TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KÌ MÔN BẢO MẬT THÔNG TIN**

**QR CODE VÀ BẢO MẬT QR CODE**

*Người hướng dẫn*: **TS HUỲNH NGỌC TÚ**

*Người thực hiện*: **NGUYỄN ĐỨC TÚ - 51800322**

**NGUYỄN ĐÌNH LUÂN – 51800994**

**PHẠM TẤN KIỆT - 51800791**

Lớp **: 18050401**

Khoá  **: 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN CUỐI KÌ MÔN BẢO MẬT THÔNG TIN**

**QR CODE VÀ BẢO MẬT QR CODE**

*Người hướng dẫn*: **TS HUỲNH NGỌC TÚ**

*Người thực hiện*: **NGUYỄN ĐỨC TÚ - 51800322**

**NGUYỄN ĐÌNH LUÂN – 51800994**

**PHẠM TẤN KIỆT - 51800791**

Lớp **: 18050401**

Khoá  **: 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020**

LỜI CẢM ƠN

Để hoàn thành bài tập lớn này trước tiên chúng em xin gửi đến các quý thầy, cô giảng viên trường Đại học Tôn Đức Thắng lời cảm ơn chân thành và sâu sắc nhất. Đặc biệt, em xin gửi đến cô Huỳnh Ngọc Tú – người đã tận tình hướng dẫn, giúp đỡ chúng em hoàn thành bài tập lớn này lời cảm ơn sâu sắc nhất.

Với điều kiện thời gian cũng như kinh nghiệm còn chế của sinh viên, bài báo cáo này không thế tránh khỏi được những thiếu sót. Chúng em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các quý thầy cô để em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn công việc thực tế sau này.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng em và được sự hướng dẫn của TS. Huỳnh Ngọc Tú. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào chúng em xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Nguyễn Đức Tú*

*Nguyễn Đình Luân*

*Phạm Tấn Kiệt*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Đối với đời sống hiện đại của chúng ta QR code đã ăn sâu vào mọi lĩnh vực việc thiếu đi nó là chuyện không thể, với tính dễ sử dụng nhanh chóng tiện lợi nhưng song đi kèm với lợi ích là những rủi ro tiềm tàng những hacker có thể lợi dụng QR code để thực hiện những hành vi xấu. Ở chủ đề này tụi em xin tiếp cận theo thứ tự: giới thiệu cơ sở lý thuyết, giải thích các thuật toán và cuối cùng là ngữ cảnh và kịch bản bảo mật. Ở phần đầu sẽ đi sâu vào giải thích các lý thuyết định nghĩa của mã hai chiều này, giới thiệu loại, ứng dụng, phân tích cấu trúc chức năng từng thành phần. Ở phần thuật toán thì sẽ phân tích quy trình tạo mã QR và quy trình phân giải nó. Phần cuối cùng sẽ nói về những mối nguy hại, kịch bản và những biện pháp bảo mật đề ra để bảo vệ người dùng khỏi hacker.

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc57712536)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc57712537)

[TÓM TẮT iv](#_Toc57712538)

[MỤC LỤC 1](#_Toc57712539)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 2](#_Toc57712540)

[1.1 Tổng quan về QR code 3](#_Toc57712541)

[1.2 Các loại QR code 4](#_Toc57712542)

[1.3 Ứng dụng 5](#_Toc57712543)

[1.4 Cấu trúc mã QR 7](#_Toc57712544)

[2.1 Quy trình tạo mã QR 12](#_Toc57712545)

[2.1.1 Tạo chuỗi nhị phân 12](#_Toc57712546)

[2.1.2 Tạo bộ mã sửa lỗi 15](#_Toc57712547)

[2.1.3 Chọn mẫu mặt nạ phù hợp 17](#_Toc57712548)

[2.2 Quy trình phát hiện và đọc mã QR 19](#_Toc57712549)

[3.1 Ngữ cảnh: 21](#_Toc57712550)

[3.2 Phương thức thường gặp: 21](#_Toc57712551)

[3.3 Kịch bản bảo mật: 23](#_Toc57712552)

[CHƯƠNG 4 – DEMO 25](#_Toc57712553)

[4.1 Sinh QR Code: 25](#_Toc57712554)

[4.2 Đọc QR Code: 28](#_Toc57712555)

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

[Hình 1. QR code 3](#_Toc57712556)

[Hình 2. Barcode và QR code 4](#_Toc57712557)

[Hình 3. Ứng dụng QR code 7](#_Toc57712558)

[Hình 4. Cấu trúc mã QR 7](#_Toc57712559)

[Hình 5. Positioning Detection Markers 8](#_Toc57712560)

[Hình 6. Aligment Markings 8](#_Toc57712561)

[Hình 7. Timing Patterns 9](#_Toc57712562)

[Hình 8. Version Infomation 9](#_Toc57712563)

[Hình 9. Format Information 10](#_Toc57712564)

[Hình 10. Data and Error Correction Keys 10](#_Toc57712565)

[Hình 11. Quiet Zone 11](#_Toc57712566)

[Hình 12. Các phiên bản QR code 11](#_Toc57712567)

[Hình 13. (nguồn: onehack.us) 20](#_Toc57712568)

[Hình 14. Mã giả mạo 20](#_Toc57712569)

[Hình 15. QR code reader 24](#_Toc57712570)

[Hình 16. Demo 27](#_Toc57712571)

[Hình 17.Demo 28](#_Toc57712572)

[Hình 18.Demo 28](#_Toc57712573)

**DANH MỤC BẢNG**

[Bảng 1. Mode Indicator 12](#_Toc57712574)

[Bảng 2. Mã hóa dữ liệu 14](#_Toc57712575)

[Bảng 3. Hạn mức sửa lỗi 16](#_Toc57712576)

[Bảng 4. Bảng bits tương ứng từng phiên bản 18](#_Toc57712577)

**CHƯƠNG 1 – CƠ SỞ LÝ THUYẾT**

Ở đời sống hiện đại chúng ta đều không quá xa lạ với QR code những biểu tượng hình các ô vuông này xuất hiện khắp mọi nơi và gần như là phương thức nhận diện chủ yếu cho ứng dụng di động. Vậy nó là gì? Ở chương này sẽ giới thiệu về nó và các ứng dụng, cấu trúc phiên bản, ….

1.1 Tổng quan về QR code

QR (Quick Response) code có nghĩa là phản hồi nhanh, vì mục đích của QR là cho phép mã được giải mã ở tốc độ cao. Dạng mã vạch hai chiều (2D) này có thể được đọc bởi điện thoại thông minh có thể chụp ảnh và có ứng dụng chuyên để quét QR code hay một máy đọc mã vạch. Nó còn được gọi là mã vạch ma trận. Dù ban đầu QR code được sản xuất chỉ nhằm theo dõi các bộ phận trong sản xuất xe hơi do công ty Denso Wave (Nhật Bản) phát minh vào năm 1994. Nhưng vì nhu cầu đời sống xã hội nên nó phát triển trên mọi lĩnh vực như bây giờ.



Hình 1. QR code

Loại mã QR code này rất đa dụng nó có thể lưu trữ vCard, email, SMS, URL, bitcoin, social media, nội dung ký tự văn bản hay thông tin vị trí địa lí. Và tùy vào thiết bị quét mã QR, nó sẽ dẫn người dùng tới một trang web, số điện thoại, xem tin nhắn, …..

Điểm khiến cho QR code lợi thế hơn mã vạch truyền thống (Barcode hình 2) là khả năng lưu trữ dữ liệu và chia sẻ. QR code có thể lưu hàng ngàn ký tự chữ số trong khi các vạch thẳng dài một chiều của Barcode chỉ có thể lưu giữ 20 ký tự chữ số. QR code dễ sử dụng hơn nắm nhiều thông tin hơn nó sẽ giúp ích người dùng trên mọi lĩnh vực. Hơn nữa, mã vạch hai chiều này còn có thể đọc được cả hai chiều ngang dọc và từ bất kỳ hướng nào mà không bị ảnh hưởng bởi nền, chất liệu mà nó đang sử dụng. Chính vì thế mà QR code đang là xu hướng hiện tại và nó đang dần thay thế cho Barcode truyền thống.



Hình 2. Barcode và QR code

Chính vì thế mà QR code đang là xu hướng hiện tại và nó đang dần thay thế cho Barcode truyền thống.

1.2 Các loại QR code

QRcode có 2 loại chính : loại động và loại tĩnh.

* **QRcode động** (QRcode trực tiếp): Liên kết chính của chúng có thể được chỉnh sửa khi khởi tạo mà không làm thay đổi code. Code động này sẽ điều hướng người dùng đến máy chủ, nơi lưu trữ thông tin cụ để trong một database và được lập trình để tương các cụ thể với hình ảnh được quét đó. Nó tiện lợi hơn vì chúng tạo ra để áp dụng một lần và ta 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.

Ví dụ:

• App Store: Liên kết với nhiều app store chỉ với một cú quét, giúp việc quảng cáo tải xuống hiệu quả hơn.

• Social Media: Chia sẻ tất cả nền tảng truyền thông xã hội với thiết bị di động.

* **QRcode tĩnh**: loại này lưu trữ các thông tin trực tiếp qua hình thức văn bản họăc dẫn đến các trang web mà không cần chuyển hướng qua liên kết thứ cấp. Nghĩa là các thông tin hoặc liên kết không thể thay đổi hoặc chỉnh sửa(mỗi khi cần cập nhật ta phải tạo code mới) . Code tĩnh rất lý tưởng cho những người muốn mã chứa thông tin của họ vĩnh viễn.

Ví dụ:

• Dùng để lưu WiFi: Mọi người kết nối chỉ bằng một thao tác quét mã QR này.

• vCard: Thay thế cho danh thiếp, mình đã có thể chia sẻ địa chỉ, email, số điện thoại, URL web, thông tin chi tiết về công ty với một cú quét.

• Email: cung cấp cho khách hàng một cách thuận tiện để liên hệ bằng Mã này.

1.3 Ứng dụng

Kiểm kê hàng hóa, thông tin sản phẩm

Thông tin cá nhân: trên card visit.

Lưu trữ URL: điện thoại chỉ việc đọc QR Code để lấy URL, sau đó tự động mở trình duyệt.

Sử dụng tại các bến xe bus, xe lửa, tàu điện ngầm: người sử dụng khi quét mã QR Code của bến xe sẽ biết thông tin về các chuyến xe.

Sử dụng tại các Viện bảo tàng: người sử dụng chỉ cần quét mã QR Code đặt cạnh vật trưng bày là biết được thông tin chi tiết và cập nhật về đồ vật đó.

Sử dụng để mua hàng ở bất kỳ đâu: người sử dụng khi đi tàu điện ngầm, xe bus … nếu thấy thích mặt hàng đang quảng cáo trên đó có thể đặt mua ngay lập tức thông qua QR Code và Mobile Internet.

Sử dụng tại siêu thị: để biết được thông tin, hướng dẫn nấu ăn cũng như hàm lượng dinh dưỡng của những thức ăn cần mua.

Sử dụng tại các buổi hội thảo, thuyết trình, tổ chức sự kiện: người tham gia có thể sử dụng QR Code thay cho Business Card của mình.

Sử dụng với các tờ báo, tạp chí giấy: người đọc có thể quét mã QR Code được in trong tờ báo, tạp chí giấy để truy cập phiên bản online/mobile của tờ báo, tạp chí này.

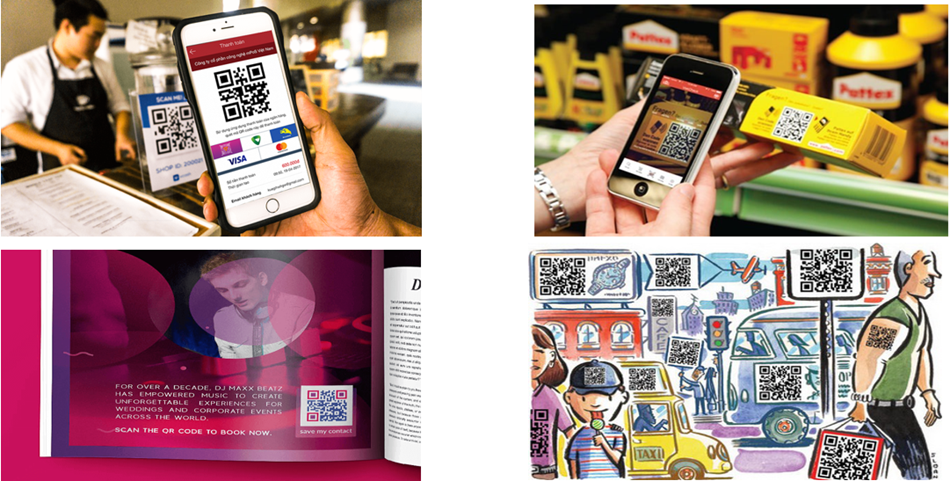
Sử dụng tại các chương trình nhạc hội, live show, bar, club: để xác định xem tiết mục, ca sỹ, ban nhạc, bài nhạc đang chơi, tác giả bài nhạc.

Sử dụng tại nhà hàng, khách sạn, coffee shop: để biết được công thức và cách chế biến món ăn, thức uống, thông tin khách sạn.

Sử dụng với đồ vật cá nhân (xe, áo thun…) : cung cấp thông tin chi tiết về món hàng, xuất xứ, giá cả.

Sử dụng trong truyền thông quảng cáo: dần thay thế các quảng cáo dưới hình thức in ấn và phát tờ bướm. QR Code sẽ đính kèm thông tin thương hiệu trên các ấn phẩm, bảng hiệu cửa hàng, các bảng quảng cáo, billboard.

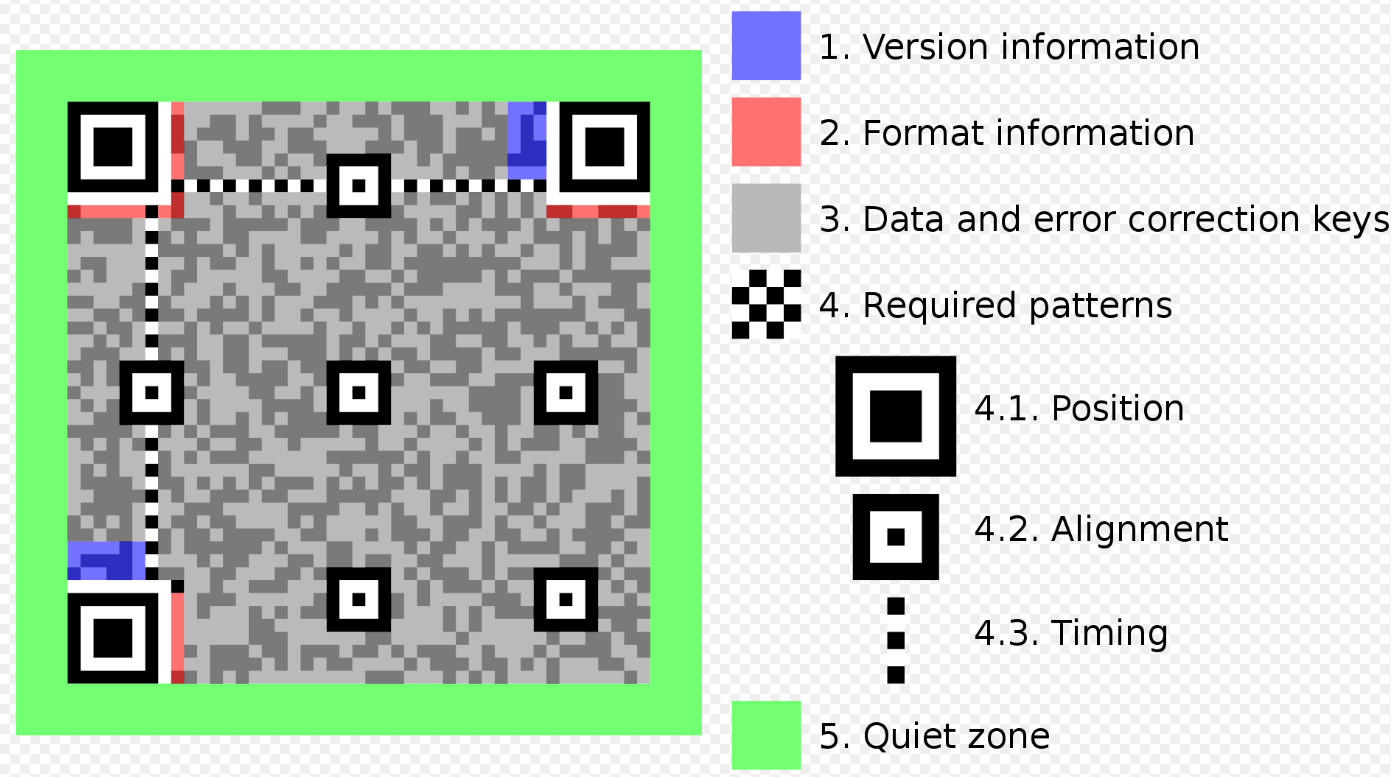
Sử dụng mã QR để tạo sự khác biệt trong chữ ký email thường dùng; trên những món quà, để tạo đặc biệt và bất ngờ cho người nhận, và làm cho người nhận thấy tò mò về nội dung của QR Code đính kèm.



Hình 3. Ứng dụng QR code

1.4 Cấu trúc mã QR

Cấu trúc chung của mã QR khá phức tạp như hình 4. Mỗi phần đều đóng vai trò quan trọng trong việc cấu thành nên một QR code hoàn chỉnh.



Hình 4. Cấu trúc mã QR

**Dưới đây sẽ giải thích chi tiết về thành phần.**



Hình 5. Positioning Detection Markers

Thành phần này nằm ở ba góc của mỗi mã, nó cho phép máy quét nhận dạng chính xác code và đọc ở tốc độ cao, đồng thời cho biết hướng Code được in. Về cơ bản, chúng giúp nhanh chóng xác định sự hiện diện của Mã QR trong một hình ảnh và định hướng của nó.



Hình 6. Aligment Markings

Những điểm này nhỏ hơn Positioning Detection Markers chúng giúp làm thẳng mã QR được vẽ trên bề mặt cong. Và, kho lưu trữ Mã càng nhiều thông tin thì mã đó càng lớn và càng cần nhiều mẫu liên kết. Định hướng thông tin.



Hình 7. Timing Patterns

Timing Patterns này là những viền trắng đen kẻ đan xen trên mã QR với ý tưởng giúp định cấu hình chính xác lưới dữ liệu. Với các line này, máy quét xác định độ lớn của ma trận dữ liệu.



Hình 8. Version Infomation

Thành phần này lưu thông tin phiên bản với 40 phiên bản QR code đang hiện hành, mỗi điểm này chỉ định cụ thể phiên bản đang sử dụng. Phổ biến hiện tại là bản 1 tới bản 7.



Hình 9. Format Information

Format Information chứa thông tin về khả năng chịu lỗi và mẫu mặt nạ dữ liệu và giúp quét Mã dễ dàng hơn.



Hình 10. Data and Error Correction Keys

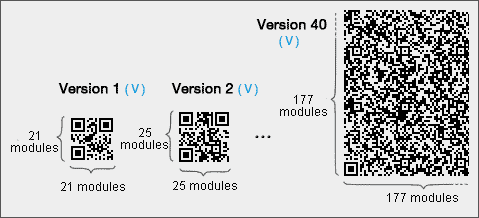
Thành phần này là cơ chế sửa lỗi vốn có trong cấu trúc mã QR là nơi chứa tất cả dữ liệu, cũng chia sẻ không gian với các khối sửa lỗi cho phép tối đa 30% mã bị hỏng.



Hình 11. Quiet Zone

Thành phần này đóng phải trò giống như một khoảng trắng trong thiết kế, nó cung cấp cấu trúc cải thiện và khả năng hiểu. Cho ai hoặc những gì ta có thể hỏi? Đối với chương trình quét. Để phân biệt Mã QR với môi trường xung quanh, Quiet Zone là rất quan trọng.

**Các phiên bản của QR code:**



Hình 12. Các phiên bản QR code

Với 40 phiên bản khác nhau hiện hành. Phiên bản phổ biến nhất là từ 1 đến 7.

**CHƯƠNG 2 – THUẬT TOÁN**

Chương này sẽ đưa ra lý thuyết thuật toán tạo mã QR và giải mã QR. Chúng em sẽ làm Demo cho phần này.

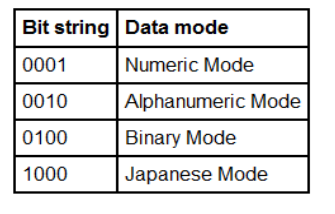
* 1. Quy trình tạo mã QR
     1. Tạo chuỗi nhị phân

Mục đích của phần này là tạo ra một chuỗi nhị phân bao gồm dữ liệu và thông tin về chế độ mã hóa, cũng như độ dài dữ liệu.

Trong phần này, ví dụ mã hóa sẽ là chuỗi HELLO WORLD sử dụng QR code phiên bản 1, hạn mức sửa lỗi Q.

Triển khai [2]:

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



Bảng 1. Mode Indicator

Do chúng ta mã hóa kiểu chuỗi, nên Mode Indicator ở đây sẽ là 0010.

Chuỗi nhị phân: 0010

***Bước 2: Mã hóa độ dài của dữ liệu***

Ở bước này, ta xác định có bao nhiêu ký tự thông điệp, chuyển độ dài nó thành một số nhị phân.

Với chuỗi HELLO WORLD, có 11 ký tự gồm khoảng trắng. Sang nhị phân: 1011

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 đên khi đủ 9 bit: 000001011. Dưới đây là danh sách các bits và phiên bản sử dụng:

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.

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

Đối với các Mode Indicator khác thì đều có cách mã hóa dữ liệu tương ứng.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | *HE* | *LL* | *O(Space)* | *WO* | *RL* | *D* |
|  | *(45\*17)+14* | *(45\*21)+21* | *(45\*24)+36* | *(45\*32)+24* | *(45\*27)+21* | *13* |
|  | *779* | *966* | *1116* | *1464* | *1236* | *13* |
| *0010 000001011* | *01100001011* | *0111100010* | *10001011100* | *10110111000* | *10011010100* | *0001101* |

Bảng 2. Mã hóa dữ liệu

***Bước 4: Hoàn thành các bit***

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 ra 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 chuối 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

* + 1. Tạo bộ mã sửa lỗi

Bộ mã sửa lỗi có các khối dữ liệu dự phòng để đảm bảo rằng mã QR vẫn đọc được dù một phần không nhận diện được.

Triển khai [3]:

***Bước1: 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. Tham khảo bảng sau:

|  |  |
| --- | --- |
| Cấp độ | Lỗi cho phép |
| L | 7% |
| M | 15% |
| H | 25% |
| Q | 30% |

Bảng 3. Hạn mức sửa lỗi

***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: 32x25 + 91x24 + 11x23 + 120x22 + 209x21 + 114x20 + 220x19 + 77x18 + 67x17 + 64x16 + 236x15 + 17x14 + 236x13

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 32x25

***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 α (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 – α) (x – α 2) … (x – α 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 – α) 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: α0 x25 + α74 x24 + α152 x23 + α176 x22 + α100 x21 + α86 x20 + α100 x19 + α106 x18 + α104 x17 + α130 x16 + α218 x15 + α206 x14 + α140 x13 + α78 x12

Dưới đây là 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

* + 1. Chọn mẫu mặt nạ phù hợp

Ở bước này tất cả dữ liệu đã được mã hóa, việc còn lại là 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 ta 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 ta dựa theo mỗi một số điểm xử phạt theo các quy tắc quy định trong tiêu chuẩn mã QR. Sau đó, chọn lọc ra mã QR có số điểm tốt nhất.

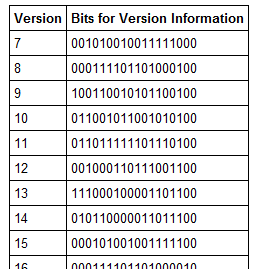
Triển khai [3]:

***Bước 1: Tạo mã QR code***

Với QR code phiên bản 1, ta cần một ma trận 21×21 (phụ thuộc vào phiên bản). Tất cả các mã QR code đều có một vài mẫu nhận diện chắc chắc phải đc thể hiện. 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.

***Bước 2: Thêm thông tin phiên bản***

Với QR 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.



Bảng 4. Bảng bits tương ứng từng phiên bản

***Bước 3: Thêm dữ liệu Bits***

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 tôi sử dụng các mẫu mặt nạ để quyết định xem có nên thay đổi màu sắc của các bit hiện tại hay không.

***Bước 4: Hoàn tất mã QR.***

Lặp lại các bước trên 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 sử dụng các mô hình mặt nạ tốt nhất có thể.

2.2 Quy trình phát hiện và đọc mã QR

Các bước thực hiện:

***B1: Phát hiện Position Detection Patterns.***

***B2: Xác định phiên bản sử dụng.***

***B3: Gỡ bỏ mặt nạ để lấy dữ liệu thô.***

***B4: Đọc cell data và cell sửa lỗi để trích rút dữ liệu cần thiết.***

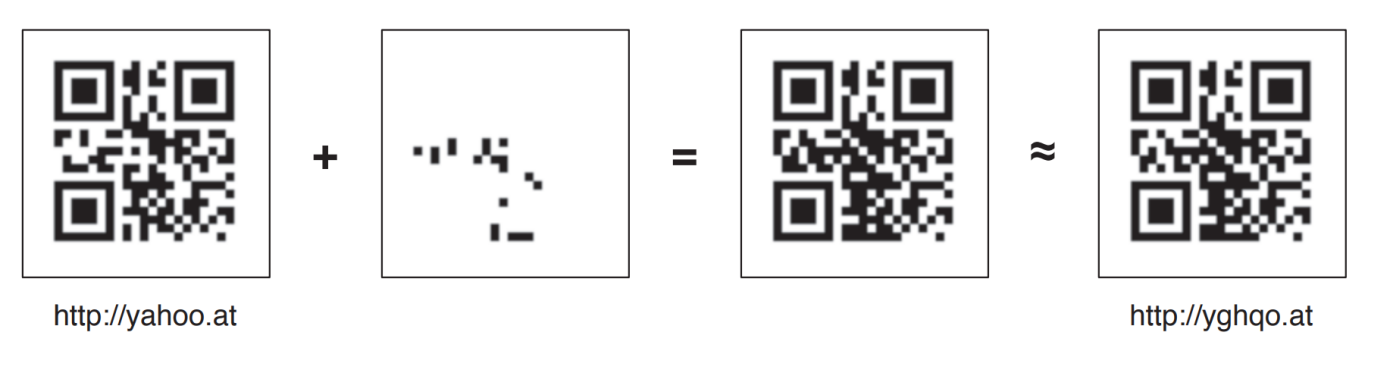
**CHƯƠNG 3 – NGỮ CẢNH, KỊCH BẢN BẢO MẬT**

Mã QRcode tuy không thể bị hack(\*) nhưng vẫn có thể được khai thác để tấn công người dùng.



Hình 13. (nguồn: onehack.us)

(\*): Do mã QR đã được mã hóa, để tấn công được, hacker phải hiểu được quá trình mã hóa, và chỉnh sửa từng ô bit nhỏ để thay đổi thông tin với điều kiện là mã bị tấn công và mã định làm giả phải có độ tương tự cao, và hơn hết nữa là hacker phải tô lên từng mã QR để hack điều đó làm giảm thiểu thiệt hại khi bị tấn công, hoặc hacker có thể in mã QR khác và dán đè lên những mã QR chính chủ.



Hình 14. Mã giả mạo

Điều này có thể được áp dụng đối với vụ tấn công có quy mô lớn như thay đổi mã QR của một số nơi quan trọng và đông như công viên, công ty, bãi giữ xe, thậm chí là dán lên gần ATM, hay gửi mail hàng loạt với nội dung như xác nhận thông tin,... Hơn nữa chúng ta không thể nhận ra được mã QR có bị thay đổi hay không do có quá nhiều ô nhỏ và các mã gần giống nhau.

3.1 Ngữ cảnh:

Ngữ cảnh : Ở đây chúng ta không nói đến việc thay đổi mã QR mà chỉ nói về việc sử dụng mã QR nguy hiểm có mục đích. Ngày nay, các phương pháp dụ dỗ người xem nhấn vào các đường link URL để liên kết tới trang web độc đã dần không còn tác dụng bởi người dùng có thể xem được đường link và phân tích chúng.

Tuy nhiên chúng ta vẫn chưa quen thuộc với loại tấn công tuy mới nhưng cũ, thay vì sử dụng URL, hacker có thể sử dụng dưới dạng mã QR để tấn công. Đa số các đối tượng nhận được mã QR độc không thể giải mã chúng như URL dẫn đến việc quét mã và bị tấn công dễ dàng hơn.

3.2 Phương thức thường gặp:

Các phương thức thường gặp :

* Phishing: việc quét mã QR dẫn đến những link của trang web mạng xã hội/ngân hàng không chính chủ có thể được khai thác để lấy thông tin cá nhân của nạn nhân.
* Mở ứng dụng: việc quét mã QR có thể dẫn đến thực hiện một số hành vi như gọi điện(gọi thông thường hay zalo,skype… những ứng dụng được cấp phép trong điện thoại), thậm chí lấy kết hợp với phishing để lấy thông tin số điện thoại của nạn nhân và làm sử dụng chúng để làm hành động “quên mật khẩu” để trang web chính chủ gửi thông tin về điện thoại nạn nhân, điều đó làm nạn nhân không phân biệt được họ có đang bị tấn công không.(shoppee chỉ gửi mã xác minh để đăng nhập chứ không gửi link, điều đó làm hacker có thể lợi dụng để lấy tài khoản hay thông tin ngân hàng).
* Link độc : QRcode dẫn đến các trang download virus hay các web độc phạm pháp

Mã QR có thể bị tấn công bởi các phương thức sau:

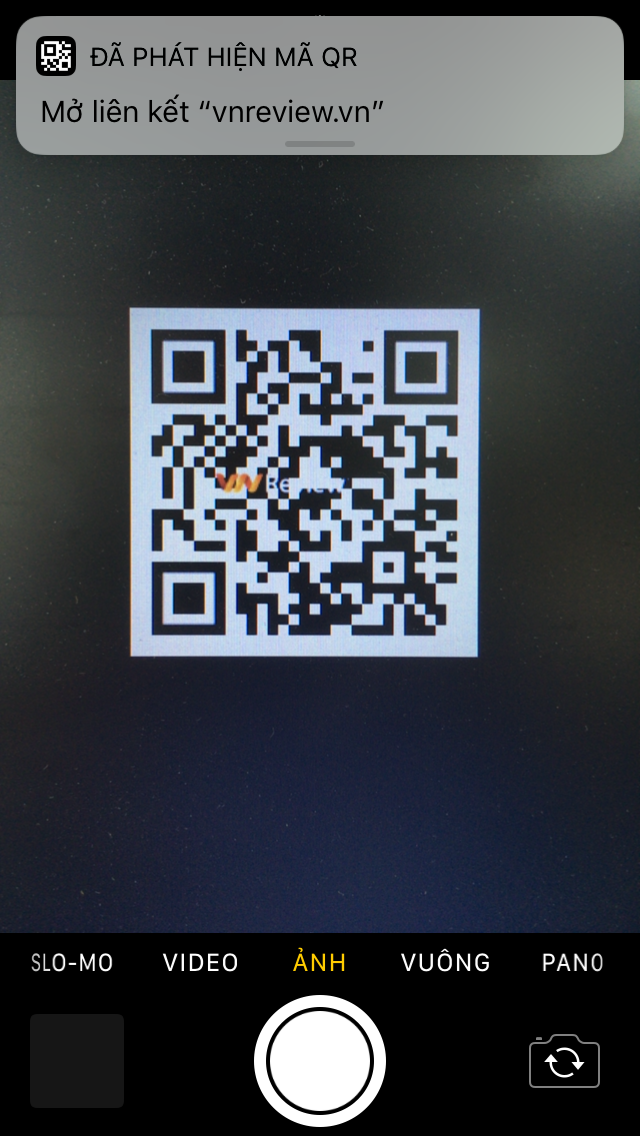
1. Thêm số liên lạc: QR code có thể thêm số liên lạc khi quét, và chúng sẽ tự động thêm mã độc trong tên liên lạc hoặc địa chỉ, và khai thác các thông tin khác của danh bạ trong điện thoại.
2. Bắt đầu cuộc gọi: khi quét QRcode, mã có thể bắt đầu một cuộc gọi, điều này có thể áp dụng để lấy thông tin như SDT, là thông tin mà các trang web thường dùng làm ID hoặc chúng có thể làm ta mất tiền từ cuộc gọi đi.
3. Gửi tin nhắn: khi quét QRcode, mã có gửi một tin nhắn, điều này có thể áp dụng để lấy thông tin như SDT, là thông tin mà các trang web thường dùng làm ID hoặc chúng có thể làm ta mất tiền từ tin nhắn đi.
4. Viết email: QR có thể tạo email và điền thông tin người gửi với chủ đề sẵn nếu ta gửi chúng đi thì chúng ta có thể bị tấn công phishing, quảng cáo, hoặc lộ địa chỉ email.
5. Thanh toán: cái này thường dùng. Nếu đang quét phải mã QR độc, chúng ta có thể vừa thực hiện 1 thao tác chuyển tiền đi nếu được cấp quyền.
6. Gửi GPS: việc quét mã QR có thể gửi thông tin GPS của ta đến 1 ứng dụng nào đó , hacker có thể mua bán thông tin ấy và chúng ta không hề thích GPS của mình bị lấy.
7. Mở trang web: quá thường dùng- có thể bị lợi dụng để tấn công phishing hoặc nội dung độc hại.
8. Tạo lịch hẹn: việc này có thể dẫn đến thông tin các cuộc hẹn hay lịch của chúng ta bị tiết lộ, giống như GPS.
9. Follow tài khoản xã hội: việc này có thể làm lộ tài khoản cá nhân và các thông tin trong tài khoản để hacker lợi dụng.
10. Đăng nhập vào wifi: như ở tiệm café, bạn có thể quét trúng mã QR dẫn đến việc kết nối wifi không chính chủ, và việc đó dẫn đến chúng có thể thao tác hoặc nghe lén tất cả các thông tin cần mạng internet của chúng ta.

3.3 Kịch bản bảo mật:

Bảo mật: trước khi quét mã QR, chúng ta có thể dùng các phần mềm QRcode reader có thể hiện nội dung của mã QR trước khi thực hiện mã. Nếu nội dung QRcode là URL, chúng ta có thể phân tích chúng hoặc không mở link thay vì tự quét mã và chạy URL như thường.

Bảo mật thông tin cá nhân, không ghi thông tin cá nhân vào URL chưa chứng thực hay nghi ngờ trong QRcode

Cài những phần mềm anti-virus để hạn chế tổn thất khi bị tấn công do download virus



Hình 15. QR code reader

Như hình trên ta có thể thấy được khi đưa ứng dụng quét đến mã QR, với nội dung của mã QR là một trang web, như bình thường thì sau khi quét mã, điện thoại của ta sẽ tự mở trang web, và với kịch bản như vậy, chúng ta sẽ thêm một bước là hiện thông tin của mã trước khi thực hiện mã để bảo đảm an toàn. Chúng ta mô phỏng lại việc tạo mã, quét mã và hiển thị nội dung mã

CHƯƠNG 4 – DEMO

4.1 Sinh QR Code:

**Bước 1: Phân tích dữ liệu**

Như đã nói ở phần thuật toán một chuẩn QR mã hóa dữ liệu theo những mode indicator. Mỗi chế độ mã hóa văn bản dưới dạng chuỗi bit (1s và 0s), nhưng mỗi chế độ sử dụng một phương pháp khác nhau để chuyển đổi văn bản thành bit và mỗi phương pháp mã hóa được tối ưu hóa để mã hóa dữ liệu bằng chuỗi bit ngắn nhất có thể. Do đó, bước đầu tiên củata nên thực hiện phân tích dữ liệu để xác định xem văn bản của bạn có thể được mã hóa ở chế độ số, chữ và số, byte hoặc Kanji hay không, sau đó chọn chế độ tối ưu nhất cho văn bản.

**Bước 2: Mã hóa dữ liệu**

Sau khi chọn được chế độ mã hóa, ta thực hiện việc mã hóa và cho ra kết quả là một chuỗi các bit được chia thành các từ mã dữ liệu có độ dài mỗi từ 8 bit.

**Bước 3: Tạo bộ mã sửa lỗi**

Sau khi tạo chuỗi các bit dữ liệu đại diện cho văn bản của mình và từ các bit đó để tạo các từ mã sửa lỗi bằng cách sử dụng một quy trình được gọi là sửa lỗi Reed-Solomon.

Máy quét QR đọc cả từ mã dữ liệu và từ mã sửa lỗi. Bằng cách so sánh cả hai, máy quét có thể xác định xem nó có đọc dữ liệu chính xác hay không và nó có thể sửa lỗi nếu đọc dữ liệu không chính xác. Phần mã hóa sửa lỗi giải thích chi tiết quá trình tạo từ mã sửa lỗi.

**Bước 4: Thông điệp cuối cùng**

Ở bước này ta cấu trúc lại thông điệp, bằng cách sắp xếp dữ liệu và mã sửa lỗi ở trên theo thứ tự thích hợp. Đối với mã QR lớn, dữ liệu và mã sửa lỗi được tạo thành các khối và các khối này phải được xen kẽ theo đặc điểm kỹ thuật của mã QR.

**Bước 5: Vị trí mô-đun trong ma trận**

Sau khi tạo các mã dữ liệu và mã sửa lỗi và sắp xếp chúng theo đúng thứ tự, ta đặt các bit vào ma trận mã QR. Các từ mã được sắp xếp trong ma trận một cách cụ thể. Trong bước này, ta cũng sẽ đặt các mẫu chung cho tất cả các mã QR, chẳng hạn như các hộp ở ba góc. Quá trình này được giải thích chi tiết trong vị trí mô-đun trong phần ma trận.

**Bước 6: Mặt nạ dữ liệu**

Một số mẫu nhất định trong ma trận mã QR có thể khiến máy quét mã QR khó đọc mã chính xác. Để chống lại điều này, đặc tả mã QR xác định tám mẫu mặt nạ, mỗi mẫu sẽ thay đổi mã QR theo một mẫu cụ thể. Ta phải xác định mẫu mặt nạ nào trong số này dẫn đến mã QR có ít đặc điểm không mong muốn nhất. Điều này được thực hiện bằng cách đánh giá từng ma trận mặt nạ dựa trên bốn quy tắc phạt. Cuối cùng sử dụng hình mặt nạ sở hữu điểm phạt thấp nhất.

**Bước 7: Định dạng và thông tin phiên bản**

Bước cuối cùng là thêm thông tin định dạng và phiên bản (nếu cần) vào mã QR bằng cách thêm pixel vào các vùng cụ thể của mã đã bị bỏ trống trong các bước trước. Các pixel định dạng xác định mức độ sửa lỗi và mẫu mặt nạ đang được sử dụng trong mã QR này. Các pixel phiên bản mã hóa kích thước của ma trận QR và chỉ được sử dụng trong các mã QR lớn hơn.

Source code phần này gồm file qrcodegen.py tuân theo 7 bước trên.

Trong đó file qrcodegen gồm:

Class QrCode: Cung cấp các hàm tính toán tạo mã QR từ dữ liệu văn bản hoặc nhị phân, vẽ mặt nạ, vẽ codeword, timming pattern, format bits, alignment pattern, position detection marker, và các hàm khác nhằm hỗ trợ tạo QR Code. Hỗ trợ phiên bản 5, các chế độ mã hóa kí tự.

Kết quả khi chạy với input là “HELLO WORLD 123”:



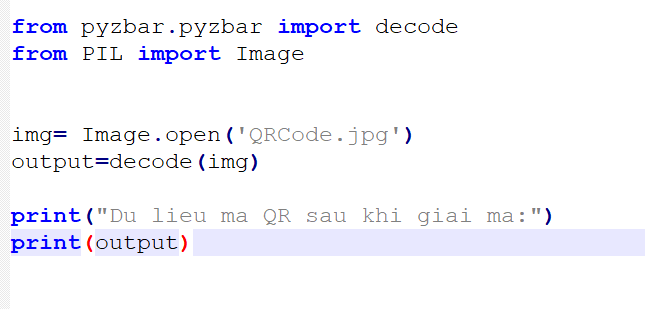
Hình . Demo

4.2 Đọc QR Code:

Về phần đọc mã QR python có thư viện pyzbar (trước đó ta phải có thư viện pillow) hỗ trợ trước tiên ta cần install pyzybar với cú pháp: pip install pyzbar.

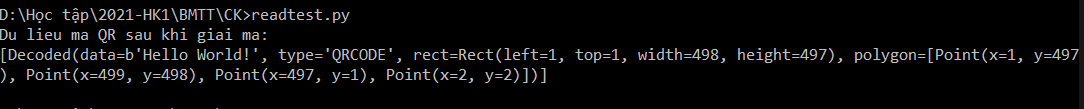
Để thực hiện "giải mã", chúng ta cần phải đọc được ảnh lên trước, chúng ta sẽ dùng thư viện PIL để thực hiện việc đọc ảnh (nếu ảnh không cùng thư mục thì hãy dùng đường dẫn tuyệt đối).

Dưới đây là đoạn code sẽ thực hiện việc “giải mã” QR code phía trên.



Hình 17.Demo

Kết quả thu được:



Hình 18.Demo

Kết quả decode mà chúng ta thu được là list danh sách các đối tượng Decoded (1 mã QR có thể chứa nhiều cụm thông tin). Mỗi đối tượng Decoded có thuộc tính data chính là nội dung của thông tin chúng ta encode lúc trước, kèm theo nó là các thuộc tính khác liên quan đến vị trí đặt các ô vuông chứa thông tin.

# **TÀI LIỆU THAM KHẢO**

1. *csa*. (n.d.). Retrieved from csa: https://www.csa.gov.sg/gosafeonline/go-safe-for-me/homeinternetusers/qr-code-falling-prey-to-malicious-website
2. *researchgate*. (n.d.). Retrieved from researchgate: https://www.researchgate.net/publication/303653249\_Malicious\_Pixels\_Using\_QR\_Codes\_as\_Attack\_Vector
3. *tinyurl*. (n.d.). Retrieved from tinyurl: https://tinyurl.com/y4jy7gj6
4. *tinyurl*. (n.d.). Retrieved from tinyurl: https://tinyurl.com/y2kvmtp7
5. *tinyurl*. (n.d.). Retrieved from tinyurl: https://tinyurl.com/y5rtaqmh
6. *wikipedia*. (n.d.). Retrieved from wikipedia: https://vi.wikipedia.org/wiki/M%C3%A3\_QR
7. *wikipedia*. (n.d.). Retrieved from wikipedia: https://en.wikipedia.org/wiki/QR\_code#Encoding