

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

---



ISO 9001:2015

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**Sinh viên : Phạm Văn Hoàng Hà**  
**Giảng viên hướng dẫn: TS. Đỗ Văn Chiểu**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

-----

**TÌM HIỂU QR CODE VÀ ỨNG DỤNG CHO BÀI TOÁN  
CHECK-IN TỰ ĐỘNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY  
NGÀNH: CÔNG NGHỆ THÔNG TIN**

**Sinh viên : Phạm Văn Hoàng Hà  
Giảng viên hướng dẫn: TS. Đỗ Văn Chiểu**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

-----

**NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP**

Sinh viên: Phạm Văn Hoàng Hà

Mã SV: 1412101104

Lớp: CT1802

Ngành: Công nghệ thông tin

Tên đề tài: Tìm hiểu QRcode và ứng dụng cho bài toán check-in tự động

## NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

a. Nội dung:

b. Các yêu cầu cần giải quyết

2. Các số liệu cần thiết để thiết kế, tính toán

3. Địa điểm thực tập

## CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

### Người hướng dẫn thứ nhất:

Họ và tên:.....

Học hàm, học vị:.....

Cơ quan công tác:.....

Nội dung hướng dẫn:

.....  
.....  
.....  
.....

### Người hướng dẫn thứ hai:

Họ và tên:

Học hàm, học vị:

Cơ quan công tác:

Nội dung hướng dẫn:

.....  
.....  
.....  
.....

Đề tài tốt nghiệp được giao ngày 15 tháng 10 năm 2018

Yêu cầu phải hoàn thành trước ngày 07 tháng 01 năm 2019

Đã nhận nhiệm vụ: Đ.T.T.N

Sinh viên

Đã nhận nhiệm vụ: Đ.T.T.N

Cán bộ hướng dẫn Đ.T.T.N

*Hải Phòng, ngày .....tháng.....năm 2019*

Hiệu trưởng

***GS.TS.NGUYỄN Trần Hữu Nghị***

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Ngành: .....

Nội dung hướng dẫn: .....

**1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp**

.....  
.....  
.....  
.....  
.....  
.....

**2. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)**

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

**3. Ý kiến của giảng viên hướng dẫn tốt nghiệp**

Đạt

Không đạt

Điểm:.....

*Hải Phòng, ngày ..... tháng 01 năm 2019*

**Giảng viên hướng dẫn**

*(Ký và ghi rõ họ tên)*

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHẤM PHẢN BIỆN**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Ngành: .....

Đề tài tốt nghiệp: .....

.....

**1. Phần nhận xét của giảng viên chấm phản biện**

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

**2. Những mặt còn hạn chế**

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

**3. Ý kiến của giảng viên chấm phản biện**

Được bảo vệ  Không được bảo vệ  Điểm: .....

*Hải Phòng, ngày ..... tháng 01 năm 2019*

**Giảng viên chấm phản biện**

*(Ký và ghi rõ họ tên)*

---

**MỤC LỤC**

<b>LỜI CẢM ƠN</b> .....	1
<b>LỜI MỞ ĐẦU</b> .....	4
<b>CHƯƠNG 1: QR CODE VÀ TÌM HIỂU VỀ LẬP TRÌNH WEB TRÊN PHP/SQL</b> ...5	
<b>I. Giới thiệu về Qrcode</b> .....	5
1. Qrcode : Mã vạch thế hệ mới.....	5
2. Thuật toán Qrcode .....	6
<b>II. World Wide Web và HTML</b> .....	19
1.1. World Wide Web.....	19
1.1.1. Khái niệm.....	19
1.1.2. Cách tạo trang web .....	19
1.1.3. Trình duyệt web (web Client hay web Browser) .....	19
1.1.4. Webservice .....	20
1.2. Phân loại Web.....	20
1.2.1. HTML .....	21
1.2.2 Cấu trúc chung của một trang HTML.....	21
1.2.3 Các thẻ HTML cơ bản.....	22
<b>III. Ngôn ngữ PHP, MySQL</b> .....	23
2.1. Ngôn ngữ PHP .....	23
2.2. MySQL.....	30
<b>CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b> .....	35
<b>I . Phát biểu bài toán</b> .....	35
<b>II . Biểu đồ nghiệp vụ</b> .....	35
1 Biểu đồ ngữ cảnh hệ thống .....	35
2 Biểu đồ phân rã chức năng .....	36



---

3	Biểu đồ luồng dữ liệu.....	37
4	Mô hình liên kết thực thể (ER).....	39
	<b>CHƯƠNG 3 :THUẬT TOÁN , ỨNG DỤNG VÀ THỰC NGHIỆM.....</b>	<b>41</b>
	<b>I Thuật toán sinh mã qrcode .....</b>	<b>41</b>
1.1	Các bước mã hóa tạo một chuỗi nhị phân .....	41
	<b>II Môi trường thử nghiệm.....</b>	<b>43</b>
	<b>III Giao Diện.....</b>	<b>44</b>
3.1	Giao diện trang chủ.....	44
3.2	Quản lý thành viên.....	44
3.3	Giao diện của danh sách thành viên .....	44
3.4	Giao diện sửa thành viên .....	45
3.5	Giao diện thêm thành viên.....	46
3.6	Giao diện chức năng check-in .....	46
	<b>KẾT LUẬN.....</b>	<b>47</b>
	<b>TÀI LIỆU THAM KHẢO.....</b>	<b>49</b>

**LỜI CẢM ƠN**

Em xin chân thành cảm ơn tới tất cả thầy cô giáo trong Khoa Công nghệ thông tin - trường Đại Học Dân Lập Hải Phòng, chân thành cảm ơn các thầy giáo, cô giáo đã giảng dạy và truyền đạt những kiến thức bổ ích cho chúng em suốt thời gian học tập tại trường, giúp em có thêm nhiều kiến thức và hiểu rõ hơn các vấn đề mình nghiên cứu, để em có thể hoàn thành đồ án này.

Đặc biệt em xin chân thành cảm ơn thầy T.S Đỗ Văn Chiêu – Khoa Công nghệ thông tin - trường Đại Học Dân Lập Hải Phòng. Trong thời gian làm tốt nghiệp vừa qua, thầy đã giành nhiều thời gian quý báu và tâm huyết để hướng dẫn em hoàn thành đề tài này

Dưới đây là kết quả của quá trình tìm hiểu và nghiên cứu mà em đã đạt được trong thời gian vừa qua. Tuy có nhiều cố gắng học hỏi trau dồi để nâng cao kiến thức nhưng không thể tránh khỏi những sai sót. Em rất mong được những sự góp ý quý báu của các thầy giáo, cô giáo cũng như tất cả các bạn để bài luận văn của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

Hải Phòng, ngày .....tháng..... năm 2019

Sinh viên thực hiện

**Phạm Văn Hoàng Hà**

## LỜI MỞ ĐẦU

Với sự tăng trưởng ngày càng mạnh mẽ của công nghệ số thì việc kiểm tra dữ liệu, thông tin của một sản phẩm hoặc cá nhân tổ chức nào đó một cách nhanh chóng và thuận tiện là điều vô cùng quan trọng.

Vì lý do đó mã qrcode được ra đời nhằm đáp ứng tối đa những yêu cầu trên. Mã QR đang rất được giới marketing và quảng cáo chuộng dùng cho các ý tưởng quảng cáo tạp chí, trên các băng ghế công viên, xe buýt, đóng gói sản phẩm hay bất kỳ sản phẩm vật lý nào mà người tiêu dùng muốn tìm hiểu thêm thông tin về nó. Ngoài ra các công ty xí nghiệp hoặc các trường học cũng đều có thể sử dụng mã qrcode để lưu trữ thông tin của nhân viên cũng như sinh viên học sinh của mình để phục vụ cho việc điểm danh, điểm chuyên cần của nhân viên và học sinh của mình một cách nhanh chóng và thuận tiện nhất.

Vì vậy em đã chọn đề tài “Tìm hiểu qrcode và ứng dụng cho bài toán check-in tự động” với mục đích nghiên cứu tìm hiểu và xây dựng hệ thống tạo mã qrcode bằng thông tin người dùng, ứng dụng cho việc check-in thông tin một cách nhanh chóng thuận tiện nhất.

# CHƯƠNG 1: QR CODE VÀ TÌM HIỂU VỀ LẬP TRÌNH WEB TRÊN PHP/SQL

## I. Giới thiệu về Qrcode

### 1. Qrcode : Mã vạch thế hệ mới

QR Code bây giờ không hẳn là quá xa lạ, nó bắt đầu xuất hiện khắp nơi như nhãn bì sản phẩm, và gần như là phương thức nhận diện chủ yếu cho ứng dụng di động. Vậy QR Code là gì? . QR Code, viết tắt của *Quick response code* (tạm dịch "Mã phản hồi nhanh") hay còn gọi là mã vạch ma trận (matrix-barcode) là dạng mã vạch hai chiều (2D) có thể được đọc bởi một máy đọc mã vạch hay smartphone (điện thoại thông minh) có chức năng chụp ảnh (camera) với ứng dụng chuyên biệt để quét mã vạch.

Một mã QR có thể chứa đựng thông tin một địa chỉ web (URL), thời gian diễn ra một sự kiện, thông tin liên hệ (như vCard), địa chỉ email, tin nhắn SMS, nội dung ký tự văn bản hay thậm chí là thông tin định vị vị trí địa lý. Tùy thuộc thiết bị đọc mã QR mà bạn dùng khi quét, nó sẽ dẫn bạn tới một trang web, gọi đến một số điện thoại, xem một tin nhắn...

Điểm khác nhau giữa mã QR và mã vạch truyền thống là lượng dữ liệu chúng nắm giữ hay chia sẻ. Các mã vạch truyền thống có các đường vạch thẳng dài một chiều và chỉ có thể lưu giữ 20 số chữ số, trong khi các mã QR hai chiều có thể lưu giữ thông tin hàng ngàn ký tự chữ số. Mã QR nắm giữ nhiều thông tin hơn và tính chất dễ sử dụng sẽ giúp ích rất nhiều cho người dùng trong mọi lĩnh vực. Ngoài ra, QR code có lợi thế hơn do có thể đọc được cả hai chiều cả ngang và dọc và từ bất kỳ hướng nào mà không bị ảnh hưởng bởi chất liệu hay nền mà nó đang sử dụng. Chính vì thế mà QR code đang là xu hướng hiện nay và dần thay thế cho Bar code truyền thống

Qrcode có 2 loại chính : động và tĩnh.

-Qrcode động, hay còn gọi là “Qrcode trực tiếp”: sau khi được khởi tạo, liên kết chính của chúng có thể được chỉnh sửa mà không cần can thiệp vào code. Khi được quét, code động sẽ chuyển hướng bạn đến máy chủ, nơi lưu trữ thông tin cụ thể trong một cơ sở dữ liệu và được lập trình để tương tác cụ thể với

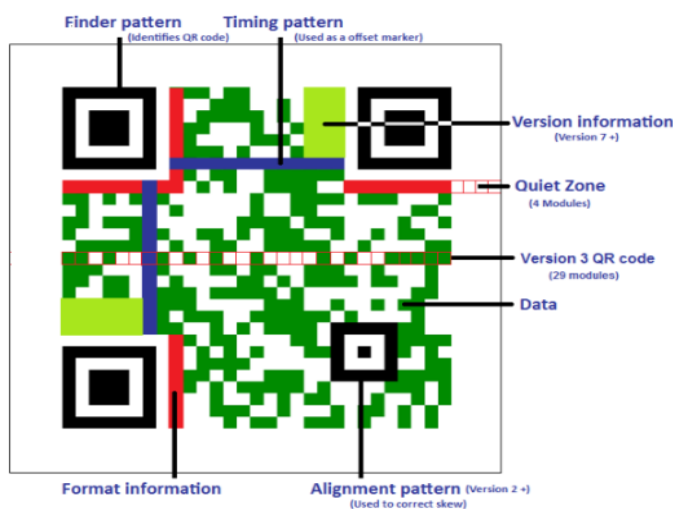
hình ảnh được quét đó. Code động tiện lợi hơn vì chúng được tạo ra áp dụng một lần, và bạn có thể thay đổi các thông tin được hiển thị khi đó được quét như các liên kết(domain,web) để hiển thị thông tin.

-QRcode tĩnh: không giống như các code động, code tĩnh có thể lưu trữ các thông tin trực tiếp qua hình thức văn bản hoặc dẫn đến các trang web mà không cần chuyển hướng qua liên kết thứ cấp. Điều này có nghĩa rằng các thông tin hoặc liên kết không thể thay đổi hoặc chỉnh sửa(bạn sẽ phải tạo ra một code tĩnh mới mỗi lần muốn cập nhật) . Code tĩnh rất lý tưởng cho những người muốn mã chứa thông tin của họ vĩnh viễn(ví dụ, thông tin về ngày sinh và nhóm máu).



Hình ảnh mã QR Code

## 2. Thuật toán QRcode



### 2.1 Tạo một chuỗi nhị phân:

Bước đầu tiên để tạo một mã QR code là tạo một chuỗi nhị phân bao gồm dữ liệu và thông tin về chế độ mã hóa, cũng như độ dài của dữ liệu.

Trong phần này chúng ta sẽ tìm hiểu mã hóa chuỗi HELLO WORLD ở QR code phiên bản 1, với mức độ sửa lỗi là Q.

*Bước 1: Mã hóa Bộ chỉ chế độ (Mode Indicator)*

Một bộ chỉ chế độ là một chuỗi 4 bit thể hiện chế độ dữ liệu bạn đang sử dụng: kiểu số, kiểu chuỗi, kiểu nhị phân hoặc Nhật ngữ.

Bit string	Data mode
0001	Numeric Mode
0010	Alphanumeric Mode
0100	Binary Mode
1000	Japanese Mode

Mode Indicator

Với chuỗi HELLO WORLD, là kiểu chuỗi, vậy nên Bộ chỉ chế độ sẽ là 0010.

Chuỗi nhị phân: 0010

*Bước 2: Mã hóa độ dài của dữ liệu*

Trong bước này, chúng ta xác định có bao nhiêu ký tự trong thông điệp, và chuyển độ dài đó thành một số nhị phân.

Với chuỗi HELLO WORLD, có 11 ký tự kể cả khoảng trắng. Ta chuyển 11 sang nhị phân, được 1011.

Xem danh sách ở dưới. Khi mã hóa độ dài của dữ liệu, chúng ta mã hóa nó sử dụng một số lượng bit đặc tả. Như đã nói ở đầu, chúng ta sử dụng QR code phiên bản 1, vậy nên chúng ta cần sử dụng 9 bit để mã hóa độ dài dữ liệu. Ta thêm số 0 vào đầu chuỗi cho đến khi đủ 9 bit: 000001011.

*Phiên bản 1 tới 9*

- Kiểu số: 10 bits
- Kiểu chuỗi: 9 bits
- Kiểu nhị phân: 8 bits
- Kiểu Nhật ngữ: 8 bits

*Phiên bản 10 tới 26*

- Kiểu số: 12 bits

- Kiểu chuỗi: 11 bits
- Kiểu nhị phân: 16
- Kiểu Nhật ngữ: 10 bits

*Phiên bản 27 tới 40*

- Kiểu số: 14 bits
- Kiểu chuỗi: 13 bits
- Kiểu nhị phân: 16 bits
- Kiểu Nhật ngữ: 12 bits

Tới đây chuỗi nhị phân chúng ta là: 0010 000001011.

*Bước 3: Mã hóa dữ liệu*

Để mã hóa chuỗi ký tự, ta chia chuỗi thành các cặp ký tự: HE,LL,O,WO,RL,D.

Với mỗi cặp ký tự, ta lấy giá trị mã ASCII của ký tự đầu nhân với 45, sau đó cộng với giá trị mã ASCII của ký tự sau. Sau đó chuyển kết quả thu được thành một chuỗi nhị phân 11 bit.

Đối với các ký tự không có cặp, tao lấy giá trị mã ASCII của nó rồi chuyển thành chuỗi nhị phân 6 bit.

Đối với các chế độ khác: kiểu số, kiểu nhị phân, kiểu Nhật ngữ, tao sử dụng phương thức khác để mã hóa dữ liệu.

HE	LL	O(space)	WO	RL	D
$(45 \cdot 17) + 14$	$(45 \cdot 21) + 21$	$(45 \cdot 24) + 36$	$(45 \cdot 32) + 24$	$(45 \cdot 27) + 21$	13
779	966	1116	1464	1236	13
0010 000001011	01100001011	01111000110	10001011100	10110111000	10011010100
					001101

*Bước 4: Hoàn thành các bit*

Với chuỗi bit thu được ở trên, ta phải chắc chắn rằng nó có đủ độ dài. Điều này phụ thuộc vào việc có bao nhiêu bit chúng ta cần để tạo phiên bản và sửa lỗi.

Với QR code phiên bản 1, sửa lỗi mức độ Q. Chúng ta phải tạo một chuỗi 104 bit. Nếu chuỗi bit nhỏ ít hơn 104, ta phải thêm 4 số 0 vào cuối chuỗi. Nếu thêm 4 số 0 được một chuỗi nhiều hơn 104, ta chỉ cần thêm số 0 cho tới khi nào đủ 104 bit.

Chuỗi của chúng ta có 59 bit, ta thêm 4 số 0 vào cuối chuỗi. (Còn nếu chuỗi có 102 bit thì ta thêm 2 số 0).

Chuỗi của ta bây giờ là:

```
0010 000001011 01100001011 01111000110 10001011100 10110111000
10011010100 001101 0000
```

*Bước 5:* Giới hạn chuỗi thành các chuỗi con 8 bit, sau đó thêm các số 0 nếu cần.

Bước này ta chia chuỗi thành các nhóm 8 bit.

Ở chuỗi con cuối cùng, nếu chưa đủ 8 bit thì ta thêm vào sau đó các số 0 cho tới khi đủ. Chuỗi của chúng ta thêm 2 số 0 vào sau chuỗi con cuối cùng.

```
00100000 01011011 00001011 01111000 11010001 01110010 11011100
01001101 01000011 01000000
```

*Bước 6:* Thêm các từ vào cuối nếu chuỗi quá ngắn

**Nếu chuỗi bit của chúng ta vẫn chưa đủ dài, có hai chuỗi con đặc biệt:** 11101100 và 00010001. Ta thêm thay phiên hai chuỗi con này vào chuỗi chuỗi của chúng ta.

Chuỗi của chúng ta chỉ có 10 chuỗi con, cần phải thêm vào 3 chuỗi con nữa. Bắt đầu thêm 11101100, sau đó thêm 00010001, rồi tới 11101100. Nếu cần nhiều hơn thì cứ thêm lần lượt như vậy cho tới khi đủ số chuỗi con.

Chuỗi của chúng ta bây giờ:

```
00100000 01011011 00001011 01111000 11010001 01110010 11011100
01001101 01000011 01000000 11101100 00010001 11101100
```

## 2.2 Tạo bộ mã sửa lỗi

*Bước 1:* Chọn mức sửa lỗi



Bước đầu tiên để tạo mã sửa lỗi là xác định có bao nhiêu từ để tạo cho phiên bản QR và mức độ sửa lỗi. Xem bảng sửa lỗi để biết thêm thông tin chi tiết

Cấp độ sửa lỗi	Khả năng sửa lỗi
L	Phục hồi 7% dữ liệu
M	Phục hồi 15% dữ liệu
Q	Phục hồi 25% dữ liệu
H	Phục hồi 30% dữ liệu

*Bước 2:* Tạo một thông điệp đa thức

Chúng ta chuyển 13 khối dữ liệu ở trên thành dạng thập phân:

00100000 01011011 00001011 01111000 11010001 01110010 11011100  
01001101 01000011 01000000 11101100 00010001 11101100

Ta được:  
32, 91, 11, 120, 209, 114, 220, 77, 67, 64, 236, 17, 236

Các số này sẽ là hệ số của đa thức. Ta có:

$$32x^{25} + 91x^{24} + 11x^{23} + 120x^{22} + 209x^{21} + 114x^{20} + 220x^{19} + 77x^{18} + 67x^{17} + 64x^{16} + 236x^{15} + 17x^{14} + 236x^{13}$$

Lũy thừa của số hạng đầu tiên được tính:

Số khối dữ liệu + số mã sửa lỗi - 1

Ở đây là  $13+13-1=25$ . Vậy nên số hạng đầu tiên là  $32x^{25}$ .

*Bước 3:* Tạo bộ tạo đa thức

Tiếp theo ta sẽ tạo một bộ tạo đa thức. Ta chia thông điệp đa thức bởi bộ tạo này để tạo mã sửa lỗi.

Các bộ tạo đa thức đến từ một cái gì đó gọi là một trường hữu hạn, cũng được biết đến như là một trường Galois. Các mã QR sử dụng một trường Galois có 256 yếu tố, có nghĩa là, mục đích của chúng ta, rằng những con số mà chúng ta sẽ được giao dịch với sẽ luôn luôn tối đa là 255 và ít nhất là 0.

Để thực hiện việc phân chia đa thức, chúng ta sẽ được chuyển đổi qua lại giữa các ký hiệu  $\alpha$  (alpha) và các số nguyên. Alpha và các giá trị số nguyên từ bảng log và antilog.

Bộ tạo đa thức luôn luôn có dạng  $(x - \alpha) (x - \alpha^2) \dots (x - \alpha^t)$ , , t tương đương với số lượng mã sửa lỗi cần thiết trừ đi 1. Chúng ta cần 13 mã sửa lỗi, do đó, t trong trường hợp này là 12.

Ta thực hiện triển khai tất cả các thành phần  $(x - \alpha)$  cho tới khi chúng ta được đa thức. Công việc này thực hiện bằng tay khá phức tạp, sử dụng generator polynomial tool. Ở đây chúng ta tạo 13 mã sửa lỗi:

$$\alpha^0 x^{25} + \alpha^{74} x^{24} + \alpha^{152} x^{23} + \alpha^{176} x^{22} + \alpha^{100} x^{21} + \alpha^{86} x^{20} + \alpha^{100} x^{19} + \alpha^{106} x^{18} + \alpha^{104} x^{17} + \alpha^{130} x^{16} + \alpha^{218} x^{15} + \alpha^{206} x^{14} + \alpha^{140} x^{13} + \alpha^{78} x^{12}$$

Dưới đây, tôi cho thấy từng bước của việc phân chia đa thức cần thiết để tạo ra mã sửa lỗi

Kết quả của bước này ta được mã sửa lỗi:

168 72 22 82 217 54 156 0 46 15 180 122 16

Ta đặt đoạn mã sửa lỗi sau đoạn mã dữ liệu:

32 91 11 120 209 114 220 77 67 64 236 17 236 **168 72 22 82 217 54 156 0**  
**46 15 180 122 16**

Chuyển qua nhị phân:

00100000 01011011 00001011 01111000 11010001 01110010 11011100  
01001101 01000011 01000000 11101100 00010001 11101100 10101000  
01001000 00010110 01010010 11011001 00110110 10011100 00000000  
00101110 00001111 10110100 01111010 00010000

### 2.3 Chọn mẫu mặt nạ phù hợp

Bây giờ chúng ta đã mã hóa các dữ liệu, tất cả những gì còn lại là chọn mẫu mặt nạ tốt nhất. Một mẫu mặt nạ thay đổi theo các bit, theo một quy tắc cụ thể. Các mẫu mặt nạ được định nghĩa trong tiêu chuẩn mã QR.

Có 8 mẫu mặt nạ. Sử dụng mô hình mặt nạ cho phép chúng tôi để tạo ra tám mã QR khác nhau và sau đó chọn một trong đó sẽ được dễ dàng nhất cho một đầu đọc QR để quét. Một đặc biệt mã QR có thể có các mẫu nhất định hoặc những đặc điểm mà làm cho nó khó khăn cho thiết bị đọc QR

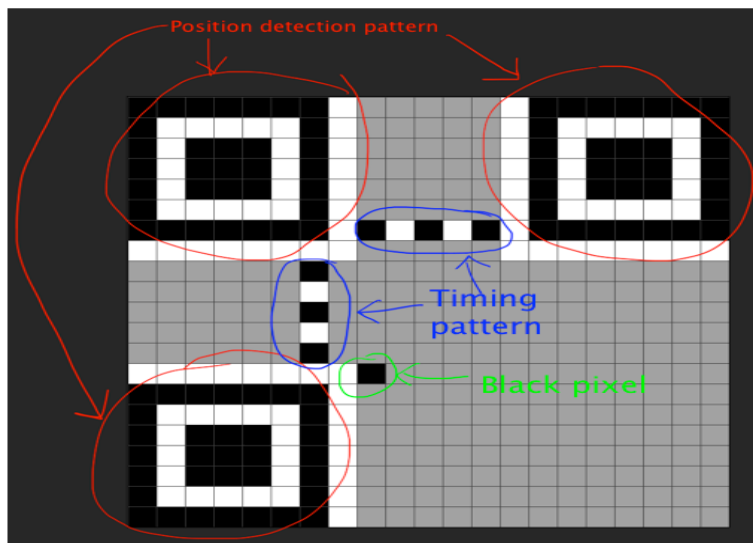
chính xác quét mã QR. Ví dụ, nếu điểm ảnh của cùng một màu sắc xảy ra gần nhau, một đầu đọc QR có thể gặp khó khăn khi đọc chính xác các điểm ảnh.

Mỗi mô hình mặt nạ tạo ra một mã QR code khác nhau. Sau khi chúng ta tạo ra các mã khác nhau QR trong nội bộ, chúng tôi cung cấp cho mỗi một số điểm xử phạt theo các quy tắc quy định trong tiêu chuẩn mã QR. Sau đó, chúng tôi cho ra mã QR có số điểm tốt nhất.

*Bước 1* : Tạo mã QR code:

Với QR code phiên bản 1, ta cần một ma trận  $21 \times 21$ .

Tất cả các mã QR code đều có một vài mẫu nhận diện chắc chắn phải đc thể hiện. Ở hình bên dưới, có ba vị trí nhận diện được đặt ở trên cùng bên trái, trên cùng bên phải và dưới cùng bên trái của ma trận. Các mã QR code còn có một pixel đen ở bên phải của vị trí nhận diện bên dưới cùng bên trái. Ngoài ra còn có mẫu timing.



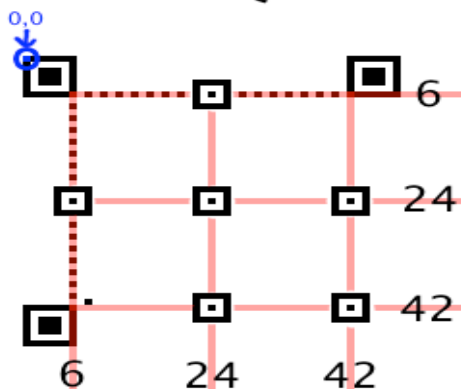
### Vị trí điều chỉnh mô hình

Từ ví dụ này là chỉ mã hóa một phiên bản 1 mã QR, chúng tôi không cần thêm bất kỳ mô hình điều chỉnh vị trí. Tuy nhiên, nếu chúng ta đã mã hóa một mã QR phiên bản 2 hoặc lớn hơn, chúng tôi sẽ thêm các mẫu điều chỉnh vị trí ma trận. Bảng dưới đây liệt kê các tọa độ của nơi để đặt các mô hình điều chỉnh vị trí. Xem ví dụ ở bảng dưới đây.

QR Version 2	6	18				
QR Version 3	6	22				
QR Version 4	6	26				
QR Version 5	6	30				
QR Version 6	6	34				
QR Version 7	6	22	38			
QR Version 8	6	24	42			
QR Version 9	6	26	46			
QR Version 10	6	28	50			
QR Version 11	6	30	54			
QR Version 12	6	32	58			

Ví dụ, hình ảnh dưới đây cho thấy các mô hình điều chỉnh vị trí cho phiên bản mã QR 8. Mỗi mô hình điều chỉnh vị trí là một điểm ảnh màu đen bao quanh bởi các điểm ảnh màu trắng được bao quanh bởi các điểm ảnh màu đen. Theo bảng trên, cho phiên bản mã QR 8, vị trí các mô hình điều chỉnh 6, 24, và 42. Như ta thấy trong hình ảnh, mô hình điều chỉnh vị trí được đặt ở tất cả các giao điểm của ba con số này, ngoại trừ đã có lớn hơn mô hình phát hiện vị trí không đặt mô hình điều chỉnh vị trí trên hàng đầu của các mô hình phát hiện vị trí trong góc của mã QR. Như hình :

Version 8 QR Code

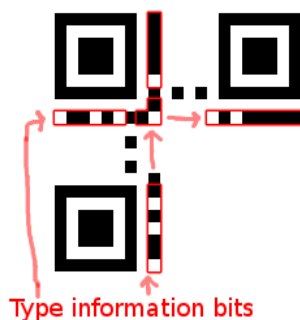


**Thêm thông tin loại**

Các thông tin về mức độ sửa lỗi và mô hình mặt nạ được mã hóa ở dạng dải với các cạnh của các mẫu phát hiện vị trí.

Bảng sau đây hiển thị các bit thông tin các loại được yêu cầu cho các cấp, sửa lỗi khác nhau và các mẫu mặt nạ.

ECC Level	Mask Pattern	Type Information Bits
L	0	111011111000100
L	1	111001011110011
L	2	111110110101010
L	3	111100010011101
L	4	110011000101111
L	5	110001100011000
L	6	110110001000001
L	7	110100101110110
M	0	101010000010010
M	1	101000100100101
M	2	101111001111100
M	3	101101101001011
M	4	100010111111001
M	5	100000011001110
M	6	100111110010111
M	7	100101010100000
Q	0	011010101011111
Q	1	011000001101000
Q	2	011111100110001
Q	3	011101000000110



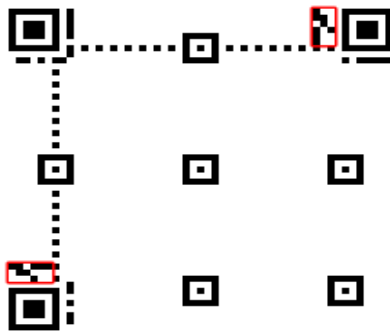
Hình: Hiển thị các bit thông tin

Trong ví dụ Hello World mô hình mặt nạ tốt nhất là 0 (điều này được giải thích dưới đây). Chúng ta đang sử dụng mức độ hiệu chỉnh lỗi Q, vì vậy các chuỗi loại thông tin là 011010101011111. Như trong hình trên, các

chuỗi loại thông tin được hiển thị hai lần. Một trong số đó là dưới hai mô hình phát hiện vị trí hàng đầu, bắt đầu từ bên trái. Nó bỏ qua quá khứ mô hình thời gian và phần dữ liệu của lưới. Điều thứ hai là quyền của các mẫu phát hiện vị trí, bắt đầu từ phía dưới, bỏ qua dấu chấm đen, phần dữ liệu, và mô hình thời gian. Cả hai sọc cùng một chuỗi: 011010101011111.

**Thêm thông tin phiên bản**

Nếu mã QR của bạn là phiên bản 7 hoặc lớn hơn, bạn cần phải thêm các bit thông tin phiên bản ma trận. Đây là những được đặt ở bên trái của mô hình trên bên phải phát hiện vị trí và trên các mô hình phát hiện vị trí phía dưới bên trái, như thể hiện trong hình ảnh dưới đây, trong đó có một phiên bản 9 Mã QR. Như hình :



Bảng sau đây cho thấy một danh sách của các bit thông tin phiên bản cần thiết cho mỗi phiên bản. Những bit được đặt trong cấu hình sau đây trong các vị trí được chỉ định trong hình :

00	03	06	09	12	15
01	04	07	10	13	16
02	05	08	11	14	17

00	01	02
03	04	05
06	07	08
09	10	11
12	13	14
15	16	17

Trong ví dụ của chúng ta, mã QR là nhỏ hơn so với phiên bản 7, vì vậy chúng ta không cần thêm các bit thông tin phiên bản mã.

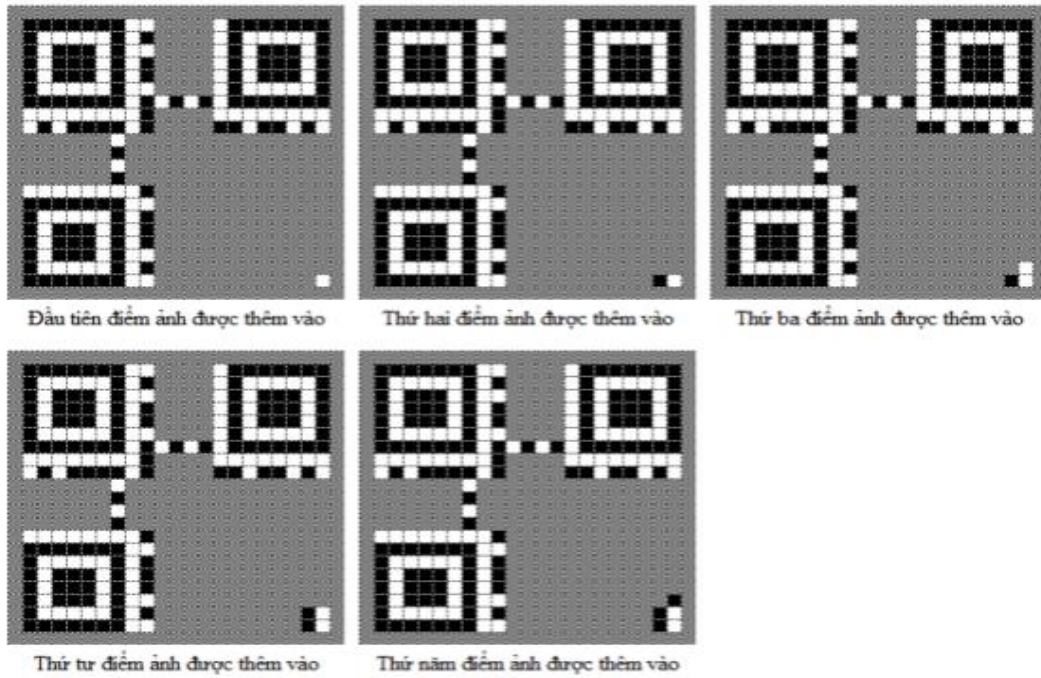
Version	Bits for Version Information
7	001010010011111000
8	000111101101000100
9	100110010101100100
10	011001011001010100
11	011011111101110100
12	001000110111001100
13	111000100001101100
14	010110000011011100
15	000101001001111100
16	000111101101000010

### Thêm dữ liệu bist

Cuối cùng, chúng ta thêm các bit dữ liệu. Trong bước này, chúng ta thay đổi chút tùy thuộc vào mô hình mặt nạ mà chúng ta đang sử dụng trước khi thêm chút thực tế, chúng ta sử dụng các mẫu mặt nạ để quyết định xem có nên thay đổi màu sắc của các bit hiện tại hay không.

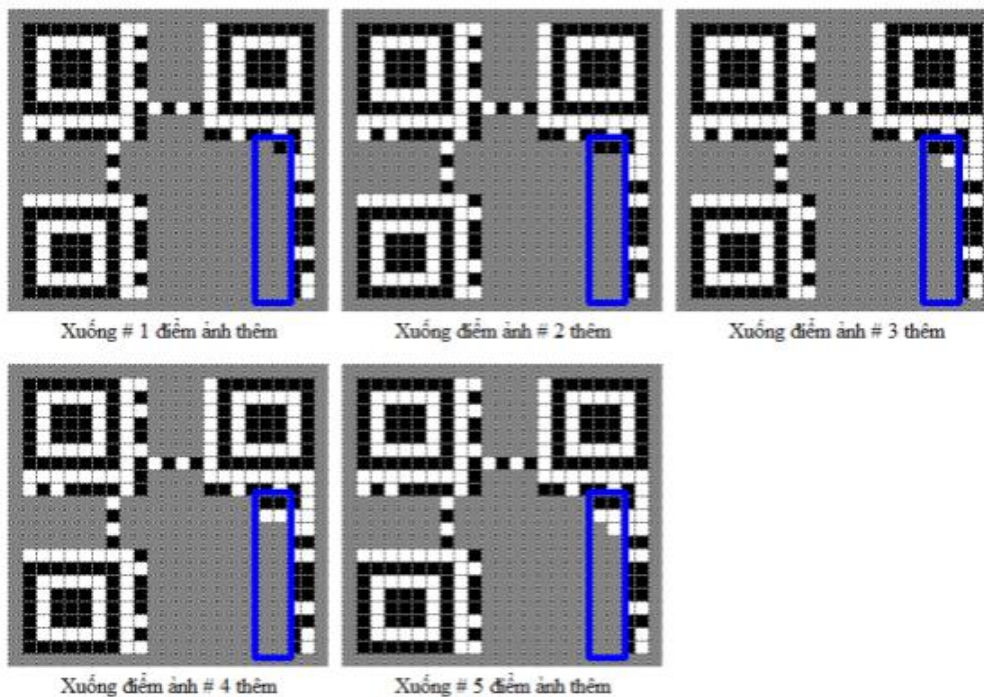
### Sắp xếp theo cột từ dưới lên

Các bit dữ liệu được thêm vào theo một thứ tự cụ thể. Các điểm ảnh đầu tiên được thêm vào phía dưới bên phải của mã QR. Điều thứ hai được đặt ở bên trái đó. Hai tiếp theo được đặt trên đầu những người đầu tiên theo thứ tự. Hai cột điểm ảnh này vẫn tiếp tục đi lên theo cách này, bỏ qua bất kỳ điểm ảnh đã được thiết lập, cho đến khi nó đạt đến hàng đầu của mã QR.



### Xuống cột

Một khi nó đạt đến đỉnh, một cột hai-pixel mới bắt đầu, thời gian này sẽ đi xuống

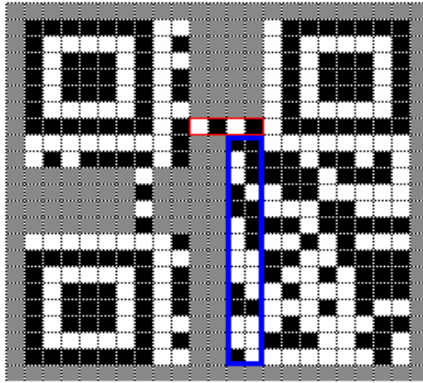


Khi bạn đạt đến dưới cùng của lưới, thay đổi hướng một lần nữa, và tiếp tục trong mô hình này giống như đường đi của con rắn.

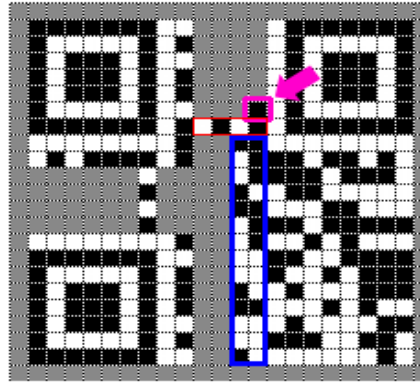
### Chắc chắn bỏ qua mẫu timing

Khi bạn thêm pixels, hãy chắc chắn để bỏ qua các mô hình thời gian.



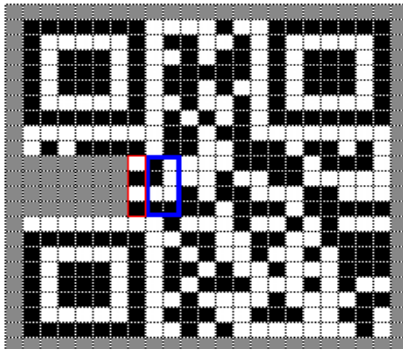


Khi bạn đạt đến các mô hình thời gian đầu tiên ...

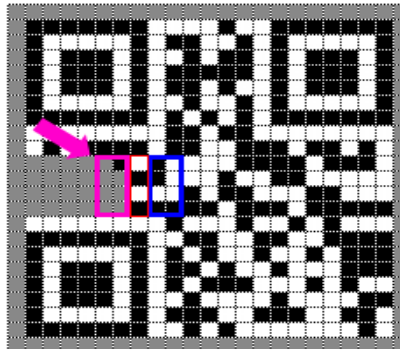


... Chắc chắn sẽ bỏ qua nó.

Và các mẫu timing ở chiều dọc:



Khi bạn đạt đến các mô hình thời gian thứ hai ...



... Chắc chắn sẽ bỏ qua nó.

### Hoàn tất QRcode

Tiếp tục theo cách này cho đến khi tất cả của các điểm ảnh dữ liệu đã được đặt. Sau đó, các mã QR là hoàn tất. Tuy nhiên, các đặc điểm kỹ thuật mã QR yêu cầu bạn sử dụng các mô hình mặt nạ tốt nhất có thể.

Hình ảnh sau khi hoàn thiện :



## ***II. World Wide Web và HTML***

### **1.1. World Wide Web**

#### **1.1.1. Khái niệm**

World Wide Web (WWW) hay còn gọi là web là một dịch vụ phổ biến nhất hiện nay trên Internet, 85% các giao dịch trên Internet ước lượng thuộc về WWW. Ngày nay số website trên thế giới đã đạt tới con số khổng lồ. WWW cho phép truy xuất thông tin văn bản, hình ảnh, âm thanh, video trên toàn thế giới. Thông qua website, các quý công ty có thể giảm thiểu tối đa chi phí in ấn và phân phát tài liệu cho khách hàng ở nhiều nơi.

#### **1.1.2. Cách tạo trang web**

Có nhiều cách để tạo trang web, có thể tạo trang web trên bất kì chương trình xử lí văn bản nào:

- Tạo web bằng cách viết mã nguồn bởi một trình soạn thảo văn bản như: Notepad, WordPad,... là những chương trình soạn thảo văn bản có sẵn trong Window.

- Thiết kế bằng cách dùng web Wizard và công cụ của Word 97, Word 2000.
- Thiết kế web bằng các phần mềm chuyên nghiệp: FrontPage, Dreamweaver, Nescape Editor,.... Phần mềm chuyên nghiệp như DreamWeaver sẽ giúp thiết kế trang web dễ dàng hơn, nhanh chóng hơn, phần lớn mã lệnh HTML sẽ có sẵn trong phần code.

Để xây dựng một ứng dụng web hoàn chỉnh và có tính thương mại, cần kết hợp cả Client Script (kịch bản trình khách) và Server Script (kịch bản trên trình chủ) với một loại cơ sở dữ liệu nào đó, chẳng hạn như: MS Access, SQL Server, MySQL, Oracle,.... Khi muốn triển khai ứng dụng web trên mạng, ngoài các điều kiện về cấu hình phần cứng, cần có trình chủ web thường gọi là web Server.

#### **1.1.3. Trình duyệt web (web Client hay web Browser)**

Trình duyệt Web là công cụ truy xuất dữ liệu trên mạng, là phần mềm giao diện trực tiếp với người sử dụng. Nhiệm vụ của Web Browser là nhận các yêu cầu của người dùng, gửi các yêu cầu đó qua mạng tới các Web Server và nhận các dữ liệu cần thiết từ Server để hiển thị lên màn hình. Để sử dụng dịch vụ WWW, Client cần có một chương trình duyệt Web, kết nối vào Internet

thông qua một ISP. Các trình duyệt thông dụng hiện nay là: Microsoft Internet Explorer, Google Chrome, Mozilla FireFox....

#### 1.1.4. Webservice

Webservice là một máy tính được nối vào Internet và chạy các phần mềm được thiết kế. Webservice đóng vai trò một chương trình xử lý các nhiệm vụ xác định, như tìm trang thích hợp, xử lý tổ hợp dữ liệu, kiểm tra dữ liệu hợp lệ.... Webservice cũng là nơi lưu trữ cơ sở dữ liệu, là phần mềm đảm nhiệm vai trò server cung cấp dịch vụ Web.

Webservice hỗ trợ các công nghệ khác nhau:

- IIS (Internet Information Service): Hỗ trợ ASP, mở rộng hỗ trợ PHP.
- Apache: Hỗ trợ PHP. - Tomcat: Hỗ trợ JSP (Java Servlet Page).

#### 1.2. Phân loại Web

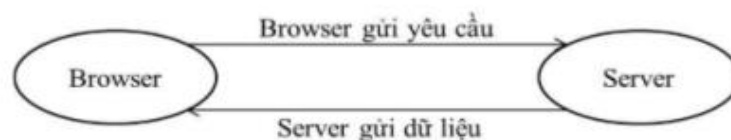
##### 1) Web tĩnh:

Tài liệu được phân phát rất đơn giản từ hệ thống file của Server. Định dạng các trang web tĩnh là các siêu liên kết, các trang định dạng Text, các hình ảnh đơn giản.

Ưu điểm: CSDL nhỏ nên việc phân phát dữ liệu có hiệu quả rõ ràng, Server có thể đáp ứng nhu cầu Client một cách nhanh chóng. Ta nên sử dụng Web tĩnh khi không thay đổi thông tin trên đó.

Nhược điểm: Không đáp ứng được yêu cầu phức tạp của người sử dụng, không linh hoạt,...

Hoạt động của trang Web tĩnh được thể hiện như sau:



Hình 1.2.1

## 2) Website động:

Về cơ bản nội dung của trang Web động như một trang Web tĩnh, ngoài ra nó còn có thể thao tác với CSDL để đáp ứng nhu cầu phức tạp của một trang Web. Sau khi nhận được yêu cầu từ Web Client, chẳng hạn như một truy vấn từ một CSDL đặt trên Server, ứng dụng Internet Server sẽ truy vấn CSDL này, tạo một trang HTML chứa kết quả truy vấn rồi gửi trả cho người dùng.

- Hoạt động của Web động:



Hình 1.2.2

### 1.2.1. HTML

Trang Web là sự kết hợp giữa văn bản và các thẻ HTML. HTML là chữ viết tắt của HyperText Markup Language được hội đồng World Wide Web Consortium (W3C) quy định. Một tập tin HTML chẳng qua là một tập tin bình thường, có đuôi .html hoặc .htm. HTML giúp định dạng văn bản trong trang Web nhờ các thẻ. Hơn nữa, các thẻ html có thể liên kết t hoặc một cụm t với các tài liệu khác trên Internet. Đa số các thẻ HTML có dạng thẻ đóng mở. Thẻ đóng dùng chung t lệnh giống như thẻ mở, nhưng thêm dấu xiên phải (/). Ngôn ngữ HTML qui định cú pháp không phân biệt chữ hoa chữ thường. Ví dụ, có thể khai báo hoặc . Không có khoảng trắng trong định nghĩa thẻ.

### 1.2.2 Cấu trúc chung của một trang HTML

```

<html>
<head>
  <title>Tiêu đề của trang Web</title>
</head>
<body>
  <!--các thẻ html và nội dung sẽ hiển thị-->
</body>
</html>
  
```

### 1.2.3 Các thẻ HTML cơ bản

- Thẻ `<head>...</head>` tạo đầu mục trang.
- Thẻ `<title>...</title>` tạo tiêu đề trang trên thanh tiêu đề, đây là thẻ bắt buộc. Thẻ title cho phép trình bày chuỗi trên thanh tựa đề của trang web mỗi khi trang Web đó được duyệt trên trình duyệt web.
  
- Thẻ `<body>...</body>` tất cả các thông tin khai báo trong thẻ đều có thể xuất hiện trên trang web. Những thông tin này có thể nhìn thấy trên trang web.
- Thẻ `<p>...</p>` tạo một đoạn mới.
- Thẻ `<font>...</font>` thay đổi phông chữ, kích cỡ và màu kí tự.
- Thẻ `<table>...</table>` đây là thẻ định dạng bảng trên trang web. Sau khi khai báo thẻ này, phải khai báo các thẻ hàng và thẻ cột cùng với các thuộc tính của nó.
- Thẻ `<img/>` cho phép chèn hình ảnh vào trang web. Thẻ này thuộc loại thẻ không có thẻ đóng.
- Thẻ `<a>...</a>` là loại thẻ dùng để liên kết giữa các trang web hoặc liên kết đến địa chỉ Internet, Mail hay Intranet (URL) và địa chỉ trong tập tin trong mạng cục bộ (UNC).
- Thẻ `<input/>` cho phép người dùng nhập dữ liệu hay chỉ thị thực thi một hành động nào đó, thẻ Input bao gồm các loại thẻ như: text, password, submit, button, reset, checkbox, radio, hidden, image.
- Thẻ `<textarea>...</textarea>` cho phép người dùng nhập liệu với rất nhiều dòng. Với thẻ này không thể giới hạn chiều dài lớn nhất trên trang Web.
- Thẻ `<select>...</select>` cho phép người dùng chọn phần tử trong tập phương thức đã được định nghĩa trước. Nếu thẻ `<select>` cho phép người dùng chọn một phần tử trong danh sách phần tử thì thẻ `<select>` sẽ giống như combobox. Nếu thẻ `<select>` cho phép người dùng chọn nhiều phần tử cùng một lần trong danh sách phần tử, thẻ `<select>` đó là dạng listbox.

- Thẻ `<form>...</form>` khi muốn submit dữ liệu người dùng nhập từ trang web phía Client lên phía Server, có hai cách để làm điều này ứng với hai phương thức POST và GET trong thẻ form. Trong một trang web có thể có nhiều thẻ `<form>` khác nhau, nhưng các thẻ `<form>` này không được lồng nhau, mỗi thẻ form sẽ được khai báo hành động (action) chỉ đến một trang khác.

### III. Ngôn ngữ PHP, MySQL

#### 2.1. Ngôn ngữ PHP

##### 2.1.1 Khái niệm PHP

PHP là chữ viết tắt của “Personal Home Page” do Rasmus Lerdorf tạo ra năm 1994. Vì tính hữu dụng của nó và khả năng phát triển, PHP bắt đầu được sử dụng trong môi trường chuyên nghiệp và nó trở thành “PHP:Hypertext Preprocessor”. Thực chất PHP là ngôn ngữ kịch bản nhúng trong HTML, nói một cách đơn giản đó là một trang HTML có nhúng mã PHP, PHP có thể được đặt rải rác trong HTML. PHP là một ngôn ngữ lập trình được kết nối chặt chẽ với máy chủ, là một công nghệ phía máy chủ (Server-Side) và không phụ thuộc vào môi trường (crossplatform). Đây là hai yếu tố rất quan trọng, thứ nhất khi nói công nghệ phía máy chủ tức là nói đến mọi thứ trong PHP đều xảy ra trên máy chủ, thứ hai, chính vì tính chất không phụ thuộc môi trường cho phép PHP chạy trên hầu hết trên các hệ điều hành như Windows, Unix và nhiều biến thể của nó... Đặc biệt các mã kịch bản PHP viết trên máy chủ này sẽ làm việc bình thường trên máy chủ khác mà không cần phải chỉnh sửa hoặc chỉnh sửa rất ít. Khi một trang web muốn được dùng ngôn ngữ PHP thì phải đáp ứng được tất cả các quá trình xử lý thông tin trong trang web đó, sau đó đưa ra kết quả ngôn ngữ HTML. Khác với ngôn ngữ lập trình, PHP được thiết kế để chỉ thực hiện điều gì đó sau khi một sự kiện xảy ra (ví dụ, khi người dùng gửi một biểu mẫu hoặc chuyển tới một URL).

### 2.1.2. Lý do nên dùng PHP

Để thiết kế web động có rất nhiều ngôn ngữ lập trình khác nhau để lựa chọn, mặc dù cấu hình và tính năng khác nhau nhưng chúng vẫn đưa ra những kết quả giống nhau. Chúng ta có thể lựa chọn cho mình một ngôn ngữ: ASP, PHP, Java, Perl,... và một số loại khác nữa. Vậy tại sao chúng ta lại nên chọn PHP? Rất đơn giản, có những lí do sau mà khi lập trình web chúng ta không nên bỏ qua sự lựa chọn tuyệt vời này:

- PHP được sử dụng làm web động vì nó nhanh, dễ dàng, tốt hơn so với các giải pháp khác.

- PHP có khả năng thực hiện và tích hợp chặt chẽ với hầu hết các cơ sở dữ liệu có sẵn, tính linh động, bền vững và khả năng phát triển không giới hạn.

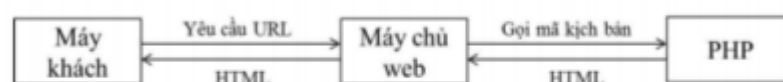
- Đặc biệt PHP là mã nguồn mở do đó tất cả các đặc tính trên đều miễn phí, và chính vì mã nguồn mở sẵn có nên cộng đồng các nhà phát triển web luôn có ý thức cải tiến nó, nâng cao để khắc phục các lỗi trong các chương trình này

- PHP vừa dễ với người mới sử dụng vừa có thể đáp ứng mọi yêu cầu của các lập trình viên chuyên nghiệp, mọi ý tưởng của các PHP có thể đáp ứng một cách xuất sắc.

- ASP vốn được xem là ngôn ngữ kịch bản phổ biến nhất, vậy mà bây giờ PHP đã bắt kịp ASP, bằng chứng là nó đã có mặt trên 12 triệu website.

### 2.1.3. Hoạt động của PHP

Vì PHP là ngôn ngữ của máy chủ nên mã lệnh của PHP sẽ tập trung trên máy chủ để phục vụ các trang web theo yêu cầu của người dùng thông qua trình duyệt. Sơ đồ hoạt động của PHP:



Hình 2.3. Sơ đồ hoạt động của PHP

Khi người dùng truy cập website viết bằng PHP, máy chủ đọc mã lệnh PHP và xử lý chúng theo các hướng dẫn được mã hóa. Mã lệnh PHP yêu cầu máy chủ gửi một dữ liệu thích hợp (mã lệnh HTML) đến trình duyệt web. Trình duyệt xem nó như là một trang HTML tiêu chuẩn. Như ta đã nói, PHP cũng chính là một trang HTML nhưng có nhúng mã PHP và có phần mở rộng là HTML. Phần mở của PHP được đặt trong thẻ mở `<?php` và thẻ đóng `?>`. Khi trình duyệt truy cập vào một trang PHP, Server sẽ đọc nội dung file PHP lên và lọc ra các đoạn mã PHP và thực thi các đoạn mã đó, lấy kết quả nhận được của đoạn mã PHP thay thế vào chỗ ban đầu của chúng trong file PHP, cuối cùng Server trả về kết quả cuối cùng là một trang nội dung HTML về cho trình duyệt.

#### 2.1.4. Tổng quan về PHP

- Cấu trúc cơ bản: PHP cũng có thẻ bắt đầu và kết thúc giống với ngôn ngữ HTML. Chỉ khác, đối với PHP chúng ta có nhiều cách để thể hiện.

- Cú pháp chính `<?php mã lệnh php ?>`
- Trong PHP để kết thúc 1 dòng lệnh chúng ta sử dụng dấu ";" . Để chú thích một đoạn dữ liệu nào đó trong PHP ta sử dụng dấu "/" cho từng dòng hoặc dùng cặp thẻ `/*.....*/` cho từng cụm mã lệnh.
- Ví dụ:`<?php echo "Hello world!";?>`

- Xuất giá trị ra trình duyệt chúng ta có những dòng cú pháp sau:

- `echo "thông tin";`
- `printf "thông tin";`

Thông tin bao gồm: biến, chuỗi, hoặc lệnh HTML ....

```
<?php
    echo "Hello World";
    print "<p>Who are you?</p>";
?>
```

Nếu giữa hai chuỗi muốn liên kết với nhau ta sử dụng dấu "."

```
<?php
    echo "Hello World"."Who are you?";
?>
```



- Biến: được xem là vùng nhớ dữ liệu tạm thời. Và giá trị có thể thay đổi được. Biến được bắt đầu bằng ký hiệu "\$" và theo sau chúng là một từ, một cụm từ nhưng phải viết liền hoặc có gạch dưới.

- Một biến được xem là hợp lệ khi nó thỏa các yếu tố:

- Tên của biến phải bắt đầu bằng dấu gạch dưới và theo sau là các ký tự, số hay dấu gạch dưới.

- Tên của biến không được phép trùng với các từ khóa của PHP.

- Trong PHP để sử dụng một biến chúng ta thường phải khai báo trước, tuy nhiên đối với các lập trình viên khi sử dụng họ thường xử lý cùng một lúc các công việc, nghĩa là vừa khai báo vừa gán dữ liệu cho biến. Bản thân biến cũng có thể gán cho các kiểu dữ liệu khác và tùy theo ý định của người lập trình mong muốn trên chúng.

```
<?php
    $a = 100; // biến a có giá trị là 100
    $a = "PHP is easy"; //biến a có giá trị là "PHP
is easy"
    bien = 120; // lỗi vì bắt đầu một biến phải có
dấu "$"
    $123a = "PHP"; //lỗi vì bắt đầu một biến không
được là số
?>
```

- Hằng: nếu biến là cái có thể thay đổi được thì ngược lại hằng là cái chúng ta không thể thay đổi được. Hằng trong PHP được định nghĩa bởi hàm define theo cú pháp: define (string tên\_hằng, giá\_trị\_hằng).

- Cũng giống với biến, hằng được xem là hợp lệ thì chúng phải đáp ứng một số yếu tố:

- Hằng không có dấu "\$" ở trước tên.

- Hằng có thể truy cập bất cứ vị trí nào trong mã lệnh.

- Hằng chỉ được phép gán giá trị duy nhất 1 lần.

- Hằng thường viết bằng chữ in để phân biệt với biến.

```
<?php
define("C", "company");
define("YELLOW", "#ffff00");
```

```
echo "Gia tri cua C la ".C;
?>
```

- Chuỗi: là một nhóm các ký tự, số, khoảng trắng, dấu ngắt được đặt trong các dấu nháy, ví dụ: 'Hello'.

- Để tạo một biến chuỗi, chúng ta phải gán giá trị chuỗi cho một biến hợp lệ,

ví dụ: \$first\_name= "Nguyen";

- Để liên kết một chuỗi và một biến chúng ta thường sử dụng dấu ".".

```
<?php
    $test = "5ace.vn";
    echo "Welcome to ".$test;
    echo "<br><font color=red>Welcome to
    ".$test."</font>";
?>
```

#### 2.1.5. Các phương thức được sử dụng trong lập trình PHP

Có 2 phương thức được sử dụng trong lập trình PHP là GET và POST.

- Phương thức GET: cũng được dùng để lấy dữ liệu từ form nhập liệu. Tuy nhiên nhiệm vụ chính của nó vẫn là lấy nội dung trang dữ liệu từ web server. Ví dụ: với url sau: shownews.php?id=50, ta dùng hàm \$\_GET['id'] sẽ được giá trị là 50.

- Phương thức POST: phương thức này được sử dụng để lấy dữ liệu từ form nhập liệu và chuyển chúng lên trình chủ webserver.

```
<?php
    echo "Welcome. ".$_GET['hoten']."!";
?>
<html>
    <form action="test" method="POST" action="#">
        Ho ten:
        <input type="text" name="hoten">;
        <input type="submit" name="ok"
        value="ok">;
    </form>
</html>
```

### 2.1.6. Cookie và Session trong PHP

Cookie và Session là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống.

- Cookie: là 1 đoạn dữ liệu được ghi vào đĩa cứng hoặc bộ nhớ của máy người sử dụng. Nó được trình duyệt gửi ngược lên lại server mỗi khi browser tải 1 trang web từ server. Những thông tin được lưu trữ trong cookie hoàn toàn phụ thuộc vào website trên server. Mỗi website có thể lưu trữ những thông tin khác nhau trong cookie, ví dụ thời điểm lần cuối ta ghé thăm website, đánh dấu ta đã login hay chưa,... Cookie được tạo ra bởi website và gửi tới browser, do vậy hai website khác nhau (cho dù cùng host trên 1 server) sẽ có hai cookie khác nhau gửi tới browser. Ngoài ra, mỗi browser quản lý và lưu trữ cookie theo cách riêng của mình, cho nên hai browser cùng truy cập vào một website sẽ nhận được hai cookie khác nhau.

1 Để thiết lập cookie ta sử dụng cú pháp:

- Setcookie ("tên cookie", "giá trị", thời gian sống).
- Tên cookie là tên mà chúng ta đặt cho phiên làm việc.
- Giá trị là thông số của tên cookie.
- Ví dụ: `setcookie("name", "admin", time()+3600);`

2 Để sử dụng lại cookie vừa thiết lập, chúng ta sử dụng cú pháp:

- Cú pháp: `$_COOKIE["tên cookies"]`.
- Tên cookie là tên mà chúng ta thiết lập phía trên.

3 Để hủy 1 cookie đã được tạo ta có thể dùng 1 trong 2 cách sau:

- Cú pháp: `setcookie("Tên cookie")`.
- Gọi hàm setcookie với chỉ duy nhất tên cookie mà thôi
- Dùng thời gian hết hạn cookie là thời điểm trong quá khứ.
- Ví dụ: `setcookie("name", "admin", time()-3600);`

- Session: được hiểu là khoảng thời gian người sử dụng giao tiếp với một ứng dụng. Một session được bắt đầu khi người sử dụng truy cập vào ứng

dụng lần đầu tiên, và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi session sẽ có được cấp một định danh (ID) khác nhau.

1 Để thiết lập 1 session ta sử dụng cú pháp: `session_start()`. Đoạn code này phải được nằm trên các kịch bản HTML hoặc những lệnh `echo`, `printf`.

2 Để thiết lập một giá trị session, ngoài việc cho phép bắt đầu thực thi session. Chúng ta còn phải đăng ký một giá trị session để tiện cho việc gán giá trị cho session đó.

3 Ta có cú pháp: `session_register("Name")`.

4 Để sử dụng giá trị của session ta sử dụng mã lệnh sau: `$_SESSION["name"]` với "name" là tên mà chúng ta sử dụng hàm `session_register("name")` để khai báo.

#### 2.1.7. Hàm

Để giảm thời gian lặp lại 1 thao tác code nhiều lần, PHP hỗ trợ người lập trình việc tự định nghĩa cho mình những hàm có khả năng lặp lại nhiều lần trong website. Việc này cũng giúp cho người lập trình kiểm soát mã nguồn một cách mạch lạc, đồng thời có thể tùy biến ở mọi trang mà không cần phải khởi tạo hay viết lại mã lệnh như HTML thuần.

##### a) Hàm tự định nghĩa

Cú pháp:

```
function function_name($gt1,$gt2)
{
//Lệnh thực thi
}
```

##### b) Hàm tự định nghĩa với giá trị trả về

Cú pháp:

```
function function_name(Có hoặc không có đối số)
{
// Lệnh thực thi
return giatri;
}
```

c) Gọi lại hàm

PHP cung cấp nhiều hàm cho phép triệu gọi lại file. Như hàm `include("URL đến file")`, `require("URL Đến file")`. Ngoài hai cú pháp trên còn có `include_once()`, `require_once()`. Hai hàm này cũng có trách nhiệm gọi lại hàm. Nhưng chúng sẽ chỉ gọi lại duy nhất một lần mà thôi.

## 2.2. MySQL

### 2.2.1 Giới thiệu cơ sở dữ liệu:

MySQL là ứng dụng cơ sở dữ liệu mã nguồn mở phổ biến nhất hiện nay (theo [www.mysql.com](http://www.mysql.com)) và được sử dụng phối hợp với PHP. Trước khi làm việc với MySQL cần xác định các nhu cầu cho ứng dụng.

MySQL là cơ sở dữ có trình giao diện trên Windows hay Linux, cho phép người sử dụng có thể thao tác các hành động liên quan đến nó. Việc tìm hiểu từng công nghệ trước khi bắt tay vào việc viết mã kịch bản PHP, việc tích hợp hai công nghệ PHP và MySQL là một công việc cần thiết và rất quan trọng.

### 2.2.2 Mục đích sử dụng cơ sở dữ liệu:

Mục đích sử dụng cơ sở dữ liệu bao gồm các chức năng như: lưu trữ (storage), truy cập (accessibility), tổ chức (organization) và xử lí (manipulation).

- Lưu trữ: Lưu trữ trên đĩa và có thể chuyển đổi dữ liệu từ cơ sở dữ liệu này sang cơ sở dữ liệu khác, nếu sử dụng cho quy mô nhỏ, có thể chọn cơ sở dữ liệu nhỏ như: Microsoft Exel, Microsoft Access, MySQL, Microsoft Visual FoxPro,... Nếu ứng dụng có quy mô lớn, có thể chọn cơ sở dữ liệu có quy mô lớn như: Oracle, SQL Server,...

- Truy cập: Truy cập dữ liệu phụ thuộc vào mục đích và yêu cầu của người sử dụng, ở mức độ mang tính cục bộ, truy cập cơ sở dữ liệu ngay trong cơ sở dữ liệu với nhau, nhằm trao đổi hay xử lí dữ liệu ngay bên trong chính nó, nhưng do mục đích và yêu cầu người dùng vượt ra ngoài cơ sở dữ liệu, nên cần có các phương thức truy cập dữ liệu giữa các cơ sở dữ liệu với nhau như: Microsoft Access với SQL Server, hay SQL Server và cơ sở dữ liệu Oracle....

- Tổ chức: Tổ chức cơ sở dữ liệu phụ thuộc vào mô hình cơ sở dữ liệu, phân tích và thiết kế cơ sở dữ liệu tức là tổ chức cơ sở dữ liệu phụ thuộc vào đặc điểm riêng của từng ứng dụng. Tuy nhiên khi tổ chức cơ sở dữ liệu cần phải tuân theo một số tiêu chuẩn của hệ thống cơ sở dữ liệu nhằm tăng tính tối ưu khi truy cập và xử lí.

- Xử lí: Tùy vào nhu cầu tính toán và truy vấn cơ sở dữ liệu với các mục đích khác nhau, cần phải sử dụng các phát biểu truy vấn cùng các phép toán, phát biểu của cơ sở dữ liệu để xuất ra kết quả như yêu cầu. Để thao tác hay xử lí dữ liệu bên trong chính cơ sở dữ liệu ta sử dụng các ngôn ngữ lập trình như: PHP, C++, Java, Visual Basic,...

### 2.2.3 Các kiểu dữ liệu trong cơ sở dữ liệu MySQL

- Kiểu dữ liệu numeric: bao gồm số nguyên và kiểu số chấm động.

Loại	Range	Bytes
Tinyint	-127 -> 128	1
Smallint	-32768 -> 32767	2
Mediumint	-8388608 -> 8388607	3
Int	$-2^{31} -> 2^{31}-1$	4
Bigint	$-2^{63} -> 2^{63}-1$	

- Kiểu chấm động

Loại	Range	byte
Float	$\pm 1.175494351E-38$ $\pm 3.402823466E+38$	4
Double	$\pm 2.2250738585072014E-308$ $\pm 1.7676931348623157E+308$	8

- Kiểu dữ Date and Time cho phép nhập dữ liệu dưới dạng chuỗi ngày tháng hay dạng số.

Loại	Range	Diễn giải
Date	1000-01-01	Date trình bày dưới dạng yyyy-mm-dd.
Time	00:00:00 23:59:59	Time trình bày dưới dạng hh:mm:ss.
DateTime	1000-01-01 00:00:00 9999-12-31 23:59:59	DateTime trình bày dưới dạng yyyy-mm-dd hh:mm:ss.
TimeStamp[(M)]	1970-01-01 00:00:00	TimeStamp trình bày dưới dạng yyyy-mm-dd hh:mm:ss

- Kiểu dữ liệu String: chia làm 3 loại: char (chiều dài cố định) và varchar (chiều dài biến thiên); Text (cho phép lưu chuỗi lớn) và Blob (cho phép lưu đối tượng nhị phân); Enum và Set.

Loại	Range	Diễn giải
Char	1-255	Chiều dài của chuỗi lớn nhất 255 ký tự

Varchar	1-255	Chiều dài của chuỗi lớn nhất 255 ký tự
Tinyblob	$2^8-1$	Khai báo cho Field chứa kiểu đối tượng nhị phân cỡ 255 ký tự
Tinytext	$2^8-1$	Khai báo cho Field chứa kiểu chuỗi cỡ 255 ký tự
Blob	$2^{16}-1$	Khai báo cho Field chứa kiểu blob cỡ 65535 ký tự
Text	$2^{16}-1$	Khai báo cho Field chứa kiểu chuỗi dạng văn bản cỡ 65535 ký tự
Mediumblob	$2^{24}-1$	Khai báo cho Field chứa kiểu blob khoảng 16.777.215 ký tự

Longblob	$2^{32}-1$	Khai báo cho Field chứa kiểu blob khoảng 4.294.967.295 ký tự
Longtext	$2^{32}-1$	Khai báo cho Field chứa kiểu chuỗi dạng văn bản cỡ 4.294.967.295 ký tự

#### 2.2.4 Các thao tác cập nhật dữ liệu

- SELECT (truy vấn mẫu tin): SELECT dùng để truy vấn từ một hay nhiều bảng khác nhau, kết quả trả về là một tập mẫu tin thỏa mãn các điều kiện cho trước nếu có, cú pháp phát biểu SQL dạng SELECT như sau:

```
SELECT < danh sách các cột>
[FROM < danh sách các bảng>]
[WHERE < các điều kiện ràng buộc>]
[GROUP BY < tên cột/ biểu thức trong SELECT>]
[HAVING < điều kiện bắt buộc của GROUP BY>]
[ORDER BY < danh sách các cột> ]
[LIMIT FromNumber/ ToNumber]
```

- INSERT (thêm mẫu tin):

Cú pháp: INSERT INTO Tên\_bảng VALUE (bộ giá trị)

- UPDATE (cập nhật dữ liệu):

Cú pháp: UPDATE TABLE tên\_bảng SET tên\_cột = Biểu\_thức [WHERE điều\_kiện]

- DELETE (xóa mẫu tin):

Cú pháp: DELETE FROM tên\_bảng [WHERE điều\_kiện]

#### 2.2.5 Các hàm thông dụng trong MySQL

- Các hàm trong GROUP BY

1 Hàm AVG: hàm trả về giá trị bình quân của cột hay trường trong câu truy vấn.



2 Hàm MIN: hàm trả về giá trị nhỏ nhất của cột hay trường trong câu truy vấn.

3 Hàm MAX: hàm trả về giá trị lớn nhất của cột hay trường trong câu truy vấn.

4 Hàm COUNT: hàm trả về số lượng mẫu tin trong câu truy vấn.

5 Hàm SUM: hàm trả về tổng các giá trị của cột, trường trong câu truy vấn

- Các hàm xử lý chuỗi

1 Hàm ASCII: hàm trả về giá trị mã ASCII của ký tự bên trái của chuỗi.

2 Hàm CHAR: hàm chuyển đổi kiểu mã ASCII từ số nguyên sang dạng chuỗi.

3 Hàm UPPER: hàm chuyển đổi chuỗi sang kiểu chữ hoa.

4 Hàm LOWER: hàm chuyển đổi chuỗi sang kiểu chữ thường.

5 Hàm LEN: hàm trả về chiều dài của chuỗi.

6 Hàm LTRIM: hàm loại bỏ khoảng trắng bên trái chuỗi.

7 Hàm RTRIM: hàm loại bỏ khoảng trắng bên phải chuỗi.

8 Hàm LEFT(STR, N): hàm trả về chuỗi bên trái tính từ đầu cho đến vị trí n.

9 Hàm RIGHT(STR, N): hàm trả về chuỗi bên phải tính từ đầu cho đến vị trí n.

10 Hàm INSTR: hàm trả về chuỗi vị trí bắt đầu của chuỗi con trong chuỗi xét.

- Các hàm xử lý về thời gian

1 Hàm CURDATE(): hàm trả về ngày, tháng, năm hiện hành của hệ thống.

2 Hàm CURTIME(): hàm trả về giờ, phút, giây hiện hành của hệ thống.

## CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### I. Phát biểu bài toán

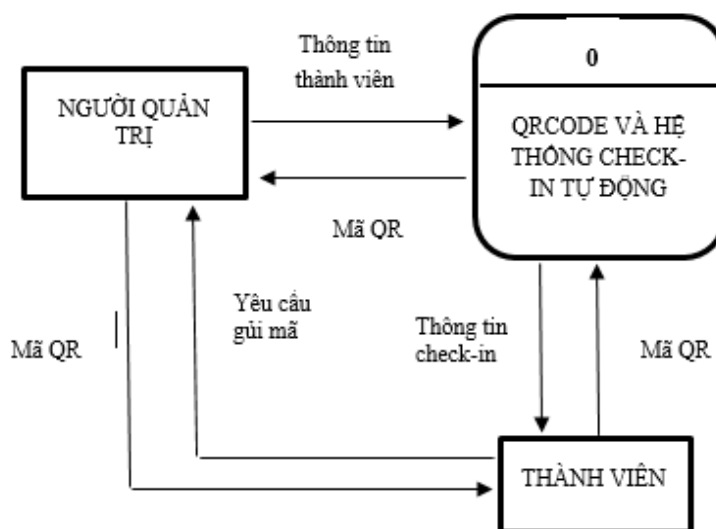
Hiện nay mã QRcode được sử dụng rộng rãi là do sự tiện ích mà nó mang lại như:

Dễ dàng tra cứu thông tin của 1 sản phẩm bất kỳ, gửi mail, sms, thanh toán, chuyên khoản ngân hàng... Thậm chí nó có thể lưu đầy đủ thông tin cá nhân của người dùng, chính vì lẽ đó chương trình tạo mã qrcode đã được viết ra nhằm mục đích giúp việc check-in tự động dễ dàng hơn. Người quản trị sẽ nhập đầy đủ thông tin người dùng vào hệ thống bao gồm: họ tên, email, sdt. Sau khi nhập xong hệ thống sẽ tự động lưu thông tin người dùng vào danh sách, đồng thời tạo 1 mã qrcode cho người dùng đó bằng chính địa chỉ email của họ.

- Người quản trị sẽ cung cấp mã qrcode đó cho người dùng. Sau khi có mã qrcode trong tay thì người dùng có thể check-in thông tin của mình tại hệ thống, sau khi check-in thành công, thông tin người dùng sẽ được lưu vào hệ thống.
- Người quản trị hệ có thể kiểm tra ngày giờ check-in của từng thành viên trong danh sách check-in.

### II. Biểu đồ nghiệp vụ

#### 1. Biểu đồ ngữ cảnh hệ thống



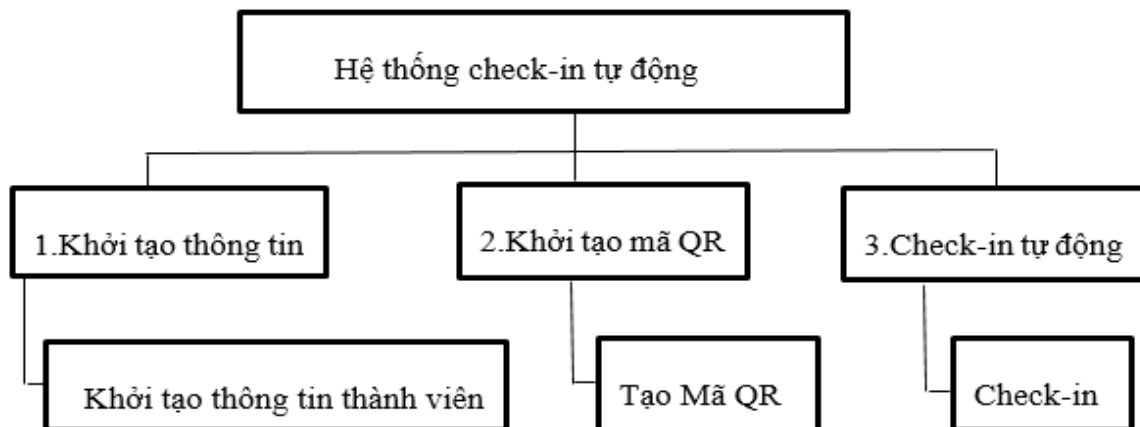
Biểu đồ ngữ cảnh hệ thống

## a) Mô tả hoạt động

Trước khi thành viên có thể check-in tự động ở hệ thống thì quá trình sẽ được thực hiện như sau:

- Để tạo được mã qrcode cho thành viên thì trước tiên người quản trị phải điền đầy đủ thông tin thành viên vào danh sách thành viên
- Sau khi đã thêm thành công hệ thống sẽ lưu thông tin thành viên và đồng thời tạo một mã qrcode theo địa chỉ email của họ
- Mã sau khi được tạo sẽ được người quản trị cung cấp cho thành viên. Khi có mã qrcode thành viên có thể tiến hành check-in tại hệ thống. Hệ thống sau khi đã nhận được mã qrcode thành công sẽ lưu vào danh sách check-in.

## 2. Biểu đồ phân rã chức năng



Mô tả chi tiết chức năng:

*Chức năng 1 – Khởi tạo thông tin:* Để người dùng có thể check-in tại hệ thống bắt buộc thành viên phải có trong danh sách thành viên. Khi đó người quản trị sẽ nhập đầy đủ thông tin thành viên vào hệ thống bao gồm : họ tên, email, sdt.

*Chức năng 2 – Khởi tạo mã QR:* Sau khi nhập đầy đủ thông tin thì hệ thống sẽ tự tạo một mã qrcode của thành viên đó. Mã sau khi được tạo sẽ được người quản trị đưa đến cho mỗi thành viên.

Chức năng 3 – *Check-in tự động* : Thành viên có mã đến hệ thống tiến hành quét mã, sau khi đã nhận diện được mã của thành viên thì hệ thống sẽ tự động lưu lại thông tin thành viên. Như vậy thành viên đã check-in thành công.

### 3. Ma trận thực thể chức năng

#### 3.1 Danh sách hồ sơ dữ liệu

- a. Thông tin thành viên
- b. Danh sách mã QR
- c. Danh sách check-in

Các chức năng:

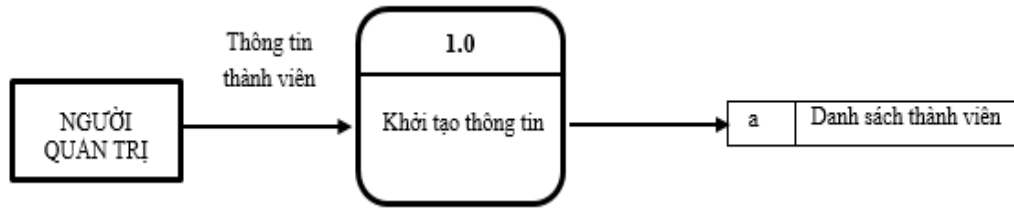
- 1 Khởi tạo thông tin thành viên
- 2 Khởi tạo mã QR
- 3 Check-in tự động

#### 3.2 Ma trận thực thể chức năng

Các thực thể dữ liệu			
a. Danh sách thành viên			
b. Danh sách mã QR			
c. Danh sách check-in			
Các chức năng nghiệp vụ	a	b	c
1. Khởi tạo thông tin thành viên	C		
2. Khởi tạo mã QR	R	C	
3. Check-in tự động	R	R	C

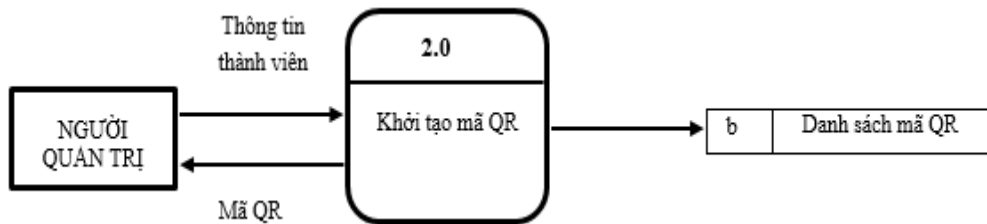
**4. Biểu đồ luồng dữ liệu**

**3.1 Biểu đồ luồng dữ liệu chức năng thêm thành viên**



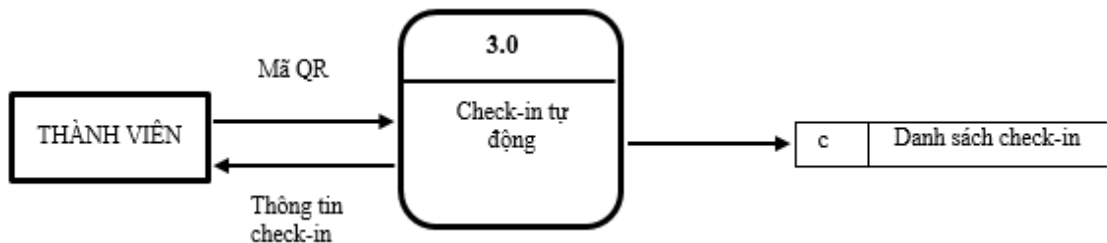
Sơ đồ trên mô tả chức năng thêm thành viên trong hệ thống. Người quản trị sẽ điền đầy đủ thông tin gồm Họ tên, Sdt, Email vào danh sách thành viên. Sau khi đã thêm thành viên thành công hệ thống sẽ tự động lưu vào danh sách.

**3.2 Biểu đồ luồng dữ liệu chức năng khởi tạo mã QR**



Sơ đồ trên mô tả chức năng khởi tạo mã qrcode. Khi người quản trị đã nhập đầy đủ thông tin thành viên và xác nhận tạo thì hệ thống sẽ tạo cho thành viên đó một mã qrcode từ email của thành viên đó.

**3.3 Biểu đồ luồng dữ liệu chức năng check-in tự động**



Sơ đồ trên mô tả chức năng check-in của hệ thống. Khi đã có mã qrcode thành viên đến check-in tại hệ thống. Hệ thống sẽ tự động lưu thông tin và ngày giờ check-in của thành viên đó vào danh sách check-in.

### 5. Mô hình thực thể E-R

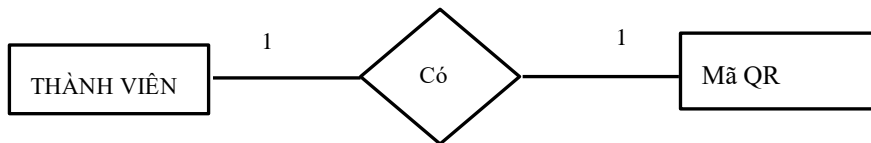
#### 5.1 Các kiểu thực thể

Danh sách thành viên có thuộc tính : Mã thành viên, Tên thành viên, số điện thoại, Email.

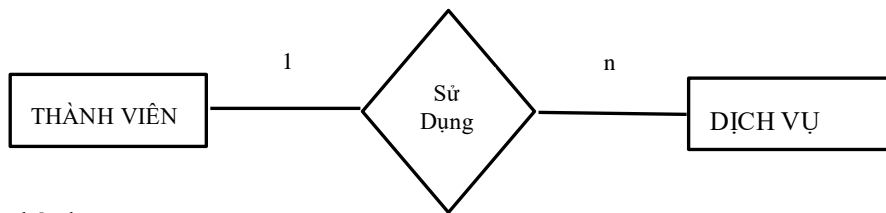
Danh sách check-in có thuộc tính : Mã dịch vụ, Email, Ngày giờ.

#### 5.2 Xác định các mối quan hệ

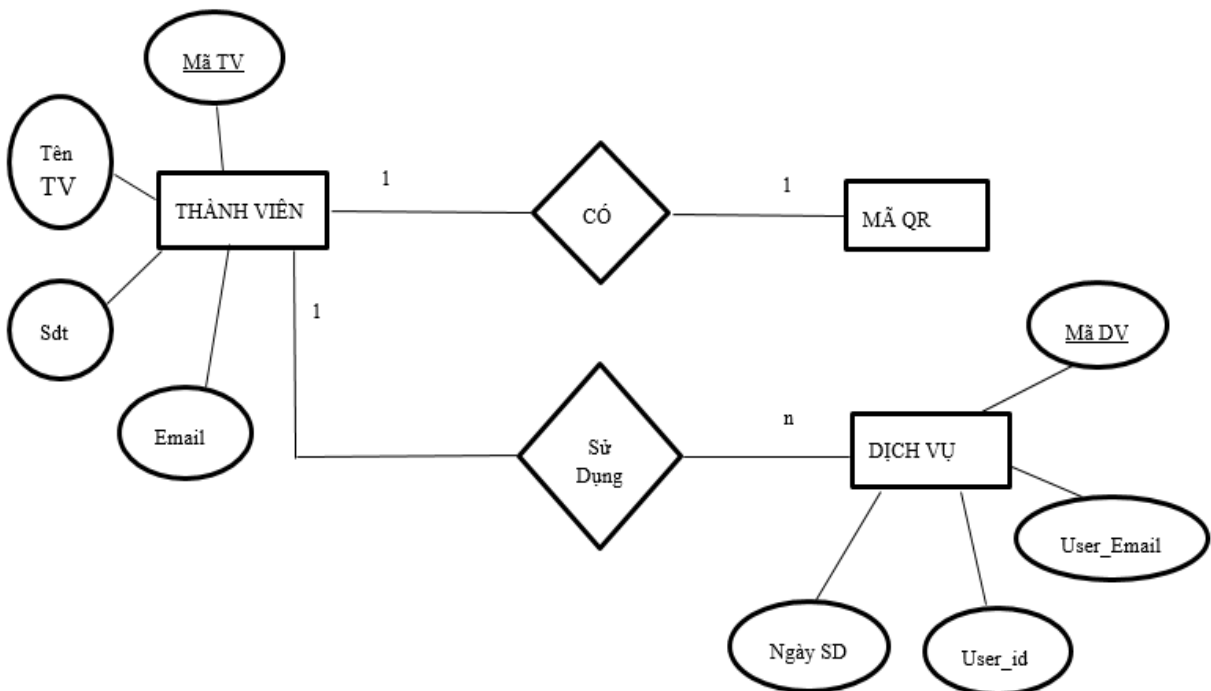
- THÀNH VIÊN có MÃ QR
- Mã QR được hệ thống tự động tạo từ thông tin thành viên



- THÀNH VIÊN sử dụng DỊCH VỤ



#### 5.3 Mô hình ER



## 6. Thiết kế các bảng dữ liệu

### 6.1 Bảng User

Stt	Tên trường	Kiểu dữ liệu	Ghi chú
1	Id(mã thành viên)	int	Khóa chính
2	Hoten	Varchar	
3	Sdt	Char	
4	Email	Varchar	
5	MaQR	Varchar	

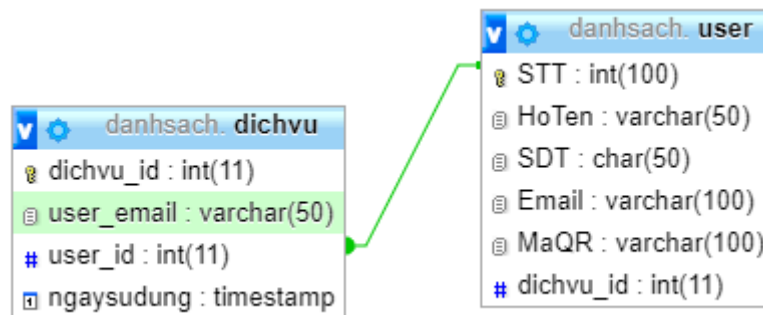
Trong bảng user có các trường : id, hoten, sdt, email, maqr, trong đó có id(mã thành viên) là khóa chính

### 6.2 Bảng DichVu

Stt	Tên trường	Kiểu dữ liệu	Ghi chú
1	Dichvu_id	int	Khóa chính
2	User_email	Varchar	
3	User_id	int	Khóa ngoại
4	ngaysudung	timestamp	

Trong bảng dichVu có các trường : dichvu\_id, user\_email, user\_id, ngaysudung trong đó có dichvu\_id là khóa chính, user\_id là khóa ngoại

## 7 Mô hình quan hệ



Mô hình quan hệ của hệ thống

## CHƯƠNG 3. THUẬT TOÁN, ỨNG DỤNG VÀ THỰC NGHIỆM

### 1. Thuật toán sinh mã qrcode

#### 1.1 Các bước mã hóa tạo một chuỗi nhị phân

- Bước 1 Mã hóa Bộ chỉ chế độ (Mode Indicator):

Một bộ chỉ chế độ là một chuỗi 4 bit thể hiện chế độ dữ liệu bạn đang sử dụng: kiểu số, kiểu chuỗi, kiểu nhị phân hoặc Nhật ngữ.

Với chuỗi HELLO WORLD, là kiểu chuỗi, vậy nên Bộ chỉ chế độ sẽ là 0010.

Chuỗi nhị phân: 0010

- Bước 2 Mã hóa độ dài của dữ liệu:

Trong bước này, chúng ta xác định có bao nhiêu ký tự trong thông điệp, và chuyển độ dài đó thành một số nhị phân.

VD: Với chuỗi HELLO WORLD, có 11 ký tự kể cả khoảng trắng. Ta chuyển 11 sang nhị phân, được 1011.

- Bước 3 Mã hóa dữ liệu :

Để mã hóa chuỗi ký tự, ta chia chuỗi thành các cặp ký tự: HE,LL,O ,WO,RL,D. Với mỗi cặp ký tự, ta lấy giá trị mã ASCII của ký tự đầu nhân với 45, sau đó cộng với giá trị mã ASCII của ký tự sau. Sau đó chuyển kết quả thu được thành một chuỗi nhị phân 11 bit.

- Bước 4 Hoàn thành các bit:

Với chuỗi bit thu được ở trên, ta phải chắc chắn rằng nó có đủ độ dài. Điều này phụ thuộc vào việc có bao nhiêu bit chúng ta cần để tạo phiên bản và sửa lỗi.

Chuỗi của chúng ta có 59 bit, ta thêm 4 số 0 vào cuối chuỗi. (Còn nếu chuỗi có 102 bit thì ta thêm 2 số 0).

Chuỗi của ta bây giờ là:

0010 000001011 01100001011 01111000110 10001011100  
10110111000 10011010100 001101 **0000**



- Bước 5 Giới hạn chuỗi thành các chuỗi con 8 bit, sau đó thêm các số 0 nếu cần.

Bước này ta chia chuỗi thành các nhóm 8 bit.

Ở chuỗi con cuối cùng, nếu chưa đủ 8 bit thì ta thêm vào sau đó các số 0 cho tới khi đủ. Chuỗi của chúng ta thêm 2 số 0 vào sau chuỗi con cuối cùng.

```
00100000 01011011 00001011 01111000 11010001 01110010
11011100 01001101 01000011 01000000
```

- Bước 6 Thêm các từ vào cuối nếu chuỗi quá ngắn

Chuỗi của chúng ta chỉ có 10 chuỗi con, cần phải thêm vào 3 chuỗi con nữa. Bắt đầu thêm 11101100, sau đó thêm 00010001, rồi tới 11101100. Nếu cần nhiều hơn thì cứ thêm lần lượt như vậy cho tới khi đủ số chuỗi con.

Chuỗi của chúng ta bây giờ:

```
00100000 01011011 00001011 01111000 11010001 01110010
11011100 01001101 01000011 01000000 11101100 00010001
11101100
```

## 1.2 Ứng dụng thực nghiệm

Đoạn code mô tả thư viện cần sử dụng trong chương trình tạo mã QRcode.

```
1 <?php
2 include('qrlib.php');
3 $PNG_TEMP_DIR = dirname(__FILE__).DIRECTORY_SEPARATOR.'temp'.DIRECTORY_SEPARATOR;
4 $PNG_WEB_DIR = 'temp/';
5 ?>
```

Ban đầu sẽ phải kết nối đến thư viện “qrlib.php”, trong thư viện này chứa các tập tin thư mục gốc và tiếp đến \$PNG\_WEB\_DIR = ‘temp/’; đó là tên thư mục chứa file ảnh qrcode

Đoạn code thực hiện kiểm tra thông tin và tạo mã QR cho thành viên.

```

66 <?php
67 require("ketnoi.php");
68 $sql = "select * from user";
69 $query = mysql_query($sql);
70 $num = mysql_num_rows($query);
71 $errorCorrectionLevel = 'L';
72 $matrixPointSize = 4;
73 if($num > 0){
74     while($row = mysql_fetch_array($query)){
75         //kiem tra da tao chua, neu chua thi tao o day
76         //QRcode la email
77         $filename = $PNG_TEMP_DIR.$row['STT'].'.png';
78         $codeContents = $row['Email'];
79         if(!file_exists($filename)){
80             QrCode::png($codeContents, $filename,$errorCorrectionLevel, $matrixPointSize, 2);
81         }
82     }
83 }
84 >>
85 <tr>
86
87 <td><?php echo $row['STT'] ?></td>
88 <td><?php echo $row['HoTen'] ?></td>
89 <td><?php echo $row['Email'] ?></td>
90 <td><?php echo $row['SDT'] ?></td>
91 <td></td>
92 <td><a href="sua.php?id=<?php echo $row['STT']?>" title=""><i class="fa fa-pencil-square-o"></i></a></td>
93 <td><a href="xoa.php?id=<?php echo $row['STT']?>" title=""><i class="fa fa-trash"></i></a></td>
94 </tr>
95 <?php
96     }
97 }
98 >>
99

```

Sau khi kết nối tới csdl thành công thì chúng ta sử dụng thẻ `$errorCorrectionLevel = 'L'`; đây là mức độ sửa lỗi được mã QR sử dụng trong bộ sửa lỗi Reed-Solomon, có 4 mức độ L,M,Q,H tương đương với 7%,15%,25%,30% phục hồi dữ liệu.

`$matrixPointSize = 4`; là kích cỡ ảnh qrcode

Nếu có dữ liệu thông tin thành viên thì tên của ảnh sẽ được đặt bằng id của thành viên và được lưu vào file 'temp' với đuôi định dạng ảnh là .png

Biến `$codeContents` được gán bằng thông tin email của thành viên.

Khi đã có đầy đủ những giá trị trên thì tiếp đến :

`QrCode::png($codeContents, $filename,$errorCorrectionLevel, $matrixPointSize, 2)`; sẽ thực thi nhiệm vụ còn lại tạo ra một code hoàn chỉnh.

## 2. Môi trường thử nghiệm

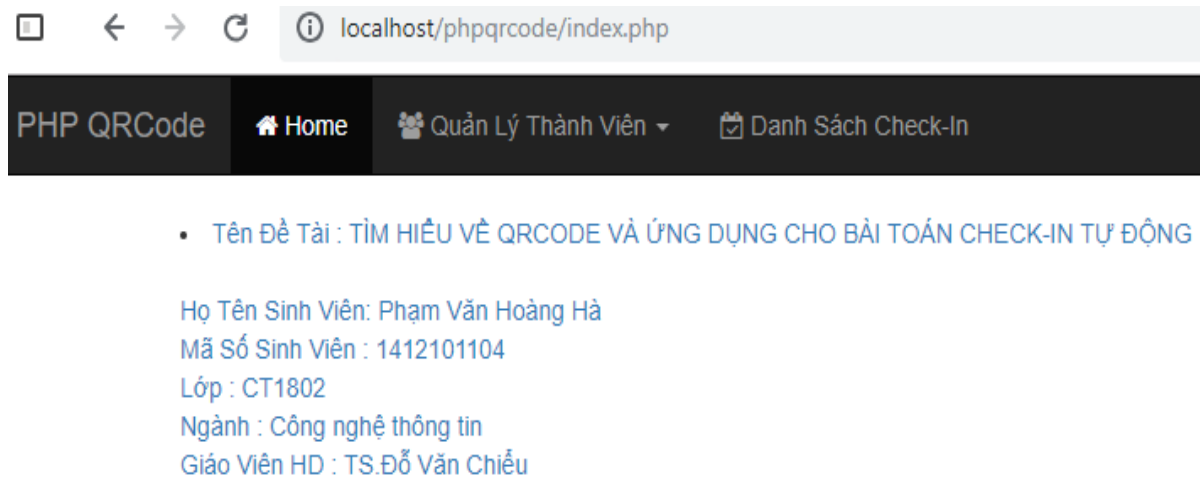
Hệ thống được thử nghiệm trên môi trường có các thông số như sau:

- Laptop Asus X550LD: Intel Core i5, RAM 8GB.
- Hệ điều hành: Window 10 64 bit.
- XAMPP: v3.2.1
- PHP Version 4.2.11

- MySQL to 5.6.21

### 3. Một Số Giao Diện Thử Nghiệm

#### 3.1 Giao diện trang chủ

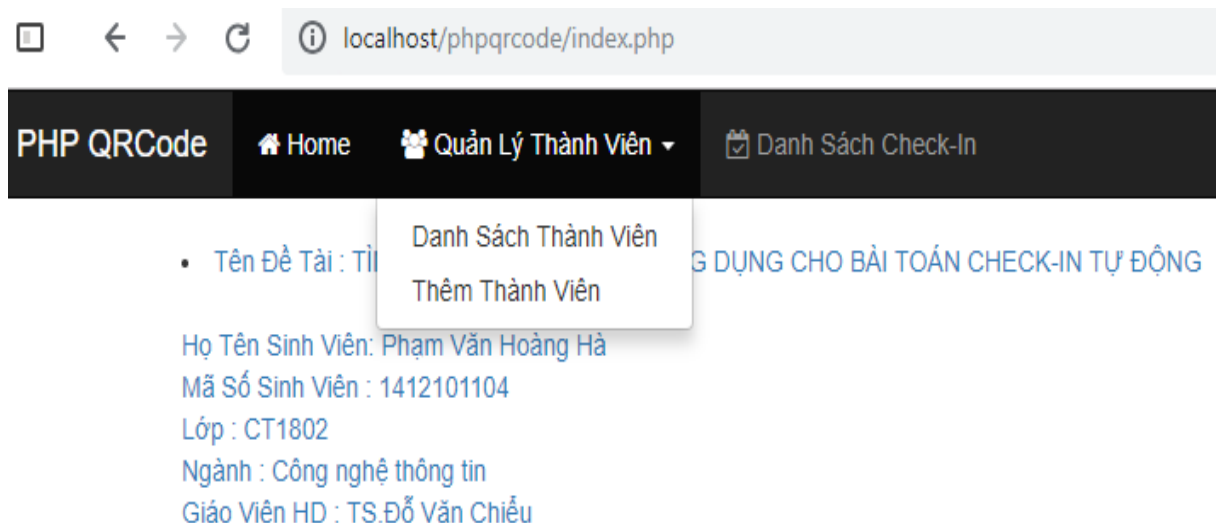


Ở trang index gồm có 3 phần chính đó là: trang chủ, quản lý thành viên, danh sách check-in. Trong phân quản lý thành viên được chia làm 2 phần nhỏ là danh sách và thêm mới thành viên.

Tiếp theo là đó phần tiêu đề và giới thiệu về website .

#### 3.2 Quản lý thành viên

- Trong phần Quản lý thành viên được chia ra làm 2 phần chính là : Danh sách thành viên và thêm thành viên



### 3.3 Giao diện của danh sách thành viên

STT	Họ Tên	Email	SDT	MãQR	Sửa	Xóa
2	Nguyen Van B	nguyenvanb@gmail.com	123456788			
7	nguyen van em	nguyenvanem@gmail.com	123654987			
8	nguyen thi thu van	nguyenthithuvan@gmail.com	132654789			

Trong giao diện của danh sách thành viên có STT ,Họ Tên, Email ,SDT, MãQR, và có thêm 2 chức năng sửa, xóa .Trong đó MãQR là hình ảnh .png được tạo lên bởi thông tin người dùng từ hệ thống.

### 3.4 Giao diện sửa thành viên

#### Sửa Thành Viên

**Họ Tên**

**Email**

**SDT**

Khi người quản trị muốn sửa thông tin thành viên thì sẽ kích vào biểu tượng sửa ở phần danh sách thì sẽ cho ra giao diện như trên.

Trong giao diện sửa có 3 phần tương ứng với 3 thông tin có thể sửa của thành viên đó là : họ tên, email, sdt

Khi đã điền đầy đủ thông tin cần sửa tiếp đó người quản trị kích vào nút button sửa sau đó hệ thống sẽ tự động chuyển về danh sách và lưu vào hệ thống.

### 3.4 Giao diện thêm thành viên

Thêm Thành Viên

Họ Tên

Email

SDT

Khi cần thêm thành viên người quản trị thực hiện thêm mới thì giao diện trên được hiện lên

Trong giao diện thêm mới thành viên có 3 thông tin bắt buộc đó là họ tên, email, sdt

Khi đã điền đầy đủ thông tin thành viên cần thêm mới người quản trị thực hiện kích vào nút button thêm

Sau đó hệ thống sẽ chuyển về danh sách thành viên và tự động lưu vào danh sách thành viên.

### 3.5 Giao diện chức năng check-in

Home Quản Lý Thành Viên Danh Sách Check-In

Xin mời quét mã

Email	Ngày giờ Check-In
doanvanhau@gmail.com	04/01/2019 22:25:48
doanvanhau@gmail.com	04/01/2019 22:24:23
doanvanhau@gmail.com	04/01/2019 22:12:46
doanvanhau@gmail.com	04/01/2019 22:12:14
doanvanhau@gmail.com	04/01/2019 22:01:52
nguyenvanb@gmail.com	21/12/2018 15:54:20
nguyenvanem@gmail.com	21/12/2018 15:53:37

Để thực hiện việc check-in thông tin thành viên trên hệ thống, thì thành viên sử dụng máy quét mã thực hiện quét mã QRcode của mình.

Bên dưới là email và ngày giờ mà hệ thống đã lưu

Khi có mã qrcode được quét qua ô “xin mời quét mã ” hệ thống sẽ tự động lưu thông tin trong mã qrode của thành viên đó, cụ thể là email và ngày giờ người đó thực hiện check-in tại hệ thống.

## KẾT LUẬN

Trong đồ án này em đã nghiên cứu, tìm hiểu về cách tạo ra mã QRcode bằng ngôn ngữ lập trình PHP và hệ quản trị cơ sở dữ liệu MySQL để ứng dụng vào bài toán check-in tự động.

Nội dung của đồ án em đã đạt được những vấn đề như sau:

Lý thuyết:

- Phát biểu và mô tả được nghiệp vụ bài toán.
- Biểu diễn các nghiệp vụ bài toán bằng cách các sơ đồ tiến trình nghiệp vụ.
- Phân tích thiết kế bài toán một cách đầy đủ.
- Thiết kế được cơ sở dữ liệu quan hệ để lưu trữ dữ liệu.

Chương trình:

- Sử dụng hệ quản trị cơ sở dữ liệu SQL.
- Sử dụng ngôn ngữ lập trình php tạo thành công mã QRcode bằng thông tin người dùng.
- Ứng dụng thành công vào hệ thống check-in tự động.
- Hệ thống được cài đặt thử nghiệm ban đầu trên localhost, rồi đưa lên hosting cho ra kết quả, đáp ứng được yêu cầu bài toán.

Hạn chế:

- Chương trình còn sơ sài chưa tối ưu hết các chức năng.
- Giao diện website còn đơn giản chưa bắt mắt.
- Chương trình cần được phát triển thêm về cả chức năng và giao diện.

Chương trình vẫn còn nhiều thiếu sót. Rất mong các thầy cô giáo và các bạn tận tình giúp đỡ để chương trình ngày càng hoàn thiện hơn.

## TÀI LIỆU THAM KHẢO

### **Tài liệu tiếng Việt**

- [1]. Hướng dẫn lập trình tạo QRcode với HTML5- JavaScript
- [2]. Lập trình web bằng PHP 5.3 và cơ sở dữ liệu MySQL 5.1 – Phạm Hữu Khang.
- [3]. Xây Dựng Ứng Dụng Web Bằng PHP Và MySQL (Ấn Bản Dành Cho Sinh Viên - Có CD Bài Tập Kèm Theo sách) – Phạm Hữu Khang.

### **Tham khảo Internet**

- [1]. <https://www.w3schools.com/>
- [2]. <http://phpqrcode.sourceforge.net/>