu hơn về lập trình mạng nhằm phục vụ cho việc học tập cũng như công việc sau này.

Đồ án khó tránh khỏi còn nhiều sai sót, mong thầy cô thông cảm, hết lòng góp ý và chỉ cho chúng em những ưu, khuyết điểm để giúp chúng em có thể hoàn thiện tốt hơn sau này.

Em xin gởi lời cảm ơn đến các thầy cô trong khoa công nghệ thông tin, đặc biệt là thầy Nguyễn Tấn Khôi đã giúp em hoàn thành được đồ án này.

# 

# **MỤC LỤC**

[LỜI MỞ ĐẦU 2](#_heading=h.1baon6m)

[DANH MỤC HÌNH VẼ 6](#_heading=h.3vac5uf)

[1.](#_heading=h.2afmg28) GIỚI THIỆU ĐỀ TÀI 7

[1.1 Mục tiêu: 8](#_heading=h.pkwqa1)

[2.](#_heading=h.39kk8xu) CƠ SỞ LÝ THUYẾT 9

[2.1.1](#_heading=h.1opuj5n) Họ giao thức TCP/IP: 9

[2.1.2](#_heading=h.48pi1tg) Tầng liên kết: 10

[2.1.3](#_heading=h.2nusc19) Tầng Internet: 10

[2.1.4](#_heading=h.1302m92) Tầng giao vận: 10

[2.1.5](#_heading=h.3mzq4wv) Tầng ứng dụng: 10

[2.2](#_heading=h.2250f4o) Giao Thức HTTP: 12

[2.2.1](#_heading=h.haapch) Khái niệm : 12

[2.2.2](#_heading=h.319y80a) So sánh giữa HTTP và HTTPS : 12

[2.2.3](#_heading=h.1gf8i83) Cơ chế hoạt động của HTTP : 13

[2.2.4](#_heading=h.40ew0vw) Kết nối HTTP : 13

[2.2.5](#_heading=h.2fk6b3p) Uniform Resource Locators (URL) : 15

[2.2.6](#_heading=h.upglbi) HTTP Request Methods : 15

[2.2.7](#_heading=h.3ep43zb) HTTP Responses Methods : 16

[2.3](#_heading=h.1tuee74) Cơ sở dữ liệu MySQL: 16

[2.3.1](#_heading=h.4du1wux) Khái niệm : 16

[2.3.2](#_heading=h.2szc72q) Cách thức hoạt động của MySQL: 17

[2.3.3](#_heading=h.184mhaj) Các tính năng cốt lõi của MySQL: 18

[2.3.4](#_heading=h.3s49zyc) Sự khác biệt giữa MySQL và SQL: 18

[2.4](#_heading=h.279ka65) Ngôn ngữ lập trình Java: 19

[2.4.1](#_heading=h.meukdy) Khái niệm: 19

[2.4.2](#_heading=h.36ei31r) Các tính năng của Java: 19

[2.4.3](#_heading=h.1ljsd9k) Tác Dụng của Java: 20

[2.4.4](#_heading=h.45jfvxd) Các kiểu Java App và Java Platforms: 21

[Java App: 21](#_heading=h.2koq656)

[Standalone App 21](#_heading=h.zu0gcz)

[Web App 21](#_heading=h.3jtnz0s)

[Enterprise App 21](#_heading=h.1yyy98l)

[Mobile App 21](#_heading=h.4iylrwe)

[Java Platforms 21](#_heading=h.2y3w247)

[Java SE (Java Standard Edition) 21](#_heading=h.1d96cc0)

[Java EE (Java Enterprise Edition) 21](#_heading=h.3x8tuzt)

[Java ME (Java Micro Edition) 21](#_heading=h.2ce457m)

[JavaFX 21](#_heading=h.rjefff)

[2.5](#_heading=h.3bj1y38) Ngôn ngữ lập trình PHP: 22

[2.5.1](#_heading=h.1qoc8b1) Khái niệm: 22

[2.5.2](#_heading=h.4anzqyu) Tác dụng của PHP: 22

[2.5.3](#_heading=h.2pta16n) Ưu và Nhược Điểm của ngôn ngữ PHP: 23

[2.6 Volley library Android 24](#_heading=h.46r0co2)

[2.6.1 Khái Niệm 24](#_heading=h.2lwamvv)

[2.6.2 Tính Năng Nổi Bật 24](#_heading=h.111kx3o)

[2.6.3 Các class sử dụng trong Volley: 25](#_heading=h.3l18frh)

[3.](#_heading=h.14ykbeg) PHÂN TÍCH THIẾT KẾ HỆ THỐNG 25

[3.1](#_heading=h.3oy7u29) Phát Biểu Bài Toán 25

[3.2](#_heading=h.243i4a2) Cơ Sở Dữ Liệu 25

[4.](#_heading=h.j8sehv) CHƯƠNG TRÌNH VÀ KẾT QUẢ 26

[4.1](#_heading=h.338fx5o) Tổ Chức Chương Trình 26

[4.1.1.](#_heading=h.1idq7dh) Sử Dụng ngôn ngữ PHP để thiết kế WebServer: 26

[4.2](#_heading=h.42ddq1a) Kết quả 33

[4.2.1](#_heading=h.2hio093) ANDROID STUDIO 33

[4.2.2](#_heading=h.wnyagw) SERVER 36

[4.3 Nhận xét 40](#_heading=h.3gnlt4p)

[5.](#_heading=h.1vsw3ci) KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 40

[❖](#_heading=h.4fsjm0b) Kết quả đạt được: 40

[⮚](#_heading=h.2uxtw84) Ưu điểm: 40

[⮚](#_heading=h.1a346fx) Nhược điểm: 40

[❖](#_heading=h.3u2rp3q) Hướng phát triển: 40

[TÀI LIỆU THAM KHẢO 41](#_heading=h.2981zbj)

**DANH MỤC HÌNH VẼ**

Hình 2.1.1: Cấu trúc dữ liệu trong TCP/IP

Hình 2.1.2: Mô hình OSI

Hình 2.2.1: Cơ chế hoạt động của FTP

Hình 2.2.5: Kết nối kênh dữ liệu dạng chủ động

Hình 2.2.5.1: Kết nối kênh dữ liệu dạng bị động

Hình 2.3.3: Cơ chế hoạt động của HTTP

Hình 2.3.4: Qui trình hoạt động của kết nối HTTP không bền vững

Hình 2.3.5: Cấu tạo của một url

Hình 2.3.6: Mô hình làm việc thư viện volley

Hình 1: Đăng nhập

Hình 2: Đăng nhập thất bại

Hình 3: Đăng nhâp thành công

Hình 4: Đăng kí thành công

Hình 5: Upload ảnh

Hình 6: Xem ảnh

Hình 7: Form đăng nhập

Hình 8: Danh sách ảnh

Hình 9: Form thêm ảnh mới

Hình 10: Thông báo xác nhận xóa ảnh

Hình 11: Form cập nhật thông tin ảnh

Hình 12: Danh sách ảnh tìm được

Hình 13: Form chi tiết thông tin ảnh

# **GIỚI THIỆU ĐỀ TÀI**

Ngày nay cùng với sự phát triển chóng mặt của khoa học kỹ thuật, một kỷ nguyên mới được mở ra kỷ nguyên của công nghệ thông tin. Nhu cầu của con người càng lớn đặc biệt các ngành khoa học kỹ thuật khác đều cần đến sự hỗ trợ của công nghệ thông tin. Mọi người đếu có nhu cầu truy cập internet để cung cấp cũng như lấy những thông tin cần thiết qua các máy chủ.

Với những lý do trên em xin chọn đề tài **“Xây dựng ứng dụng di động gửi ảnh lên server để quản lí”.**

Người sử dụng chương trình là người có nhu cầu truyền nhận file, tải ảnh lên một server nào đó để quản lí ảnh của bản thân bằng giao thức FTP (File Transfer Protocol) và HTTP thông qua mạng Internet.

Trong môi trường Internet, khi cần truyền nhận file các file có kích thước lớn trên vài chục KB người sử dụng thường gặp trường hợp việc truyền nhận file bị gián đoạn. Để giải quyết vấn đề này, người sử dụng thường sử dụng các chương trình chuyên dùng để lấy file như ReGet để có thể tiếp tục lấy file từ vị trí xảy ra lỗi.

Người sử dụng có xu hướng lấy đồng thời nhiều file tại một server nào đó trên Internet. Như vậy chương trình phải hỗ trợ lấy nhiều file đồng thời. Tuy nhiên không phải lúc nào chương trình cũng có thể lấy đồng thời nhiều file tại một site nếu server hạn chế số kết nối đồng thời tới một địa chỉ IP.

Người sử dụng cũng có nhu cầu lấy file bằng nhiều giao thức khác nhau như FTP (File Transfer Protocol), HTTP (HyperText Transfer Protocol), …

Tuy nhiên trong phạm vi của luận văn này chúng em chỉ có thể hỗ trợ được việc lấy file bằng giao thức HTTP.

Bất cứ khi nào bạn tạo một URL trên trình duyệt của bạn mục đích lấy dữ liệu (resource) từ server sử dụng HTTP. Trình duyệt đưa URL vào một request message, và gửi yêu cầu đến máy chủ HTTP. Máy chủ HTTP phân tích tin yêu cầu và trả lại bạn thông tin tương ứng, thông tin trả về có thể là một tài nguyên bạn yêu cầu hoặc một thông báo lỗi

Mặt khác chương trình cũng phải tạo sự tiện lợi cho người sử dụng. Chương trình phải có giao diện đồ họa thân thiện, giúp cho người sử dụng xem được nội dung các thư mục từ xa và cục bộ. Từ đó giúp cho việc upload và download dễ dàng hơn.

## **1.1 Mục tiêu:**

Mục tiêu của đề tài:

* Tạo một ứng dụng di động để quản lí ảnh
* Nghiên cứu tìm hiểu về họ giao thức TCP/IP, cách thức hoạt động, ưu nhược điểm của giao thức này.
* Nghiên cứu sử dụng hệ CSDL MySQL
* Nghiên cứu về ngôn ngữ lập trình PHP và Java (Android)

# **CƠ SỞ LÝ THUYẾT**

## **Họ giao thức TCP/IP:**

TCP/IP là một giao thức để kết nối và truyền dữ liệugiữa các máy tính với nhau.

* Mục đích chính của IP là cung cấp khả năng kết nối các mạng con thành liên mạng để truyền dữ liệu. IP là một giao thức kiểu không liên kết.
* TCP là một giao thức kiểu "có liên kết", nghĩa là phải thiết lập liên kết giữa một cặp thực thể TCP truớc khi chúng trao đổi dữ liệu với nhau.

Địa chỉ IP dùng để các máy tính liên lạc được với nhau và phân biệt được nhau.

* IP tĩnh và IP động Cấu trúc của địa chỉ IP này gồm một chuỗi số 32 bit và được chia thành 4 nhóm, cách nhau bởi dấu chấm ( . ). Mỗi nhóm gồm 3 chữ số có giá trị từ 000 đến 255.
* Địa chỉ IP là những dãy số, do đó người dùng Internet rất khó nhớ. Vì vậy người ta dùng khái niệm tên miền (Domain Name) để ánh xạ tới địa chỉ IP. Tuy nhiên tên miền chỉ là tên dành cho người dùng dễ nhớ, cốt lõi của địa chỉ Internet vẫn là IP.

### ***Tầng liên kết:***

Tầng liên kết (còn được gọi là tầng liên kết dữ liệu hay là tầng giao tiếp mạng) là tầng thấp nhất trong mô hình TCP/IP, bao gồm các thiết bị giao tiếp mạng và chương trình cung cấp các thông tin cần thiết để có thể hoạt động, truy nhập đường truyền vật lý qua thiết bị giao tiếp mạng đó.

### ***Tầng Internet:***

Tầng Internet (còn gọi là tầng mạng) xử lý quá trình truyền gói tin trên mạng. Các giao thức của tầng này bao gồm: IP (Internet Protocol), ICMP (Internet Control Message Protocol), IGMP (Internet Group Messages Protocol).

### ***Tầng giao vận:***

Tầng giao vận phụ trách luồng dữ liệu giữa hai trạm thực hiện các ứng dụng của tầng trên. Tầng này có hai giao thức chính: TCP (Transmission Control Protocol) và UDP (User Datagram Protocol) TCP cung cấp một luồng dữ liệu tin cậy giữa hai trạm, nó sử dụng các cơ chế như chia nhỏ các gói tin của tầng trên thành các gói tin có kích thước thích hợp cho tầng mạng bên dưới, báo nhận gói tin, đặt hạn chế thời gian time-out để đảm bảo bên nhận biết được các gói tin đã gửi đi.

UDP cung cấp một dịch vụ đơn giản hơn cho tầng ứng dụng. Nó chỉ gửi các gói dữ liệu từ trạm này tới trạm kia mà không đảm bảo các gói tin đến được tới đích. Các cơ chế đảm bảo độ tin cậy cần được thực hiện bởi tầng trên.

### ***Tầng ứng dụng:***

Tầng ứng dụng là tầng trên cùng của mô hình TCP/IP bao gồm các tiến trình và các ứng dụng cung cấp cho người sử dụng để truy cập mạng. Có rất nhiều ứng dụng được cung cấp trong tầng này, mà phổ biến là: Telnet: sử dụng trong việc truy cập mạng từ xa, FTP (File Transfer Protocol).

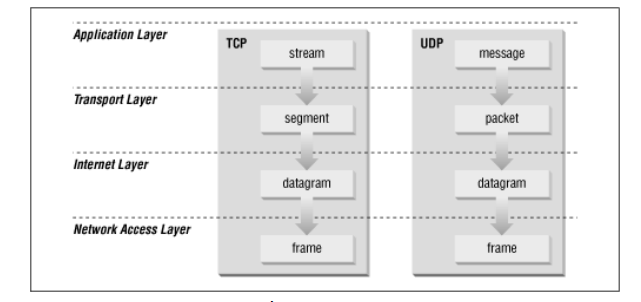
Cũng tương tự như trong mô hình OSI, khi truyền dữ liệu, quá trình tiến hành từ tầng trên xuống tầng dưới, qua mỗi tầng dữ liệu được thêm vào một thông tin điều khiển được gọi là phần header. Trong hình vẽ này ta thấy tại các tầng khác nhau dữ liệu được mang những thuật ngữ khác nhau:

Trong tầng ứng dụng dữ liệu là các luồng được gọi là stream.

Trong tầng giao vận, đơn vị dữ liệu mà TCP gửi xuống tầng dưới gọi là TCP segment.

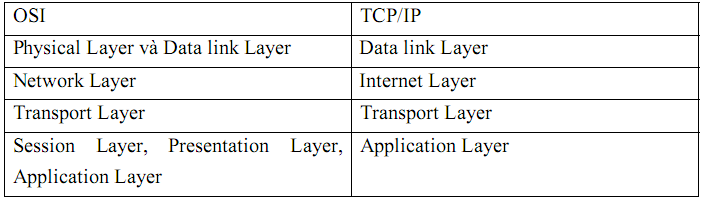
Trong tầng mạng, dữ liệu mà IP gửi tới tầng dưới được gọi là IP datagram.

Trong tầng liên kết, dữ liệu được truyền đi gọi là frame.



Hình 2.1.1: Cấu trúc dữ liệu trong TCP/IP

Bảng sau chỉ rõ mối tương quan giữa các tầng trong mô hình TCP/IP với OSI



Hình 2.1.2: Mô hình OSI

Sự khác nhau giữa TCP/IP và OSI chỉ là:

Tầng ứng dụng trong mô hình TCP/IP bao gồm luôn cả 3 tầng trên của mô hình OSI

Tầng giao vận trong mô hình TCP/IP không phải luôn đảm bảo độ tin cậy của việc truyển tin như ở trong tầng giao vận của mô hình OSI mà cho phép thêm một lựa chọn khác là UDP.s

## **Giao Thức HTTP:**

### ***Khái niệm :***

HTTP ( HyperText Transfer Protocol - Giao thức truyền tải siêu văn bản ) là một trong các giao thức chuẩn về mạng Internet, được dùng để liên hệ thông tin giữa Máy cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client), là giao thức Client/Server dùng cho World Wide Web – WWW

HTTP là một giao thức ứng dụng của bộ giao thức TCP/IP (các giao thức nền tảng cho Internet).

### ***So sánh giữa HTTP và HTTPS :***

HTTPS là viết tắt của Hypertext Transfer Protocol Secure. Vấn đề với giao thức HTTP thông thường là thông tin từ máy chủ đến trình duyệt không được mã hóa, có nghĩa là nó có thể dễ dàng bị đánh cắp .

Giao thức HTTPS khắc phục điều này bằng cách sử dụng chứng chỉ SSL **(**lớp cổng bảo mật**)** , giúp tạo kết nối được mã hóa an toàn giữa máy chủ và trình duyệt, từ đó bảo vệ thông tin nhạy cảm bị đánh cắp khi thông tin được chuyển giữa máy chủ và trình duyệt.

HTTPS hoạt động tương tự như HTTP nhưng được bổ sung thêm SSL và giao thức TSL. Các giao thức này đảm bảo rằng không ai khác ngoài các máy khách và máy chủ có thể hack thông tin, dữ liệu ra ngoài. Cho dù bạn sử dụng máy tính cá nhân hay công cộng đi chăng nữa, các chứng chỉ SSL vẫn đảm bảo thông tin liên lạc của máy khách với máy chủ luôn được an toàn và chống bị dòm ngó.

Sự khác biệt quan trọng nhất giữa hai giao thức là chứng chỉ SSL . Trong thực tế, HTTPS về cơ bản là một giao thức HTTP với bảo mật bổ sung. Tuy nhiên, bảo mật bổ sung này có thể cực kỳ quan trọng, đặc biệt là đối với các trang web có dữ liệu nhạy cảm từ người dùng, chẳng hạn như thông tin thẻ tín dụng và mật khẩu .

Ngoài ra :

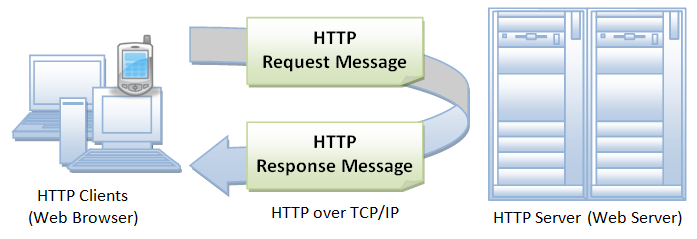
HTTP sử dụng số cổng 80 để giao tiếp và sử dụng HTTPS 443

HTTP được coi là không an toàn và HTTPS an toàn

HTTP hoạt động ở tầng ứng dụng và HTTPS hoạt động ở tầng giao vận

HTTP không yêu cầu bất kỳ chứng chỉ nào và HTTPS cần chứng chỉ SSL

### ***Cơ chế hoạt động của HTTP :***



Hình 2.3.3: Cơ chế hoạt động của HTTP

HTTP hoạt động dựa trên mô hình Client – Server. Trong mô hình này, các máy tính của người dùng sẽ đóng vai trò làm máy khách (Client). Sau một thao tác nào đó của người dùng, các máy khách sẽ gửi yêu cầu đến máy chủ (Server) và chờ đợi câu trả lời từ những máy chủ này.

HTTP là một stateless protocol. Hay nói cách khác, request hiện tại không biết những gì đã hoàn thành trong request trước đó.

HTTP cho phép tạo các yêu cầu gửi và nhận các kiểu dữ liệu, do đó cho phép xây dựng hệ thống độc lập với dữ liệu được truyển giao.

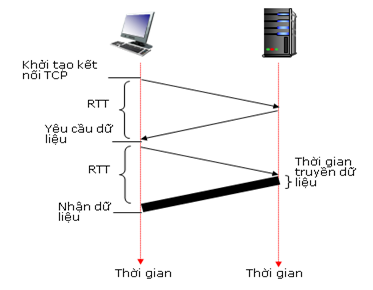
### ***Kết nối HTTP :***

Có hai loại kết nối HTTP là kết nối không bền vững và kết nối bền vững.

Kết nối không bền vững: sau khi, server gửi đi một đối tượng thì kết nối TCP sẽ được đóng. Như vậy, mỗi kết nối TCP chỉ truyền được duy nhất một yêu cầu từ client và nhận lại một thông điệp trả lời từ server.

Kết nối bền vững: server sẽ duy trì kết nối TCP cho việc gửi nhiều đối tượng. Như vậy, sẽ có nhiều yêu cầu từ client được gửi đến server trên cùng một kết nối. Thông thường kết nối TCP này sẽ được đóng lại trong một khoảng thời gian định trước.

Quy trình hoạt động của kết nối HTTP không bền vững:



Hình 2.3.4: Qui trình hoạt động của kết nối HTTP không bền vững

Hình 2.3.4 thể hiện chi tiết các bước hoạt động trong kết  nối HTTP không bền vững giữa HTTP client (Firefox, Chrome,…) và HTTP server:

- Bước 1: client khởi tạo kết nối TCP bằng việc gửi yêu cầu đến server. Server nhận được yêu cầu và chấp nhận kết nối bằng việc gửi trả lời về cho client. Nếu sau khoảng thời gian RTT mà không nhận được trả lời từ phía server thì client sẽ gửi lại yêu cầu.

- Bước 2: sau khi kết nối được thiết lập, client sẽ gửi thông điệp yêu cầu chứa tên đường dẫn của các đối tượng ( ví dụ: www.hutech.edu.vn/homepage/index.php) đến server. Server nhận được thông điệp yêu cầu và tiến hành lấy ra các đối tượng được yêu cầu. Sau đó, các đối tượng được đóng gói thành thông điệp trả lời và gửi đến client.

- Bước 3: server đóng kết nối TCP (Lưu ý: server chỉ đóng kết nối TCP khi chắc chắn rằng client nhận được thông điệp trả lời)

- Bước 4: client nhận thông điệp trả lời chứa tập tin HTML và hiển thị các đối tượng.

- Bước 5: lặp lại các bước từ 1-4 cho các đối tượng khác.

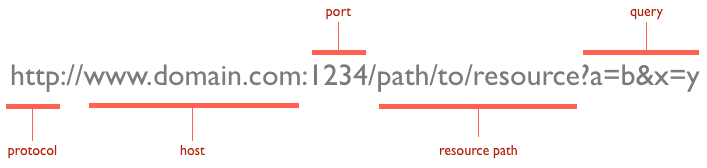
\* Lưu ý: Trong kết nối HTTP không bền vững cần có một RTT (Round Trip Time) để khởi tạo kết nối TCP. Ngoài ra, cần có một RTT cho thông điệp HTTP yêu cầu và một vài byte đầu tiên của thông điệp HTTP  trả lời được trả về.Tổng thời gian truyền tập tin = 2RTT+ thời gian truyền.

Qui trình hoạt động của kết nối HTTP bền vững:

* Kết nối bền vững không có pipelining:
* Client phát ra yêu cầu mới, chỉ khi đáp ứng trước đó đã nhận xong.
* RTT cho mỗi đối tượng tham chiếu.
* Kết nối bền vững có pipelining:
* Mặc định có trong HTTP/1.1.
* Client gửi yêu cầu ngay sau khi gặp một đối tượng tham chiếu.
* Ít nhất 1 RTT cho tất cả đối tượng tham chiếu.

### ***Uniform Resource Locators (URL) :***

Trọng tâm của truyền thông web là thông báo yêu cầu, được gửi thông qua Uniform Resource Locators (URL), dùng để tham chiếu tới tài nguyên trên Internet.



Hình 2.3.5: Cấu tạo của một url

**Protocol** : chính là Giao thức sử dụng, thường là HTTP hoặc HTTPs. Trong trường hợp truyền nhận file thì là FTP.

**Host** : cái này chính là một tên miền hoặc một địa chỉ IP của một máy tính nào đó nơi sẽ xử lý của yêu cầu từ Client.

**Port** : Số cổng mà server đang chạy để lắng nghe việc xử lý các Request HTTP.

Thông thường dịch vụ web sẽ chạy ở cổng 80 của máy tính theo mặc định.

**Resource Path** : Đây chính là đường đến tài nguyên, đây là phần cực kỳ quan trọng, xác định tài nguyên nào sẽ được yêu cầu.

**Query**: Đây là tập các đối số (Param) gửi kèm theo để cung cấp thêm thông tin.

### ***HTTP Request Methods :***

Giao thức HTTP định nghĩa ra một tập phương thức yêu cầu ( Request methods ) để bạn sử dụng trong việc tương tác với máy chủ . cụ thể là :

**GET :**GET được sử dụng để lấy thông tin từ Server.

  Khi bạn truy câp một trang web từ trình duyệt thông qua URL thì bạn đang thực hiện một GET request trên URL đó để lấy dữ liệu.

**POST :**Một yêu cầu POST được sử dụng để gửi dữ liệu tới Server.

  Khi bạn nhập mật khẩu và password trong một trang web nào đó và nhấp Login, thì lúc này bạn đang gửi dữ liệu của bạn đến máy chủ của trang web đó thông qua phương thức POST.

**PUT :**Phương thức này được dùng với ý nghĩa là bạn đang thay đổi, cập nhật lại nội dung của một nguồn dữ liệu nào đó.

  Ví dụ bạn muốn cập nhật thông tin Profile trong trang Sociss Class, sau khi điền thông tin, bạn nhấp vào Save, thì lúc này mình đang gửi một PUT request tới Server để cập nhật lại thông tin cho các bạn.

**DELETE :**Phương thức này sử dụng với ý nghĩa bạn muốn xóa một tài nguyên nào đó.

 Ví dụ bạn muốn xóa một bình luận của bạn, thì bạn sẽ gửi một DELETE request kèm theo ID của bình luận đó tới Server, lúc này Server sẽ nhận ra là bạn đang muốn xóa sản phẩm với ID đó và thực hiện yêu cầu của bạn.

**OPTIONS :**Phương thức này thường được gửi đến Server trước mỗi Request POST, PUT, GET , DELETE ở trên để kiếm tra xem Server chấp nhận những yêu cầu loại nào.

  **HEAD :**Phương thức này chỉ lấy về phần Header của tài nguyên không có phần thân (Body) , thường dùng để kiểm tra thông tin của tài nguyên đó , ứng dụng nhiều trong việc Caching của trình duyệt.

### ***HTTP Responses Methods :***

Cấu trúc của một HTTP response:

* Một **Status-line** = **Phiên bản HTTP** + **Mã trạng thái** + **Trạng thái**
* Có thể có hoặc không có các trường header
* Một dòng trống để đánh dấu sự kết thúc của các trường header
* Tùy chọn một thông điệp

## **Cơ sở dữ liệu MySQL:**

### ***Khái niệm :***

MySQL là một hệ thống quản lý cơ sở dữ liệu quan hệ mã nguồn mở (RDBMS) dựa trên ngôn ngữ truy vấn có cấu trúc ( SQL) được phát triển, phân phối và hỗ trợ bởi tập đoàn Oracle. MySQL chạy trên hầu hết tất cả các nền tảng, bao gồm cả Linux , UNIX và Windows. MySQL thường được kết hợp với các ứng dụng web.

SQL là ngôn ngữ phổ biến nhất để thêm, truy cập và quản lý nội dung trong cơ sở dữ liệu. Nó được chú ý nhất vì khả năng xử lý nhanh, độ tin cậy đã được chứng minh, dễ sử dụng và linh hoạt. MySQL là một phần thiết yếu của hầu hết mọi ứng dụng [PHP](https://viettuts.vn/php) mã nguồn mở. Các ví dụ điển hình cho các tập lệnh dựa trên PHP và MySQL là WordPress, Joomla, Magento và Drupal. MySQL đang trở nên phổ biến vì nhiều lý do tốt:

* MySQL được phát hành theo giấy phép nguồn mở. Vì vậy, bạn không phải trả tiền để sử dụng nó.
* MySQL là một chương trình rất mạnh theo đúng nghĩa của nó. Nó xử lý một tập hợp lớn các chức năng của các gói cơ sở dữ liệu mạnh mẽ và đắt tiền nhất.
* MySQL sử dụng một dạng chuẩn của ngôn ngữ dữ liệu SQL nổi tiếng.
* MySQL hoạt động trên nhiều hệ điều hành và với nhiều ngôn ngữ bao gồm PHP, PERL, C, C ++, JAVA, v.v.
* MySQL hoạt động rất nhanh và hoạt động tốt ngay cả với các tập dữ liệu lớn.
* MySQL rất thân thiện với PHP, ngôn ngữ được đánh giá cao nhất để phát triển web.
* MySQL hỗ trợ cơ sở dữ liệu lớn, lên tới 50 triệu hàng hoặc nhiều hơn trong một bảng. Giới hạn kích thước tệp mặc định cho một bảng là 4GB, nhưng bạn có thể tăng mức này (nếu hệ điều hành của bạn có thể xử lý nó) đến giới hạn lý thuyết là 8 triệu terabyte (TB).
* MySQL là tùy biến. Giấy phép GPL mã nguồn mở cho phép các lập trình viên sửa đổi phần mềm MySQL để phù hợp với môi trường cụ thể của riêng họ.

### ***Cách thức hoạt động của MySQL:***

MySQL dựa trên mô hình client-server. Cốt lõi của MySQL là máy chủ MySQL, xử lý tất cả các hướng dẫn cơ sở dữ liệu (hoặc các lệnh). Máy chủ MySQL có sẵn như là một chương trình riêng biệt để sử dụng trong môi trường mạng client-server và như một thư viện có thể được nhúng (hoặc liên kết) vào các ứng dụng riêng biệt.

MySQL hoạt động cùng với một số chương trình tiện ích hỗ trợ quản trị cơ sở dữ liệu MySQL. Các lệnh được gửi đến MySQLServer thông qua máy khách MySQL, được cài đặt trên máy tính.

MySQL ban đầu được phát triển để xử lý cơ sở dữ liệu lớn một cách nhanh chóng. Mặc dù MySQL thường chỉ được cài đặt trên một máy, nhưng nó có thể gửi cơ sở dữ liệu đến nhiều vị trí, vì người dùng có thể truy cập thông qua các giao diện máy khách MySQL khác nhau. Các giao diện này gửi các câu lệnh SQL đến máy chủ và sau đó hiển thị kết quả.

### ***Các tính năng cốt lõi của MySQL:***

MySQL cho phép dữ liệu được lưu trữ và truy cập trên nhiều công cụ lưu trữ, bao gồm InnoDB, CSV và NDB. MySQL cũng có khả năng sao chép dữ liệu và phân vùng bảng để có hiệu suất và độ bền tốt hơn. Người dùng MySQL không bắt buộc phải học các lệnh mới; họ có thể truy cập dữ liệu của mình bằng các lệnh [SQL](https://viettuts.vn/sql) tiêu chuẩn.

MySQL được viết bằng C và C++ và có thể truy cập và có sẵn trên hơn 20 nền tảng, bao gồm Mac, Windows, Linux và Unix. RDBMS hỗ trợ cơ sở dữ liệu lớn với hàng triệu bản ghi và hỗ trợ nhiều loại dữ liệu bao gồm các số nguyên có chữ ký hoặc không dấu có độ dài 1, 2, 3, 4 và 8 byte(s); FLOAT; DOUBLE; CHAR; VARCHAR; BINARY; VARBINARY; TEXT; BLOB; DATE; TIME; DATETIME; TIMESTAMP; YEAR; SET; ENUM; và các kiểu OpenGIS. Các loại chuỗi có độ dài cố định và biến đổi cũng được hỗ trợ.

Để bảo mật, MySQL sử dụng một đặc quyền truy cập và hệ thống mật khẩu được mã hóa cho phép xác minh dựa trên máy chủ. Các máy khách MySQL có thể kết nối với Máy chủ MySQL bằng một số giao thức, bao gồm cả giao thức TCP/IP trên bất kỳ nền tảng nào. MySQL cũng hỗ trợ một số chương trình máy khách và tiện ích, chương trình dòng lệnh và công cụ quản trị như MySQL Workbench.

### ***Sự khác biệt giữa MySQL và SQL:***

|  |  |  |
| --- | --- | --- |
| **Tiêu chí** | **SQL** | **MYSQL** |
| **Định nghĩa** | SQL là một ngôn ngữ truy vấn có cấu trúc (Structured Query Language). Nó rất hữu ích để quản lý cơ sở dữ liệu quan hệ. | MySQL là một RDBMS để lưu trữ, truy xuất, sửa đổi và quản trị cơ sở dữ liệu bằng cách sử dụng MySQL. |
| **Kiểu** | SQL là một ngôn ngữ truy vấn. | MySQL là phần mềm cơ sở dữ liệu. Nó đã sử dụng ngôn ngữ "SQL" để truy vấn cơ sở dữ liệu. |
| **Hỗ trợ kết nối** | SQL không cung cấp trình kết nối. | MySQL cung cấp một công cụ tích hợp được gọi là 'MySQL workbench' để thiết kế và phát triển cơ sở dữ liệu. |
| **Mục đích** | Để truy vấn và vận hành hệ thống cơ sở dữ liệu. | Cho phép xử lý dữ liệu, lưu trữ, sửa đổi, xóa theo định dạng bảng. |
| **Sử dụng** | Mã và lệnh SQL được sử dụng trong các hệ thống DBMS và RDMS khác nhau bao gồm MYSQL. | MYSQL được sử dụng làm cơ sở dữ liệu RDBMS. |
| **Cập nhật** | Ngôn ngữ là cố định, và lệnh vẫn giữ nguyên. | Nhận cập nhật thường xuyên. |

*Bảng 2.3.4: Bảng khác biệt giữa MySQL và SQL*

## **Ngôn ngữ lập trình Java:**

### ***Khái niệm:***

Java là một một ngôn ngữ lập trình hiện đại, bậc cao, hướng đối tượng, bảo mật và mạnh mẽ. và là một Platform.

Platform: Bất cứ môi trường phần cứng hoặc phần mềm nào mà trong đó có một chương trình chạy, thì được hiểu như là một Platform. Với môi trường runtime riêng cho mình (JRE) và API, Java được gọi là Platform.

Ngôn ngữ lập trình Java ban đầu được phát triển bởi Sun Microsystems do James Gosling khởi xướng và phát hành vào năm 1995. Phiên bản mới nhất của Java Standard Edition là Java SE 8. Với sự tiến bộ của Java và sự phổ biến rộng rãi của nó, nhiều cấu hình đã được xây dựng để phù hợp với nhiều loại nền tảng khác nhau. Ví dụ: J2EE cho các ứng dụng doanh nghiệp, J2ME cho các ứng dụng di động.

Các phiên bản J2 mới đã được đổi tên thành Java SE, Java EE và Java ME. Phương châm của java là "Write Once, Run Anywhere" - viết một lần chạy nhiều nơi, nghĩa là bạn chỉ cần viết một lần trên window chẳng hạn, sau đó vẫn chương trình đó bạn có thể chạy trên Linux, Android, các thiết bị J2ME...

### ***Các tính năng của Java:***

Ngôn ngữ lập trình java có các tính năng sau:

* **Hướng đối tượng** - Trong Java, mọi thứ đều là một Object. Java có thể dễ dàng mở rộng và bảo trì vì nó được xây dựng dựa trên mô hình Object.
* **Nền tảng độc lập** - Không giống nhiều ngôn ngữ lập trình khác bao gồm cả C và C ++, khi Java được biên dịch, nó không được biên dịch thành ngôn ngữ máy nền tảng cụ thể, thay vào mã byte - nền tảng độc lập. Mã byte này được thông dịch bởi máy ảo (JVM) trên nền tảng nào đó mà nó đang chạy.
* **Đơn giản** - Java được thiết kế để dễ học. Nếu bạn hiểu khái niệm cơ bản về OOP Java, sẽ rất dễ để trở thành master về java.
* **Bảo mật** - Với tính năng an toàn của Java, nó cho phép phát triển các hệ thống không có virut, giả mạo. Các kỹ thuật xác thực dựa trên mã hoá khóa công khai.
* **Kiến trúc - trung lập** - Trình biên dịch Java tạo ra định dạng tệp đối tượng kiến trúc trung lập, làm cho mã biên dịch được thực thi trên nhiều bộ vi xử lý, với sự hiện diện của hệ điều hành Java.
* **Portable** - Là kiến trúc tập trung và không có khía cạnh thực hiện phụ thuộc của đặc tả này làm cho Java khả chuyển. Trình biên dịch trong Java được viết bằng ANSI C, đó là một tập con POSIX.
* **Mạnh mẽ** - Java làm nỗ lực để loại trừ các tình huống dễ bị lỗi bằng cách kiểm tra lỗi tại thời gian biên dịch và kiểm tra lỗi tại runtime.
* **Đa luồng** - Với tính năng đa luồng của Java có thể viết các chương trình có thể thực hiện nhiều tác vụ đồng thời. Tính năng thiết kế này cho phép các nhà phát triển xây dựng các ứng dụng tương tác có thể chạy trơn tru hơn.
* **Thông dịch** - Mã byte Java được dịch trực tiếp tới các máy tính gốc và không được lưu trữ ở bất cứ đâu.
* **Hiệu năng cao** - Với việc sử dụng trình biên dịch Just-In-Time, Java cho phép thực hiện hiệu năng cao.
* **Phân tán** - Java được thiết kế cho môi trường phân tán của Internet.
* **Năng động** - Java là năng động hơn C hoặc C++ vì nó được thiết kế để thích nghi với môi trường đang phát triển. Các chương trình Java có thể mang một lượng lớn thông tin tại runtime mà có thể được sử dụng để xác minh và giải quyết các truy cập vào các đối tượng tại runtime.

### ***Tác Dụng của Java:***

Java đã được sử dụng trong các lĩnh vực khác nhau. Ví dụ:

1. Desktop App như acrobat reader, media player, antivirus, ...
2. Web App như irctc.co.in, javatpoint.com, ...
3. Enterprise App như các ứng dụng về xử lý nghiệp vụ ngân hàng, ...
4. Thiết bị Mobile như các ứng dụng Android.
5. Hệ thống nhúng
6. Smart Card
7. Robot
8. Game App

### ***Các kiểu Java App và Java Platforms:***

## Java App:

Có 4 kiểu ứng dụng chính của java app:

### *Standalone App*

Standalone App cũng được biết đến như Desktop App hoặc Window-based App. Để tạo ra ứng dụng kiểu này người ta thường sử dụng AWT, Swing hoặc JavaFX framework.

### *Web App*

Web App là ứng dụng chạy trên server và tạo được các trang động. Hiện nay, servlet, jsp, struts, jsf, spring... là những công nghệ được sử dụng để tạo Web App trong java.

### *Enterprise App*

Một ứng dụng dạng như Banking App, có lợi thế là tính bảo mật cao, cân bằng tải (load balancing) và clustering. Trong java, EJB được sử dụng để tạo các Enterprise App.

### *Mobile App*

Mobile App là ứng dụng được tạo ra cho các thiết bị di động. Hiện nay Android và Java ME được sử dụng để chạy các ứng dụng này.

## Java Platforms

Có 4 nền tảng Java:

### *Java SE (Java Standard Edition)*

Java SE là một nền tảng lập trình Java. Nó bao gồm các API lập trình Java như java.lang, java.io, java.net, java.util, java.sql, java.math, v.v. Nó bao gồm các chủ đề cốt lỗi như OOPs, String, Regex, Exception, Inner classes, Multithreading, I/O Stream, Networking, AWT, Swing, Reflection, Collection, v.v.

### *Java EE (Java Enterprise Edition)*

Đây là một nền tảng doanh nghiệp chủ yếu được sử dụng để phát triển các ứng dụng web và doanh nghiệp. Nó được xây dựng trên nền tảng Java SE. Nó bao gồm các chủ đề như Servlet, JSP, Web Services, EJB, JPA , v.v.

### *Java ME (Java Micro Edition)*

Đây là một nền tảng vi mô chủ yếu được sử dụng để phát triển các ứng dụng di động.

### *JavaFX*

JavaFX là một nền tảng phần mềm phát triển các ứng dụng Internet phong phú (Rich Internet Applications – RIAs) có thể chạy trên nhiều loại thiết bị, nhiều hệ điều hành khác nhau. JavaFX là một giải pháp công nghệ cho GUI trên nền tảng Java nhằm tạo giao diện đồ họa người dùng dựa trên Swing và Java2D.

## **Ngôn ngữ lập trình PHP:**

### ***Khái niệm:***

**PHP** (viết tắt của Hypertext Preprocessor) là ngôn ngữ lập trình đa mục đích được phát triển từ đầu năm 1994. Cụ thể hơn, PHP là ngôn ngữ kịch bản mã nguồn mở, chạy ở phía server và được dùng để tạo ra các ứng dụng web. Đến hiện nay đã được nhiều người sử dụng để phát triển các ứng dụng phần mềm thông qua lập trình web đặc điểm của ngôn ngữ lập trình này là sử dụng mã nguồn mở, dễ dàng nhúng vào HTML để phát triển website.

Ngôn ngữ lập trình PHP phù hợp với các web để có thể dễ dàng ứng dụng vào các trang HTML. Do đó nó được tối ưu hóa cho các ứng dụng web với những tính năng ưu việt như nhanh, gọn, cùng cấu trúc sử dụng tượng tự như C hay Java. Đây cũng được xem là ngôn ngữ lập trình khá dễ học và nhanh hơn so với một số ngôn ngữ lập trình khác. Bởi vậy, rất nhanh chóng, PHP trở thành ngôn ngữ lập trình web phổ biến nhất thế giới.

* PHP là viết tắt của Hypertext tiền xử lý
* PHP là một ngôn ngữ kịch bản phía máy chủ. Điều này có nghĩa là nó được thực thi trên máy chủ. Các ứng dụng khách không cần cài đặt PHP.
* Các tệp PHP được lưu với phần mở rộng tệp ".php" và mã phát triển PHP được đính kèm trong các thẻ.
* PHP là mã nguồn mở và đa nền tảng

### ***Tác dụng của PHP:***

PHP có thể được sử dụng trên tất cả các hệ điều hành chính, bao gồm Linux, nhiều biến thể Unix (bao gồm HP-UX, Solaris và OpenBSD), Microsoft Windows, macOS, RISC OS và có thể cả các biến thể khác. Ngôn ngữ lập trình PHP cũng đã hỗ trợ cho hầu hết các máy chủ web hiện nay. PHP hoạt động như một mô-đun hoặc bộ xử lý CGI. Vì vậy, ngôn ngữ lập trình PHP, bạn có quyền tự do lựa chọn một hệ điều hành và một máy chủ web. Hơn nữa, bạn cũng có thể lựa chọn sử dụng lập trình thủ tục hoặc lập trình hướng đối tượng (OOP) hoặc kết hợp cả hai.

PHP không bị giới hạn ở HTML đầu ra. Các khả năng của PHP bao gồm xuất ra hình ảnh, tệp PDF, …  được tạo khi đang di chuyển. Cũng có thể dễ dàng xuất bất kỳ văn bản nào, chẳng hạn như XHTML và bất kỳ tệp XML nào khác. PHP có thể tự động tạo các tệp này và lưu chúng trong hệ thống tệp, thay vì in ra, tạo thành bộ đệm phía máy chủ cho nội dung động.

Một trong những tính năng mạnh nhất và quan trọng nhất trong PHP là sự hỗ trợ của nó cho một loạt các cơ sở dữ liệu . Viết một trang web hỗ trợ cơ sở dữ liệu cực kỳ đơn giản bằng cách sử dụng một trong các tiện ích mở rộng dành riêng cho cơ sở dữ liệu (ví dụ: đối với mysql ) hoặc sử dụng lớp trừu tượng như PDO hoặc kết nối với bất kỳ cơ sở dữ liệu nào hỗ trợ chuẩn Kết nối cơ sở dữ liệu mở thông qua tiện ích mở rộng ODBC . Các cơ sở dữ liệu khác có thể sử dụng cURL hoặc ổ cắm , như CouchDB.

PHP cũng có hỗ trợ để nói chuyện với các dịch vụ khác bằng các giao thức như LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (trên Windows) và vô số các dịch vụ khác. Bạn cũng có thể mở các ổ cắm mạng thổ và tương tác bằng bất kỳ giao thức nào khác. PHP đã hỗ trợ cho việc trao đổi dữ liệu phức tạp WDDX giữa hầu như tất cả các ngôn ngữ lập trình Web. Nói về kết nối, PHP có hỗ trợ khởi tạo các đối tượng Java và sử dụng chúng trong suốt như các đối tượng PHP.

PHP có các tính năng xử lý văn bản hữu ích  như Perl ( PCRE ), nhiều phần mềm mở rộng nhằm cho phép người dùng truy cập các tài liệu XML. Ngôn ngữ lập trình PHP có thể đem lại rất nhiều hữu ích cho con người

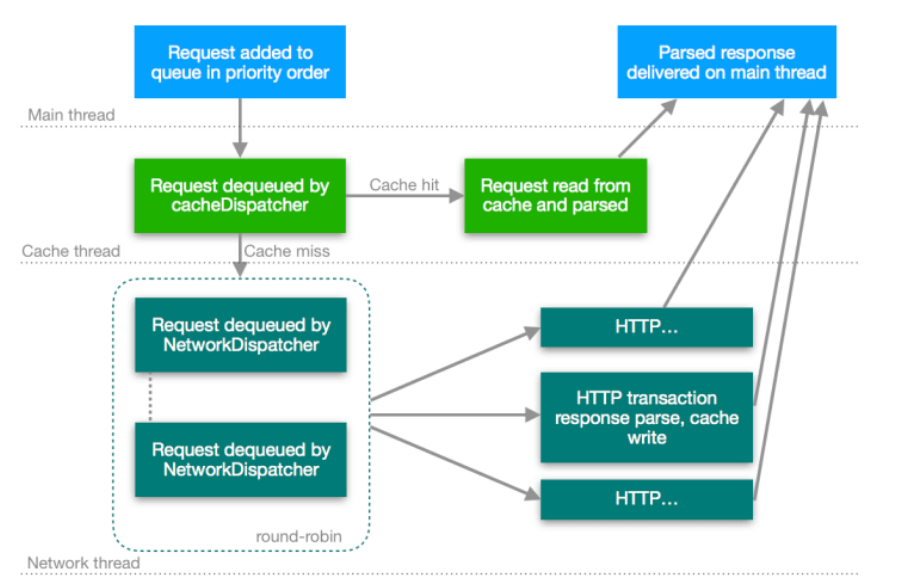
### ***Ưu và Nhược Điểm của ngôn ngữ PHP:***

* **Ưu điểm:**
* Phải kể đến đầu tiên đó chính là PHP là một mã nguồn mở. Việc cài đặt và sử dụng PHP rất dễ dàng và hoàn toàn miễn phí . Chính vì vậy PHP luôn được cài đặt phổ biến trên các WebSever như: Apache, Nginx, IIS.
* Có tính cộng đồng cao: là một mã nguồn mở và dễ sử dụng nên PHP luôn được cộng đồng ưu chuộng. Cộng đồng PHP được coi là khá rộng và chất lượng trên toàn thế giới. Đã có nhiều blog, diễn đàn trong và ngoài nước nói về PHP nên khả năng tiếp cận của mọi người nhanh chóng, dễ dàng hơn.
* Thư viện phong phú : Với tính chất được nhiều người sử dụng nên thư viện về ngôn ngữ lập trình PHP ngày càng được mở rộng. Với thư viện code phong phú, việc học tập và ứng dụng PHP trở nên rất dễ dàng và nhanh chóng. Đây cũng chính là đặc điểm khiến PHP trở nên khá nổi bật và cũng là nguyên nhân vì sao ngày càng có nhiều người sử dụng PHP để phát triển web. Không những vậy mà PHP còn có thể kết hợp được với những cơ sở dữ liệu lớn : MySQL, Oracle, Cassandra.
* Tính bảo mật cao: Bản thân PHP là mã nguồn mở và cộng đồng phát triển khá mạnh nên PHP được coi là khá an toàn khi sử dụng. Khi sử dụng PHP kết hợp các kỹ thuật bảo mật ở các tầng khác thì PHP sẽ trở lên chắc chắn hơn và đảm bảo hoạt động cho website.
* **Nhược điểm:**
* Nhược điểm đầu tiên phải nói đến của PHP là cấu trúc ngữ pháp của nó không được gọn gàng, đẹp mắt như các ngôn ngữ lập trình khác. Và PHP chỉ hoạt động được trên các ứng dụng website.

## **2.6** Volley library Android

### ***2.6.1 Khái Niệm***

Volley là một HTTP library giúp các app Android kết nối mạng đơn giản, developer viết code ít hơn. Đặc biệt là thư viện Volley giúp tăng tốc độ ứng dụng nhờ hỗ trợ cache.



Hình 2.3.6:Mô hình làm việc thư viện Volley Android

### ***2.6.2 Tính Năng Nổi Bật***

* Hỗ trợ lên schedule để tạo request
* Cho phép cùng lúc thực hiện nhiều request trên các thread khác nhau theo độ ưu tiên.
* Cache trên Disk hay RAM
* Cho phép hủy một request.
* Dễ dàng tùy chỉnh cho phù hợp với yêu cầu của ứng dụng. Như thiết lập retry, back off.
* Dễ debug.

### ***2.6.3 Các class sử dụng trong Volley:***

**RequestQueue:** Việc sử dụng Volley rất đơn giản, các bạn chỉ cần tạo một RequestQueue bằng lệnh

RequestQueue mRequestQueue = Volley.newRequestQueue(context);

Phương thức này về một RequestQueue. Khi đã có RequestQueue chúng ta chỉ cần các loại Request và add các request đó vào RequestQueue

**Request:** là lớp cơ sở của các Request trong Volley, chứa thông tin về request HTTP.

**StringRequest:** Kết thừa từ Request, là class đại diện cho request trả về String.

**JSONObjectRequest:** Request có response trả về là JSONObject.

Nếu Request thành công chúng ta có thể lấy ra đối tượng JSONObject trong phương thức onResponse. Ngược lại chúng ta sẽ có thông tin lỗi trong phương thức onErrorResponse.

**JSONArrayRequest**: Tương tự như JSONObjectRequest nhưng JSONArrayRequest trả về một JSONArray nếu thành công trong phương thức onResponse

**ImageRequest:** ImageRequest trả về response là một bitmap nếu thành công. Ngược lại trả về lỗi trong phương thức onErrorResponse.

# **PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

## **Phát Biểu Bài Toán**

Xây dựng WebSite quản lý ảnh

* Xây dựng app di động android chọn ảnh từ thư viện, gửi ảnh lên server, xem ảnh từ server.
* Server sẽ quản lý ảnh: hiển thị, thêm mới, xóa, tìm kiếm ảnh.
* Cần thực hiện chức năng đăng nhập, đăng ký tài khoản mới (nếu chưa có tài khoản) trước khi thực hiện được các chức năng trên.

## **Cơ Sở Dữ Liệu**

* Sử dụng hệ quản trị cơ sở dữ liệu MySQL thông qua chương trình XAMPP với phpmyadmin.
* PhpMyAdmin là phần mềm mã nguồn mở được viết bằng ngôn ngữ PHP giúp quản trị cở sở dữ liệu MySQL thông qua giao diện web.
* MYSQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (gọi tắt là RDBMS) hoạt động theo mô hình client-server. Với RDBMS là viết tắt của Relational Database Management System .MYSQL được tích hợp apache, PHP .MySQL quản lý các cơ sở dữ liệu .Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu . MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL .

Cơ sở dữ liệu gồm 2 bảng chính:

* Account

|  |  |  |
| --- | --- | --- |
| **Tên trường** | Kiểu dữ liệu | Mô tả |
| **id** | Int | ID account, Khoá Chính |
| name | Text | Tên người dùng |
| username | Text | Tên đăng nhập |
| password | Text | Mật khẩu |

*Bảng 3.2.1: Account*

* Image

|  |  |  |
| --- | --- | --- |
| **Tên trường** | Kiểu dữ liệu | Mô tả |
| **id** | Int | ID Ảnh, Khoá Chính |
| name | Text | Tên Ảnh |
| image | Text | Đường dẫn Ảnh |
| id\_account | Int | Id của Account, Khoá Phụ |

*Bảng 3.2.2: Image*

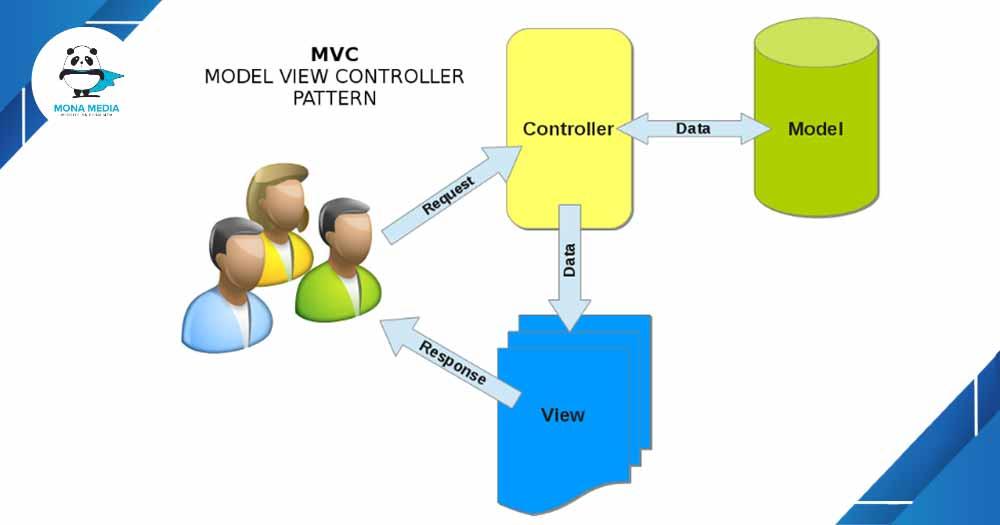
# **CHƯƠNG TRÌNH VÀ KẾT QUẢ**

## **Tổ Chức Chương Trình**



### ***Sử Dụng ngôn ngữ PHP để thiết kế WebServer:***

Lập trình theo Mô Hình MVC (Model-View-Controller):



*Hình 3.3.1: Mô hình MVC*

* Model:

1. E\_Account

|  |  |
| --- | --- |
| **File** | **E\_Account.php** |
| **Xử Lý** | Gồm các thuộc tính(id. name, username, password ) và 1 hàm construct.  public function \_\_construct($\_id,$\_name,$\_image,$\_idaccount); |
| **Chức năng** | Chứa class Entity\_Account có 4 thuộc tính trùng với Bảng Acount ở cơ sở dữ liệu MySQL để lưu dữ liệu. |

*Bảng 3.3.1: E\_Account*

* E\_Image

|  |  |
| --- | --- |
| **File** | **E\_Image.php** |
| **Xử Lý** | Gồm các thuộc tính(id. name, image, id\_account ) và 1 hàm construct  public function \_\_construct($\_id ,$\_name,$\_username,$\_password); |
| **Chức năng** | Chứa class Entity\_Account có 4 thuộc tính trùng với Bảng Acount ở cơ sở dữ liệu MySQL để lưu dữ liệu. |

*Bảng 3.3.2: E\_Image*

* M\_Image

|  |  |
| --- | --- |
| **File** | **M\_Image.php** |
| **Xử Lý** | Gồm tất cả các hàm kết nối với cơ sở dư liệu và các hàm tương tác với Cơ sở dữ liệu:  public function CheckLogin($user,$pass);  public function GetAllImage();  public function GetImageDetail($id);  public function Search($stringSearch);  public function Update($id,$tenanh,$image);  public function Delete($checkbox);  public function Insert($id,$tenanh,$duongdan); |
| **Chức năng** | Chứa các hàm tương tác với cơ sở dữ liệu để gửi dữ liệu về cho Controller và chỉnh sửa trên Server |

*Bảng 3.3.3: M\_Image*

* *View*
* *Login*

|  |  |
| --- | --- |
| **File** | **Login.php** |
| **Xử Lý** | Dùng Html thiết kế Giao diện với Form Login với action = “C\_Login.php” và method = “get”  Form Login gồm 1 tag Text – Nhập Username  1 tag Password – Nhập Password  1 nút Submit để gọi đến C\_Login.php trên form action  1 nút Reset để reset dữ liệu đang nhập vào |
| **Chức năng** | Giao diện đăng nhập của WebServer |

*Bảng 3.3.4: Login*

* Main
* Show

|  |  |
| --- | --- |
| **File** | **WebShow.php** |
| **Xử Lý** | Dùng Html và php để tạo Giao diện Show của WebServer với các block hiện thị ra từng ảnh và tên ảnh tương ứng của từng account được gửi lên và là giao diện chính chứa tất cả các chức năng của server |
| **Chức năng** | Giao diện hiển thị danh sách ảnh của WebServer |

*Bảng 3.3.6: Show*

* Search

|  |  |
| --- | --- |
| **Form** | **Search** |
| **Xử Lý** | Dùng Html thiết kế Giao diện với Form Search với action = “C\_Search.php” và method = “get”  Form Search gồm 1 tag Text – Nhập Tên ảnh cần tìm  1 nút Submit để gọi đến C\_Search.php trên form action  1 nút Reset để reset dữ liệu đang nhập vào |
| **Chức năng** | Giao diện tìm kiếm ảnh của WebServer |

*Bảng 3.3.7: Search*

* Update

|  |  |
| --- | --- |
| **File** | **WebUpdate.php** |
| **Xử Lý** | Dùng Html thiết kế Giao diện với Form Upadte với action = “C\_Update.php” và method = “get”  Form Update gồm tag Text – name = “id” Hiển thị ra Id của Ảnh đang được chọn để Sửa (Không được phép chỉnh sửa id)  tag Text – name = “ta” – Hiện thị ra tên của ảnh đang được chọn để sửa (chỉnh sửa tên ảnh)  tag Text – name = “oldimage” – Hiện thị ra đường dẫn của ảnh đang được chọn để sửa (chỉnh sửa đường dẫn)  tag File – name =”newimage” = Hiển thị ra chức năng chọn 1 file từ máy tính (Chọn một đường dẫn mới – Thay đổi bức ảnh)  1 nút Submit để gọi đến C\_Update.php trên form action  1 nút Reset để reset dữ liệu đang nhập vào |
| **Chức năng** | Giao diện chỉnh sửa thông tin ảnh của WebServer |

*Bảng 3.3.8: Update*

* Insert

|  |  |
| --- | --- |
| **File** | **WebInsert.php** |
| **Xử Lý** | Dùng Html thiết kế Giao diện với Form Insert với action = “C\_Add.php” và method = “get”  Form Insert gồm  tag Text – name = “ta” – Nhập Tên ảnh để Thêm  tag File – name =”image” = Hiển thị ra chức năng chọn 1 file từ máy tính để thêm  1 nút Submit để gọi đến C\_Add.php trên form action  1 nút Reset để reset dữ liệu đang nhập vào |
| **Chức năng** | Giao diện thêm ảnh của WebServer |

*Bảng 3.3.9: Insert*

* + Controller
* C\_Login

|  |  |
| --- | --- |
| **File** | **C\_Login.php** |
| **Xử Lý** | Khi nút Submit trong Form Login.php từ View được kích hoạt thì sẽ lấy dữ liệu từ Form Login để truy cập tới M\_Image.php truy cập về cơ sở dữ liệu để kiểm tra xem tên tài khoản và mật khẩu có tồn tại hay không  Nếu có tồn tại thì chuyển đến C\_Show.php của WebServer còn không tồn tại hoặc nhập sai thì sẽ chuyển ngược về Form Login.php trong View |
| **Chức năng** | Xử Lý chức năng Login của WebServer |

*Bảng 3.3.11: C\_Login*

* C\_Show

|  |  |
| --- | --- |
| **File** | **C\_Show.php** |
| **Xử Lý** | Sau khi đăng nhập xong, thêm mới một ảnh, chỉnh sửa ảnh, hay xóa ảnh thì C\_Show.php được gọi, truy xuất cơ sở dữ liệu để lấy danh sách ảnh rồi hiển thị danh sách ảnh trên server |
| **Chức năng** | Xử Lý chức năng Xem của WebServer |

*Bảng 3.3.12: C\_Show*

* C\_Search

|  |  |
| --- | --- |
| **File** | **C\_Search.php** |
| **Xử Lý** | Nếu nút submit ở Form Search được kích hoạt thì sẽ lấy dữ liệu từ form search truy cập đến M\_Image.php truy cập về cơ sở dữ liệu để tìm những ảnh theo yêu cầu. |
| **Chức năng** | Xử Lý chức năng Tìm Kiếm của WebServer |

*Bảng 3.3.13: C\_Search*

* C\_Add

|  |  |
| --- | --- |
| **File** | **C\_Add.php** |
| **Xử Lý** | Nếu nút submit ở Form Insert được kích hoạt thì sẽ lấy dữ liệu từ form Insert truy cập đến M\_Image.php truy cập về cơ sở dữ liệu để thêm thông tin những ảnh mà account đang đăng nhập vào server  Nếu nút submit chưa được kích hoạt thì sẽ chuyển đến Form Insert. |
| **Chức năng** | Xử Lý chức năng Xem của WebServer |

*Bảng 3.3.13: C\_Add*

* C\_ Delete

|  |  |
| --- | --- |
| **File** | **C\_Delete.php** |
| **Xử Lý** | Nếu nút Delete ở Form Webshow được kích hoạt thì sẽ truy cập đến M\_Image.php truy cập về cơ sở dữ liệu để xoá thông tin ảnh tương ứng mà account đang đăng nhập khỏi server |
| **Chức năng** | Xử Lý chức năng Xem của WebServer |

*Bảng 3.3.14: C\_Delete*

* C\_Update

|  |  |
| --- | --- |
| **File** | **C\_Update.php** |
| **Xử Lý** | Nếu nút submit ở Form Update được kích hoạt thì sẽ lấy dữ liệu từ form Update truy cập đến M\_Image.php truy cập về cơ sở dữ liệu để sửa thông tin những ảnh mà account đang đăng nhập vào server theo những thông tin được đưa vào  Nếu nút submit chưa được kích hoạt thì sẽ chuyển đến Form Update. |
| **Chức năng** | Xử Lý chức năng chỉnh sửa thông tin ảnh của WebServer |

*Bảng 3.3.15: C\_ Update*

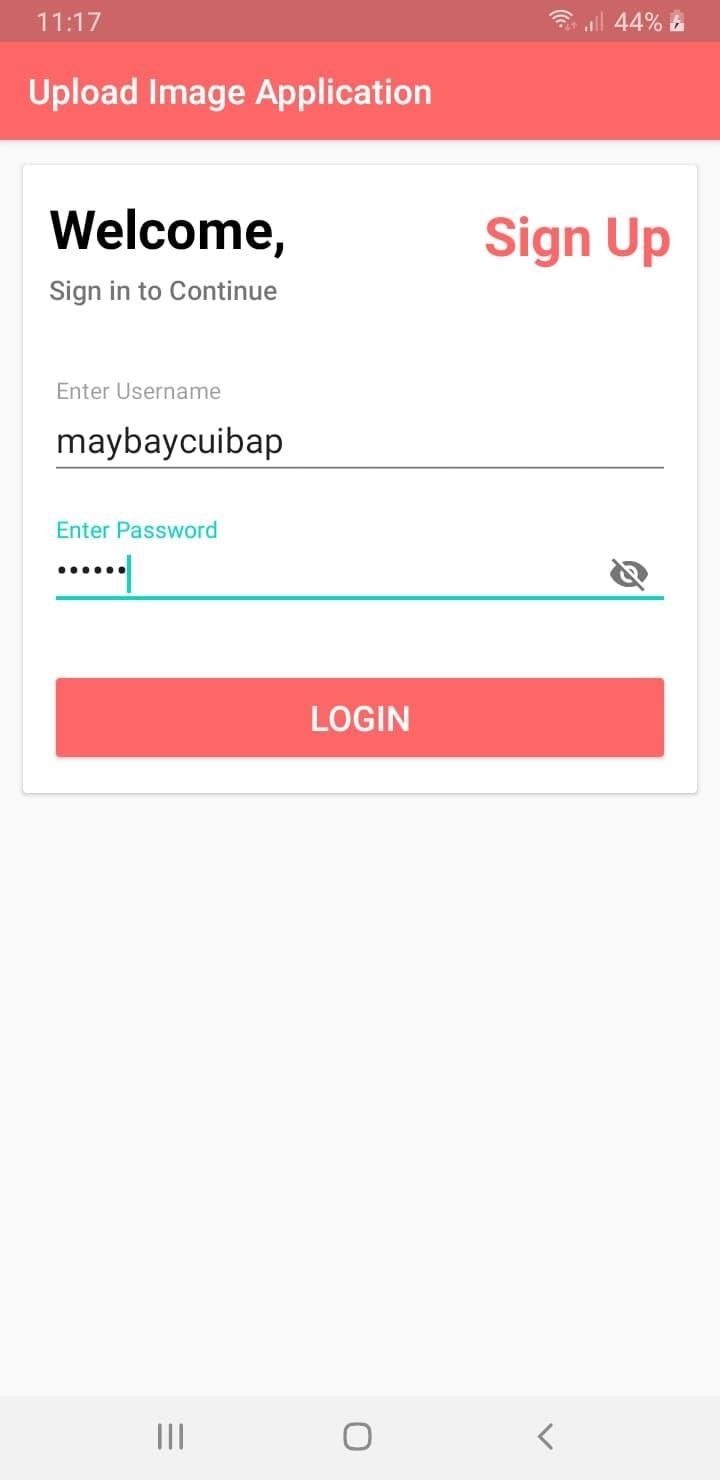


## **Kết quả**

### ***ANDROID STUDIO***

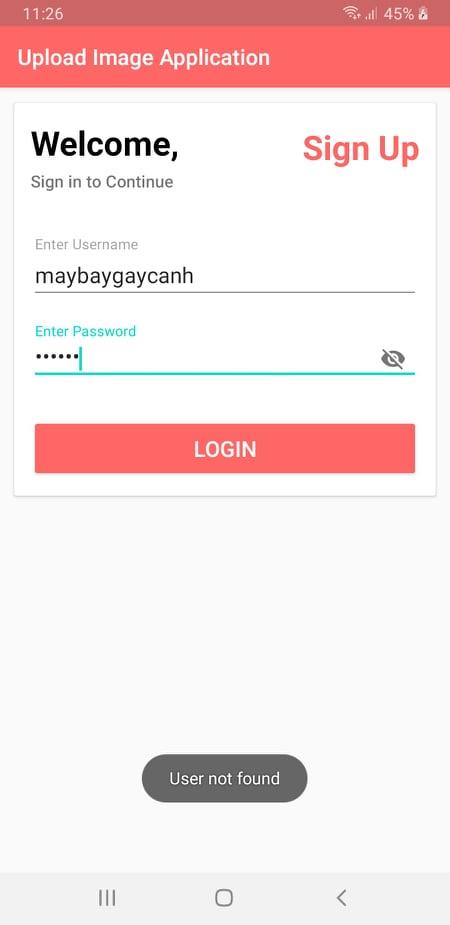
1. Form “LOGIN”

* Người dùng nhập tên đăng nhập và mật khẩu để đăng nhập vào hệ thống.



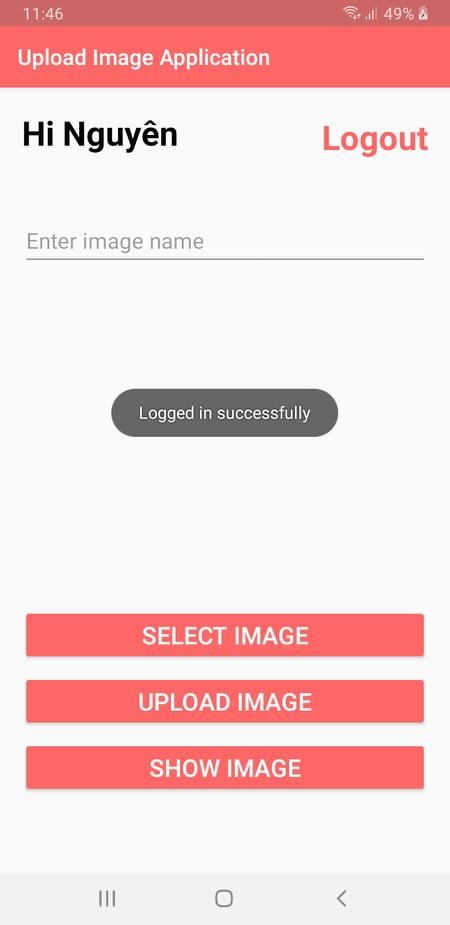
*Hình 1: Đăng nhập*

* Nếu người dùng nhập sai username hoặc password thì chương trình hiện ra thông báo “User not found”.



*Hình 2: Đăng nhập thất bại*

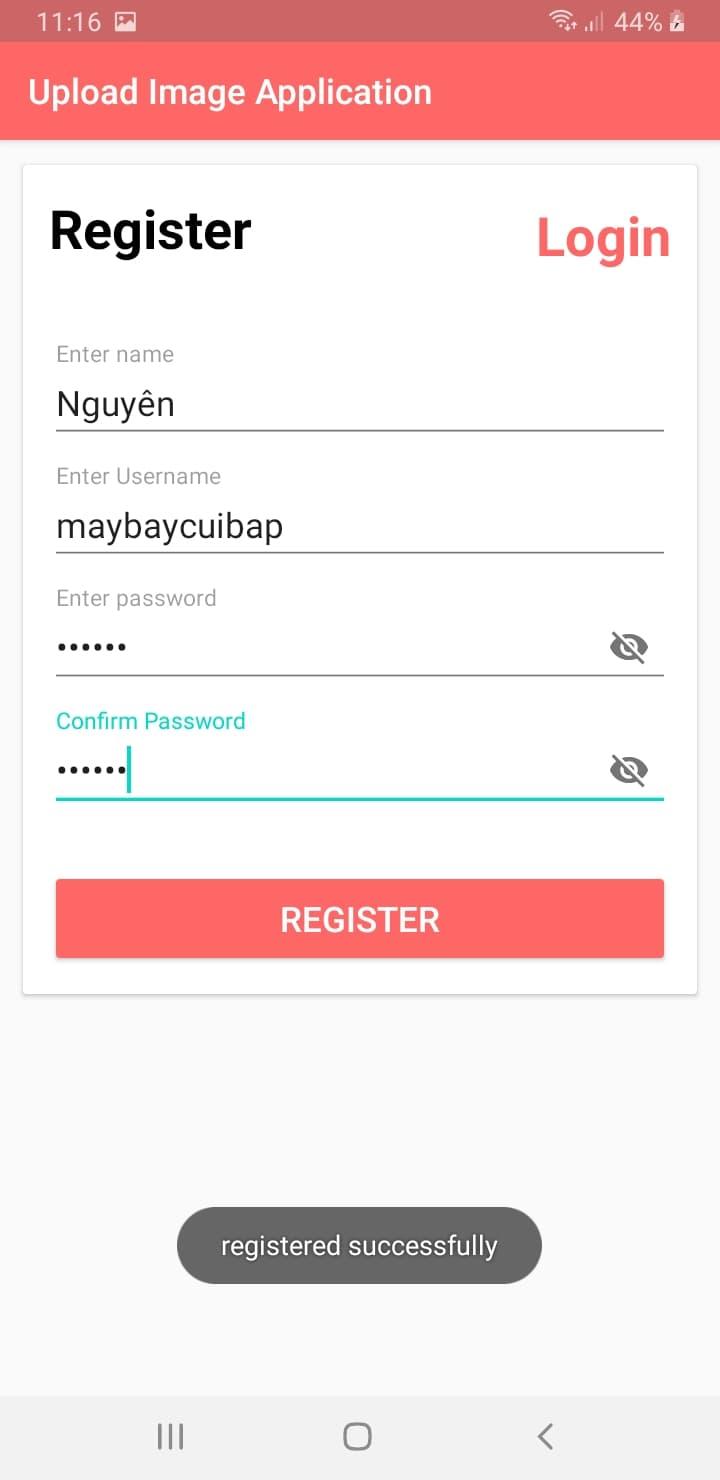
* Sau khi người dùng nhập đúng username và password sẽ chuyển sang form “Upload Image” và thông báo “Logged in successfully”.



*Hình 3: Đăng nhâp thành công*

1. Form đăng ký

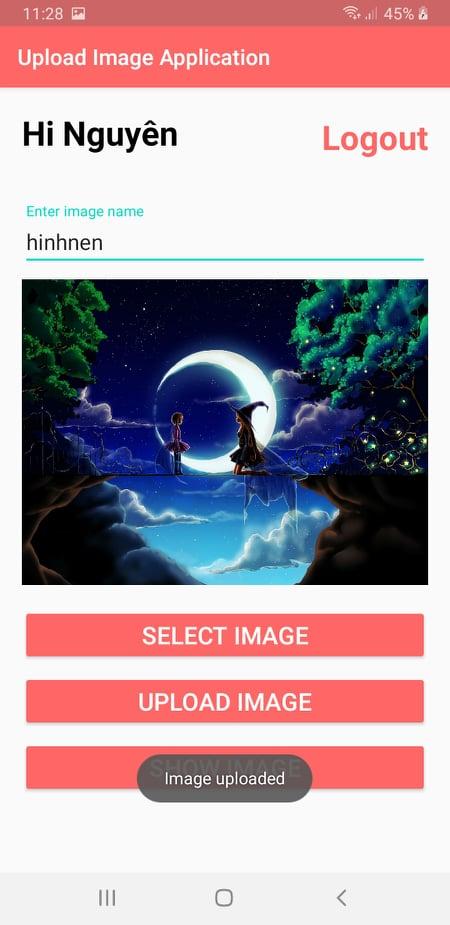
* Người dùng có thể đăng ký tài khoản để đăng nhập vào hệ thống (nếu chưa có tài khoản). Tại đây người dùng nhập tên (name), tên đăng nhập (username), mật khẩu (password), nhập lại mật khẩu (confirm password) để đăng ký tài khoản. Đăng ký thành công chương trình hiện thị thông báo “registered successfully”.



*Hình 4: Đăng kí thành công*

1. Form “UPLOAD IMAGE”

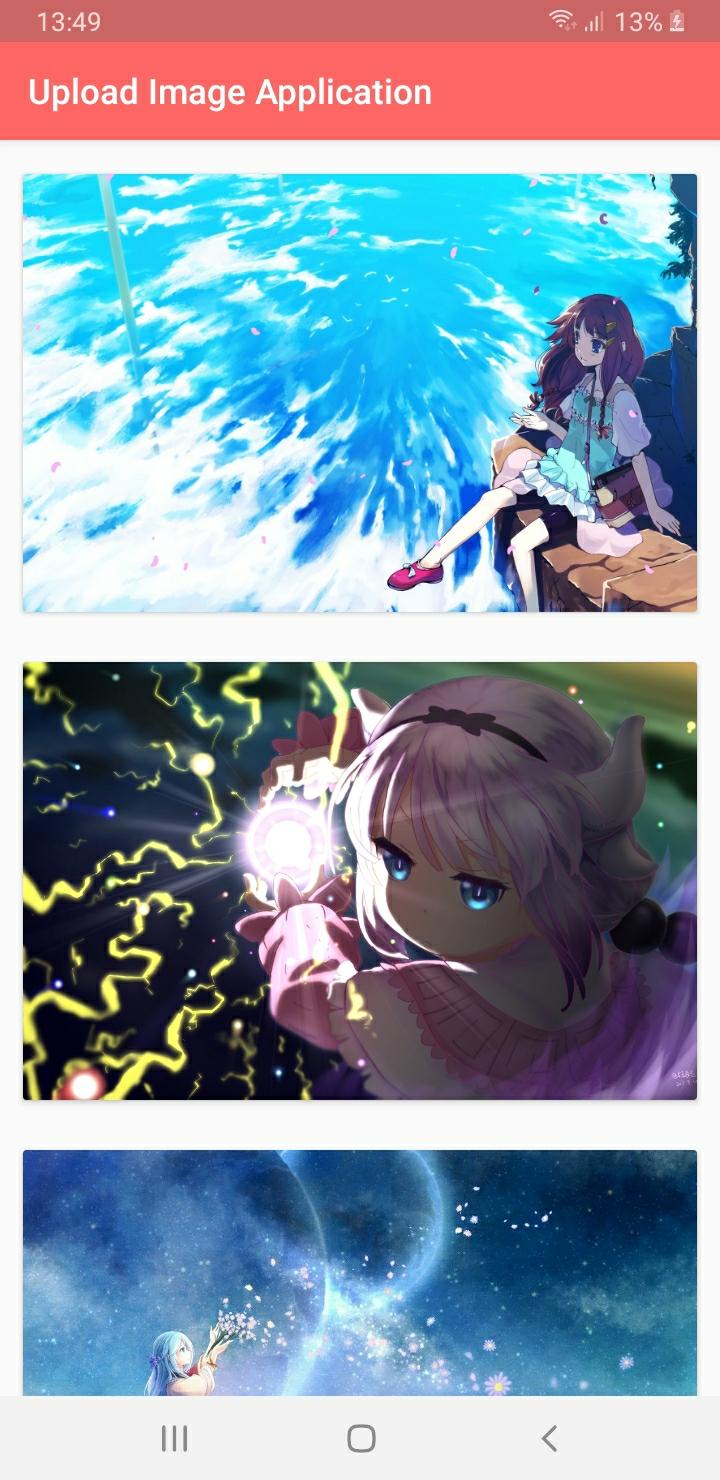
* Để tải ảnh lên server người dùng chọn “SELECT IMAGE” chương trình sẽ hiện ra danh sách ảnh trong thư viện, người dùng chọn một ảnh cần tải và nhập tên cho ảnh đó. Nhấn “UPLOAD IMAGE” để tải ảnh lên.



*Hình 5: Upload ảnh*

1. Form “SHOW IMAGE”

* Để xem tất cả hình ảnh của mình người dùng chọn “SHOW IMAGE”. Tại đây chương trình sẽ hiển thị tất cả hình ảnh của người dùng đó.

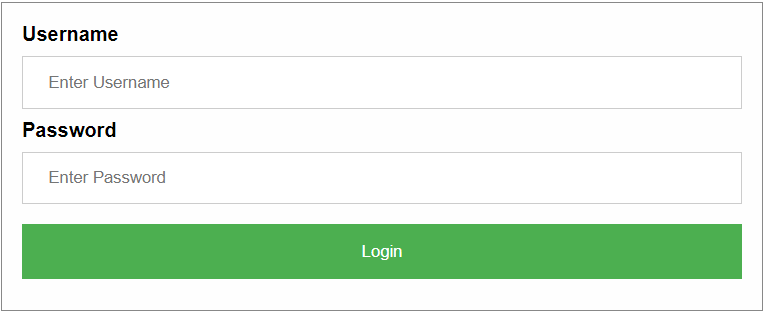


*Hình 6: Xem ảnh*

### ***SERVER***

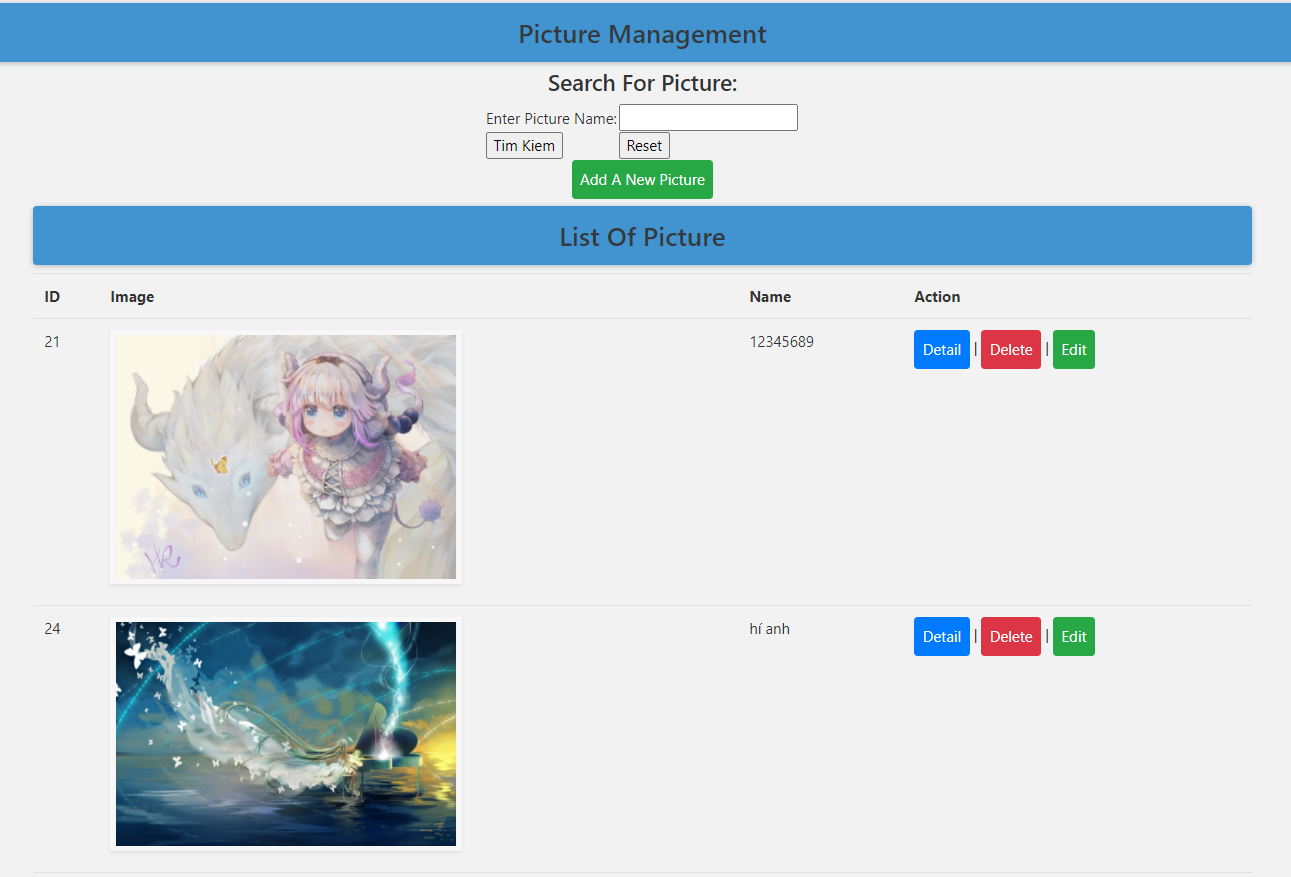
1. Đăng Nhập

* Người dùng nhập tên đăng nhập(username) và mật khẩu(password) để đăng nhập vào hệ thống. Đăng nhập thành công thì chuyển sang form danh sách ảnh, quá trình đăng nhập thành công khi người dùng nhập đúng username và password.



*Hình 7: Form đăng nhập*

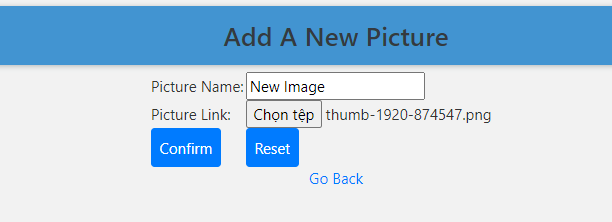
1. Xem ảnh

* Tại form danh sách ảnh sẽ hiển thị tất cả ảnh mà người dùng tải lên server, gồm các button chức năng như “Add a new picture”, “Search”, “Detail”, “Delete”, “Edit”. 

*Hình 8: Danh sách ảnh*

1. Thêm ảnh

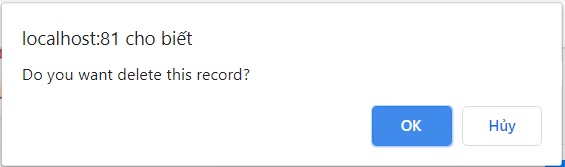
* Để thêm một ảnh mới lên server, người dùng chọn chức năng “Add a new picture”, người dùng nhập tên ảnh và chọn ảnh cần tải lên vào form thêm ảnh, nhấn “Confirm” để thêm mới ảnh.



*Hình 9: Form thêm ảnh mới*

1. Xóa ảnh

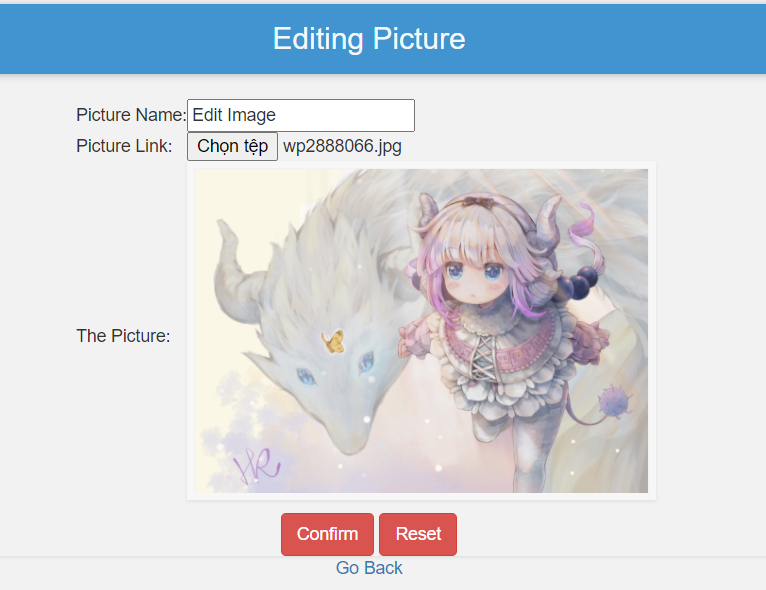
* Để xóa một ảnh người dùng chọn action “Delete” tương ứng với ảnh cần xóa, chương trình sẽ hiển thị một thông báo để xác nhận việc xóa ảnh đó. Chọn “OK” để tiến hành xóa ảnh đó khỏi server.



*Hình 10: Thông báo xác nhận xóa ảnh*

1. Cập nhật ảnh

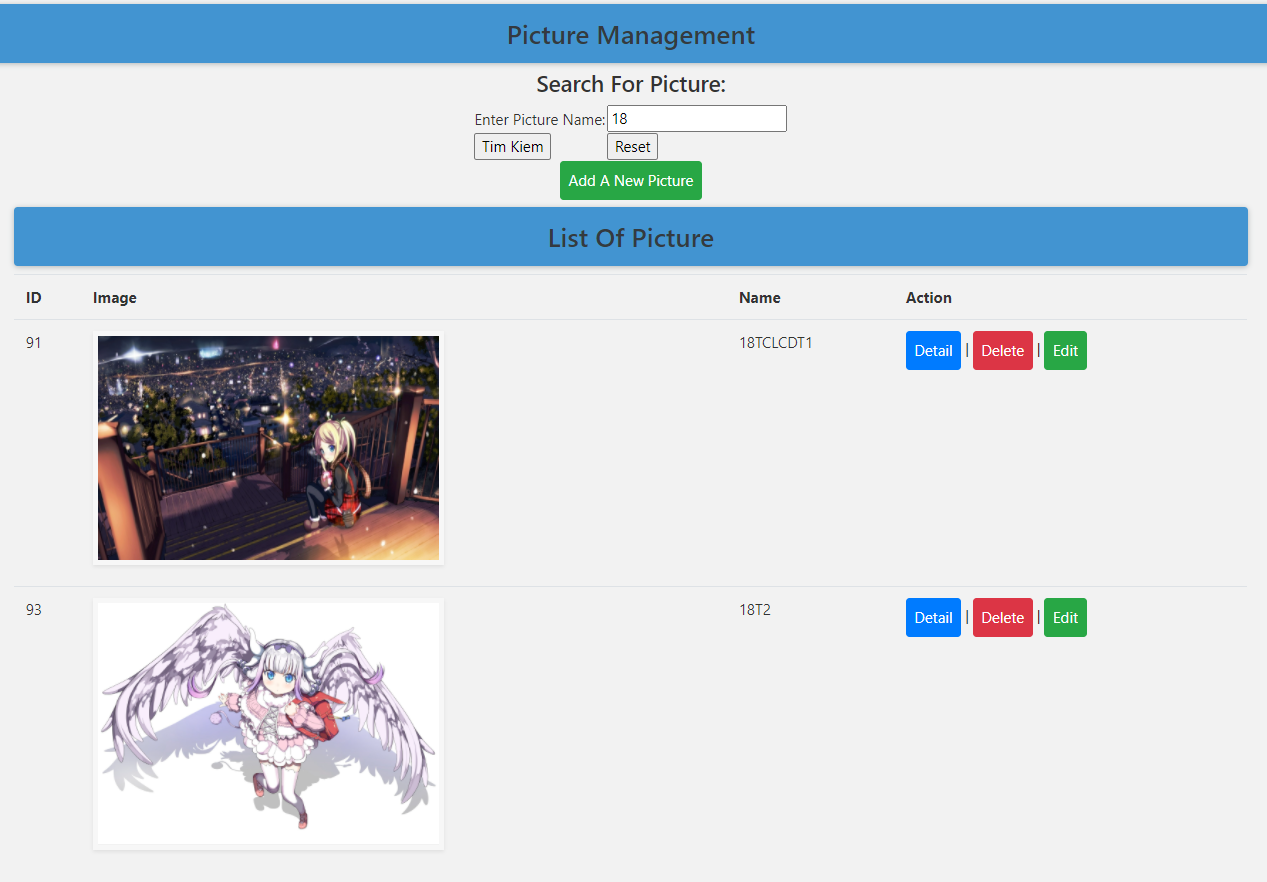
* Để sửa thông tin một ảnh người dùng chọn action “Edit” tương ứng với ảnh cần sửa, chương trình sẽ hiển thị form cập nhật thông tin ảnh, người dùng thông tin ảnh chỉnh sửa vào form. Chọn “Confirm” để tiến hành sửa thông tin ảnh đó.



*Hình 11: Form cập nhật thông tin ảnh*

1. Tìm kiếm ảnh

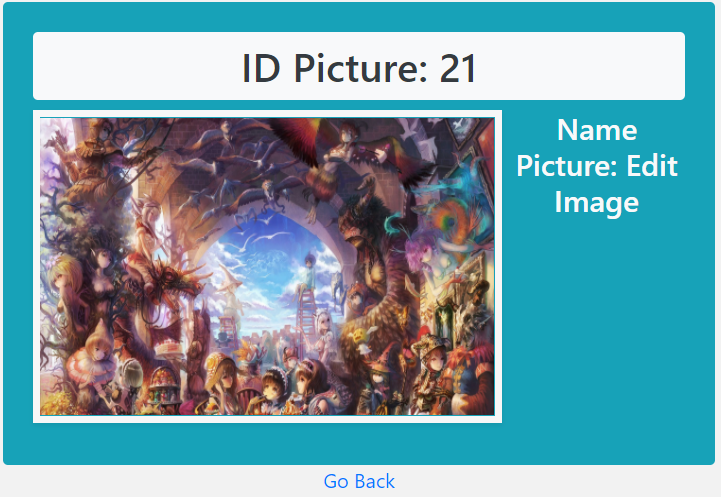
* Để tìm kiếm theo tên của ảnh, người dùng nhập tên ảnh vào form tìm kiếm, sau đó nhấn “Tìm kiếm” để tìm ảnh, hệ thống sẽ hiện thị danh sách ảnh tìm được theo tên ảnh.



*Hình 12: Danh sách ảnh tìm được*

1. Xem chi tiết ảnh

* Để xem chi tiết thông tin một ảnh, người dùng chọn action “Detail” tương ứng với ảnh đó, hệ thống sẽ hiển thị form chi tiết thông tin ảnh.



*Hình 13: Form chi tiết thông tin ảnh*

## **4.3 Nhận xét**

- Các chức năng được hoàn thiện tương đối đầy đủ .

- Bẫy lỗi khi người dùng nhập sai dữ liệu đầy đủ .

- Giao diện đẹp mắt gần với người dùng nên dễ sử dụng .

# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## **Kết quả đạt được:**

### ***Ưu điểm:***

* Ứng dụng có giao diện dễ dàng sử dụng gần với người dùng .
* Đầy đủ các chức năng cần thiết của chương trình .
* Tích hợp ngôn ngữ PHP, Java giúp tối ưu hệ thống cho người lập trình, nâng cao hiệu quả năng suất công việc. Dễ dàng tùy biến.
* Người lập trình có thể dễ dàng trao đổi cũng như chuyển giao công nghệ về sau.

### ***Nhược điểm:***

* Hệ thống vẫn còn nhiều bất cập như đáp ứng nhu cầu người dùng còn thấp, khả năng quản lý chưa thật sự tối ưu.
* Một số lệnh đang được thử nghiệm của giao thức chưa được hiện thực trong chương trình.
* Chưa nghiên cứu về việc bảo mật thông tin khi dùng giao thức FTP và HTTP.
* Các chức năng còn chưa rõ ràng. Vẫn còn khá nhiều bất ổn cần khắc phục.
* Code chưa sạch .
* Chưa đa dạng về các ngôn ngữ khi cài đặt .

## **Hướng phát triển:**

* Phát triển giao diện đẹp mắt kích thích người dùng.
* Hoàn thiện mọi yêu cầu mặt chức năng từ yêu cầu người dùng .
* Truyền ảnh giữa các người dùng.

**TÀI LIỆU THAM KHẢO**

**[1]** Slide môn Lập Trình Mạng của Giảng viên Mai Văn Hà – Đại Học Bách Khoa Đà Nẵng.

**[2]** Slide môn Công Nghệ Web của Giảng viên Mai Văn Hà – Đại Học Bách Khoa Đà Nẵng.

**[3]** Slide môn Lập Trình Java của Giảng viên Mai Văn Hà – Đại Học Bách Khoa Đà Nẵng.