

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN CUỐI KÌ MÔN IOT CƠ BẢN

**Đề tài: Thiết kế hệ thống khóa cửa bằng
thẻ từ, xem trạng thái truy nhập
qua web server**

Người hướng dẫn: **GV. TRẦN TRUNG TÍN**

Người thực hiện: **PHẠM NGUYỄN HOÀNG QUÂN - 51900419**

PHẠM TRƯỜNG GIANG - 51900793

Khoá : 23

HỒ CHÍ MINH, NĂM 2022

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



ĐỒ ÁN CUỐI KÌ MÔN IOT CƠ BẢN

**Đề tài: Thiết kế hệ thống khóa cửa bằng
thẻ từ, xem trạng thái truy nhập
qua web server**

Người hướng dẫn: **GV. TRẦN TRUNG TÍN**

Người thực hiện: **PHẠM NGUYỄN HOÀNG QUÂN - 51900419**

PHẠM TRƯỜNG GIANG - 51900793

Khoá : 23

HỒ CHÍ MINH, NĂM 2022

LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn những góp ý từ thầy. Từ đó có thể tìm ra được hướng giải quyết và sửa đổi sao cho đồ án hoàn thiện một cách chỉnh chu nhất. Mong thầy luôn giữ được lửa nhiệt huyết trong công việc, để những thế hệ học sinh sau này ngày càng đạt được những thành tích lớn hơn. Từ đó góp phần mang lại niềm tự hào cho trường đại học Tôn Đức Thắng nói riêng và đất nước chúng ta nói chung.

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 tôi và được sự hướng dẫn của Gv.Trần Trung Tín. 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 tôi 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)

Quân

Phạm Nguyễn Hoàng Quân

Giang

Phạm Trường Giang

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)

MỤC LỤC

LỜI CẢM ƠN	i
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	iii
MỤC LỤC.....	1
DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ	3
CHƯƠNG 1 – TỔNG QUAN ĐỒ ÁN	5
1.1 Giới thiệu đề tài.....	5
1.2 Lý do lựa chọn đề tài.....	5
CHƯƠNG 2 – LÝ THUYẾT	6
2.1 IOT là gì.....	6
2.1.1 Thuật ngữ.....	6
2.1.2 Lịch sử hình thành	6
2.1.3 Xu hướng và tính chất.....	6
2.1.4 Khả năng định danh độc nhất.....	7
2.1.5 Ứng dụng của IoT	7
2.2 Mạch Arduino là gì ?	8
2.2.1 Khái niệm và lịch sử hình thành.....	8
2.2.2 Cấu tạo mạch Arduino.....	8
2.2.3 Phần mềm và hướng phát triển.....	9
2.2.3.1 Phần mềm	9
2.2.3.1 Phát triển	10
2.2.4 Ứng dụng.....	10
2.3 Module Wireless ESP8266.....	11
2.3.1 ESP8266 là gì.....	11
2.3.2 Ứng dụng ESP8266.....	11
2.4 Thẻ từ và một số thiết bị liên kết khác	11
2.3.3 Thẻ từ.....	11

2.3.4 Liên kết giữa các thiết bị.	12
CHƯƠNG 3 – MÔ TẢ HỆ THỐNG	13
3.1 Tổng quan hệ thống.....	13
3.2 Lợi ích của hệ thống.....	13
3.3 Nguyên lí hoạt động và sơ đồ khối.	14
3.4 Sơ đồ lắp đặt và nối dây.	16
3.5 Mô tả thiết bị có trong hệ thống.....	17
3.6 Thông tin kết nối port.....	19
3.7 Code nạp cho mạch Arduino Uno R3	20
3.8 Code nạp cho mạch ESP8266	23
3.9 Server nhận dữ liệu	25
3.10 Link github source code bài đồ án	26
CHƯƠNG 4 – KẾT LUẬN.....	27
BẢNG PHÂN CÔNG CÔNG VIỆC	28

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

DANH MỤC HÌNH

Hình 1. Một chương trình diễn hình cho một bộ vi điều khiển đơn giản	10
Hình 2. Sơ đồ nguyên lí hoạt động cơ bản.....	14
Hình 3. Sơ đồ thuật toán hoạt động chi tiết.....	15
Hình 4. Sơ đồ lắp đặt và nối dây thiết bị.....	16
Hình 5. Khai báo thư viện và các biến	20
Hình 6. Hàm khởi tạo các giá trị ban đầu	20
Hình 7. Hàm sử dụng cho đèn LED tín hiệu.....	21
Hình 8. Hàm bật tắt còi báo hiệu.....	21
Hình 9. Hàm kiểm tra UID của thẻ từ (thẻ tag)	21
Hình 10. Hàm xử lý việc đọc tín hiệu từ thẻ từ (thẻ tag)	22
Hình 11. Hàm kiểm tra số lần truy cập sai quá 3 lần và bật còi cảnh báo	22
Hình 12. Vòng lặp chính	23
Hình 13. Khai báo thư viện và các biến	23
Hình 14. Hàm xử lý và kết nối với wifi	24
Hình 15. Hàm đọc nhận dữ liệu từ Arduino và gửi lên server.....	24
Hình 16. Hàm kết nối với server Node JS	25
Hình 17. Vòng lặp chính	25
Hình 18. Server nhận và xử lý dữ liệu lên MongoDB	25

DANH MỤC BẢNG

Bảng 1. Thiết bị.....	18
Bảng 2. Kết nối mạch Arduino và mạch RC522.....	19

Bảng 3. Kết nối mạch Arduino và mạch ESP8266	19
Bảng 4. Kết nối mạch Arduino và còi Buzzer	19
Bảng 5. Kết nối mạch Arduino và đèn LED	19

CHƯƠNG 1 – TỔNG QUAN ĐỒ ÁN

1.1 Giới thiệu đề tài

Trong thời cuộc công nghệ và hiện đại hóa đã và đang phát triển một cách chóng mặt hiện nay. Việc bảo quản tài sản cá nhân được xem như một nhu cầu tất yếu đối với mỗi cá nhân cũng như mỗi gia đình.

Hầu hết các công cụ, phương pháp chống trộm thông thường như ổ khóa cơ, kết sắt, thậm chí bảo vệ trực tiếp bằng con người ngày càng lỗi thời và mang tính bảo mật không cao. Tiềm tàng những nguy cơ mất trộm có thể mang lại rủi ro lớn đối với cá nhân, doanh nghiệp

Xuất phát từ những nhu cầu như vậy, hệ thống chống trộm hiện đại qua các thiết bị điện tử được lập trình một cuộc cách mạng. Mang đến giải pháp tốt hơn rất nhiều so với trước đây.

1.2 Lý do lựa chọn đề tài

Thông qua môn học cũng như nhìn thấy được khả năng kết nối giữa IOT và việc bảo mật mang lại nhiều lợi ích và thuận tiện, giúp cho việc giữ an toàn cho tài sản trở nên dễ dàng và bảo mật hơn rất nhiều. Đó cũng chính là lý do chúng em lựa chọn đề tài “Thiết kế hệ thống khóa cửa bằng thẻ từ, báo truy nhập bất thường qua điện thoại”.

Đồ án là một hệ thống thông minh có thể sử dụng để thay thế các loại khóa cơ sử dụng chìa khóa hoặc mật khẩu thông thường bằng thẻ từ hoặc có thể mở trực tiếp thông qua điện thoại thông minh có kết nối mạng. Nếu phát hiện có truy cập bất thường lập tức thông báo cho người dùng. Từ đó người dùng có thể chủ động được trong việc ngăn chặn và bảo quản tài sản tốt hơn rất nhiều.

CHƯƠNG 2 – LÝ THUYẾT

2.1 IOT là gì

2.1.1 Thuật ngữ

Internet Vạn Vật (tiếng Anh, viết tắt: IoT) là thuật ngữ dùng để chỉ các đối tượng có thể được nhận biết (identifiable) cũng như chỉ sự tồn tại của chúng trong một kiến trúc mạng tính kết nối. Cụm từ này được đưa ra bởi Kevin Ashton vào năm 1999. Năm 2013, tổ chức Global Standards Initiative on Internet of Things (IoT-GSI) định nghĩa IoT là "hạ tầng cơ sở toàn cầu phục vụ cho xã hội thông tin, hỗ trợ các dịch vụ (điện toán) chuyên sâu thông qua các vật thể (cả thực lẫn ảo) được kết nối với nhau nhờ vào công nghệ thông tin và truyền thông hiện hữu được tích hợp.

2.1.2 Lịch sử hình thành

Năm 1994 Reza Raji mô tả khái niệm này trên tờ IEEE Spectrum là chuyển các gói dữ liệu nhỏ sang tập hợp các nút mạng lớn, để tích hợp và tự động hóa mọi thứ từ các thiết bị gia dụng với cả một nhà máy sản xuất.

Năm 1999, Kevin Ashton đã đưa ra cụm từ Internet of Things nhằm để chỉ các đối tượng có thể được nhận biết cũng như sự tồn tại của chúng.

Đến năm 2016, Internet Vạn Vật khẳng định được bước tiến của mình nhờ sự hội tụ của nhiều công nghệ, bao gồm truyền tải vô tuyến hiện diện đầy đặc, phân tích dữ liệu thời gian thực, học máy, cảm biến hàng hóa, và hệ thống nhúng.

2.1.3 Xu hướng và tính chất

Điểm quan trọng của IoT đó là các đối tượng phải có thể được nhận biết và định dạng (identifiable). Nếu mọi đối tượng, kể cả con người, được "đánh dấu" để phân biệt

bản thân đối tượng đó với những thứ xung quanh thì chúng ta có thể hoàn toàn quản lý được nó thông qua máy tính. Việc đánh dấu (tagging) có thể được thực hiện thông qua nhiều công nghệ, chẳng hạn như RFID, NFC, mã vạch, mã QR, watermark kỹ thuật số... Việc kết nối thì có thể thực hiện qua Wi-Fi, mạng viễn thông băng rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại...

Ngoài những kỹ thuật nói trên, nếu nhìn từ thế giới web, chúng ta có thể sử dụng các địa chỉ độc nhất để xác định từng vật, chẳng hạn như địa chỉ IP. Mỗi thiết bị sẽ có một IP riêng biệt không nhầm lẫn. Sự xuất hiện của IPv6 với không gian địa chỉ cực kỳ rộng lớn sẽ giúp mọi thứ có thể dễ dàng kết nối vào Internet cũng như kết nối với nhau.

2.1.4 Khả năng định danh độc nhất

Sự thông minh và tự động trong điều khiển thực chất không phải là một phần trong ý tưởng về IoT. Các máy móc có thể dễ dàng nhận biết và phản hồi lại môi trường xung quanh (ambient intelligence), chúng cũng có thể tự điều khiển bản thân (autonomous control) mà không cần đến kết nối mạng.

Kiến trúc dựa trên sự kiện :Các thực thể, máy móc trong IoT sẽ phản hồi dựa theo các sự kiện diễn ra trong lúc chúng hoạt động theo thời gian thực. Một số nhà nghiên cứu từng nói rằng một mạng lưới các sensor chính là một thành phần đơn giản của IoT.

Kích thước :Một mạng lưới IoT có thể chứa đến 50 đến 100 nghìn tỉ đối tượng được kết nối và mạng lưới này có thể theo dõi sự di chuyển của từng đối tượng. Một con người sống trong thành thị có thể bị bao bọc xung quanh bởi 1000 đến 5000 đối tượng có khả năng theo dõi.

2.1.5 Ứng dụng của IoT

IoT có ứng dụng rộng vô cùng, có thể kể ra một số thứ như sau:

- ✓ Quản lý chất thải
- ✓ Quản lý và lập kế hoạch quản lý đô thị

- ✓ Quản lý môi trường
- ✓ Phản hồi trong các tình huống khẩn cấp
- ✓ Mua sắm thông minh
- ✓ Quản lý các thiết bị cá nhân
- ✓ Đồng hồ đo thông minh
- ✓ Tự động hóa ngôi nhà

2.2 Mạch Arduino là gì ?

2.2.1 Khái niệm và lịch sử hình thành.

Arduino là một nền tảng mã nguồn mở được sử dụng để xây dựng các ứng dụng điện tử tương tác với nhau hoặc với môi trường. Arduino giống như một máy tính thu nhỏ để người dùng có thể lập trình và thực hiện các dự án điện tử mà không cần phải có các công cụ chuyên biệt để phục vụ việc nạp code. Arduino tương tác với thế giới thông qua các cảm biến điện tử, đèn, và động cơ.

Được giới thiệu vào năm 2005, những nhà thiết kế của Arduino cố gắng mang đến một phương thức dễ dàng, không tốn kém cho những người yêu thích, sinh viên và giới chuyên nghiệp để tạo ra những thiết bị có khả năng tương tác với môi trường thông qua các cảm biến và các cơ cấu chấp hành.

2.2.2 Cấu tạo mạch Arduino.

Một mạch Arduino bao gồm một vi điều khiển AVR với nhiều linh kiện bổ sung giúp dễ dàng lập trình và có thể mở rộng với các mạch khác.

Có rất nhiều các phiên bản mạch Arduino khác nhau và chúng có thể được sử dụng với nhiều mục đích. Nhưng hầu hết các mạch đều giống nhau về các thành phần chính:

- ✓ Nguồn (USB/ Barrel Jack)
- ✓ Các chân (5V, 3.3V, GND, Digital, Analog, PWM, ISF)

- ✓ Nút Reset (Reset Button)
- ✓ Đèn LED báo nguồn (Power LED Indicator)
- ✓ LED TX và RX (TX RX LEDs)
- ✓ IC chủ (Main IC)
- ✓ Bộ điều chỉnh điện áp (Voltage Regulator)

2.2.3 Phần mềm và hướng phát triển.

2.2.3.1 Phần mềm

Môi trường phát triển tích hợp (IDE) của Arduino là một ứng dụng cross-platform (đa nền tảng) được viết bằng Java, và từ IDE này sẽ được sử dụng cho Ngôn ngữ lập trình xử lý (Processing programming language) và project Wiring. Nó được thiết kế để dành cho những người mới tập làm quen với lĩnh vực phát triển phần mềm. Nó bao gồm một chương trình code editor với các chức năng như đánh dấu cú pháp, tự động brace matching, và tự động canh lề, cũng như compile(biên dịch) và upload chương trình lên board chỉ với 1 cú nhấp chuột. Một chương trình hoặc code viết cho Arduino được gọi là một sketch.

Người dùng chỉ cần định nghĩa 2 hàm để tạo ra một chương trình vòng thực thi (cyclic executive) có thể chạy được:

- ✓ setup(): hàm này chạy mỗi khi khởi động một chương trình, dùng để thiết lập các cài đặt
- ✓ loop(): hàm này được gọi lặp lại cho đến khi tắt nguồn board mạch

```

1  #define LED_PIN 13
2
3  void setup () {
4    pinMode (LED_PIN, OUTPUT); // Đặt chân 13 làm đầu ra digital
5  }
6
7  void loop () {
8    digitalWrite (LED_PIN, HIGH); // Bật LED on
9    delay (1000); // chờ trong 1 giây (1000 mili giây)
10   digitalWrite (LED_PIN, LOW); // Tắt LED off
11   delay (1000); // chờ trong 1s
12 }

```

Hình 1. Một chương trình đèn hình cho một bộ vi điều khiển đơn giản

2.2.3.1 Phát triển

Arduino là một nền tảng phần cứng mã nguồn mở: Các thiết kế phần cứng tham khảo của Arduino được phân phối dưới dạng Creative Commons Attribution Share-Alike 2.5 license và có sẵn trên website của Arduino. Một vài phiên bản phần cứng của Arduino còn đưa lên cả file Layout và thành phẩm. Mã nguồn cho IDE này cũng khả dụng và được xuất bản dưới dạng GNU General Public License, version 2.

2.2.4 Ứng dụng.

Nói tới ứng dụng của Arduino phải kể tới một số lĩnh vực như sau:

- ✓ Arduino làm Robot với khả năng đọc những thiết bị cảm biến, điều khiển động cơ.... Arduino giúp bộ xử lý trung tâm hoạt động nhiệm vụ của mình qua nhiều loại robot.
- ✓ Game tương tác: Arduino sử dụng để tương tác với màn hình, Joystick,... khi chơi game như phá gạch, Mario, Tetris,...
- ✓ Máy bay không có người lái.
- ✓ Điều khiển đèn tín hiệu giao thông, hiệu ứng đèn led nhấp nháy.
- ✓ Làm máy in 3D.

- ✓ Thiết kế đàn bằng ánh sáng.
- ✓ Làm lò nướng bánh có tweet cảnh báo khi bánh chín.

2.3 Module Wireless ESP8266.

2.3.1 *ESP8266 là gì.*

ESP8266, hay gọi đầy đủ là ESP8266EX là một vi mạch Wi-Fi giá rẻ, có hỗ trợ bộ giao thức TCP/IP và có thể tích hợp vào thành phần của vi điều khiển, được sản xuất bởi hãng Espressif Systems ở Thượng Hải, Trung Quốc.

ESP8285 là một chip ESP8266 với 1 MiB bộ nhớ flash được tích hợp, cho phép người dùng có thể sản xuất các thiết bị có khả năng kết nối với Wi-Fi chỉ với một chip đơn

2.3.2 *Ứng dụng ESP8266.*

ESP8266 có thể được dùng làm module Wifi bên ngoài, sử dụng firmware tập lệnh AT tiêu chuẩn bằng cách kết nối nó với bất kỳ bộ vi điều khiển nào sử dụng UART nối tiếp hoặc trực tiếp làm bộ vi điều khiển hỗ trợ Wifi, bằng cách lập trình một chương trình cơ sở mới sử dụng SDK được cung cấp. Các chân GPIO cho phép IO Analog và Digital, cộng với PWM, SPI, I2C, v.v. ESP8266 có nhiều ứng dụng khi nói đến Io.

2.4 Thẻ từ và một số thiết bị liên kết khác

2.3.3 *Thẻ từ.*

Thẻ từ là những chiếc thẻ bằng nhựa được thiết kế với kích thước tiêu chuẩn theo quy định sử dụng để lưu trữ dữ liệu. Chìa khóa của loại khóa cửa từ này là một loại thẻ có hình dạng hệt như một chiếc thẻ tín dụng, thẻ ngân hàng. Chiếc thẻ từ này sẽ được sản xuất với lõi ngầm bên trong có chứa hệ thống chip và vi mạch để có thể tương hợp với đầu đọc thẻ bên trong khóa.

Mỗi thẻ này được lập trình điện tử với thông tin duy nhất được cài đặt sẵn. Mỗi người dùng sẽ chỉ sử dụng một thẻ, mỗi thẻ có một mã số duy nhất không trùng lặp với bất kỳ mã số thiết bị nào khác, đồng thời các thông tin chi tiết về mỗi cá nhân được lưu trong phần mềm ứng với một mã thẻ và được bảo mật hoàn toàn với người ngoài.

2.3.4 Liên kết giữa các thiết bị.

Có nhiều cách khác nhau để thiết lập liên kết giữa Arduino với điện thoại thông minh. Ở demo chúng em có sử dụng những thiết bị sau :

- ✓ Giao thức HTTP: có thể gửi và nhận dữ liệu bằng giao thức HTTP. Yêu cầu HTTP POST và GET được sử dụng để thiết lập liên kết. Có thể sử dụng mô-đun ESP8266-NodeMCU. Mô-đun này có thể tạo điều kiện kết nối và thậm chí có cổng GPI
- ✓ OneSignal: là một trong những công cụ Push Notifications giúp tạo ra những thông báo đẩy trên website, được rất nhiều nhà quản trị website ưu tiên sử dụng

CHƯƠNG 3 – MÔ TẢ HỆ THỐNG

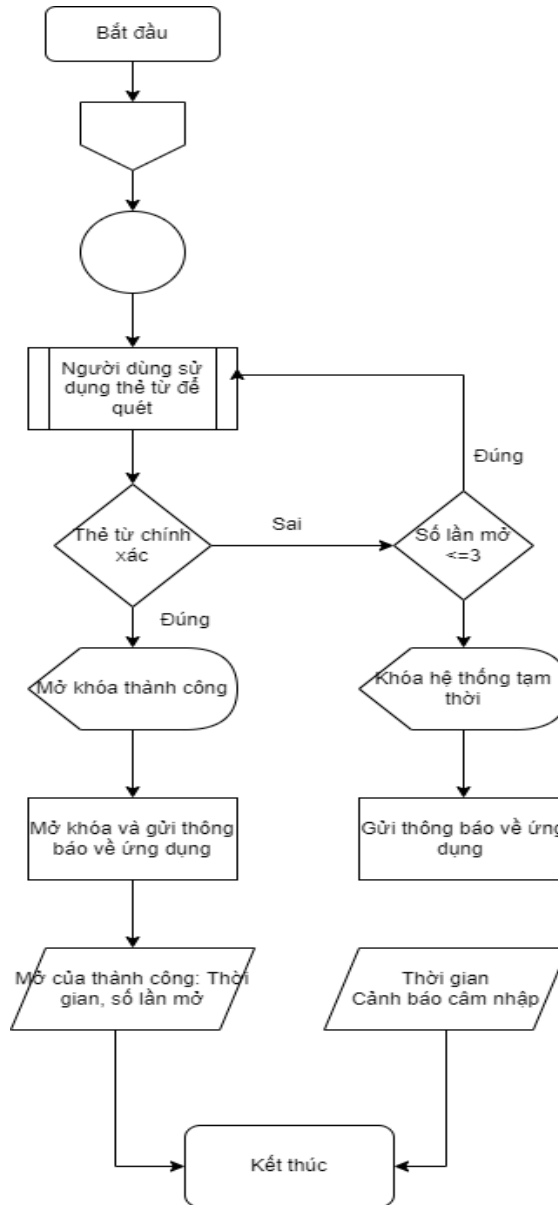
3.1 Tổng quan hệ thống

Hệ thống cho phép sử dụng thẻ cảm ứng từ để mở khóa. Nếu có sự cố xảy ra trong khi hoạt động hệ thống sẽ nhận được tin nhắn cảnh báo thông qua thiết bị di động.

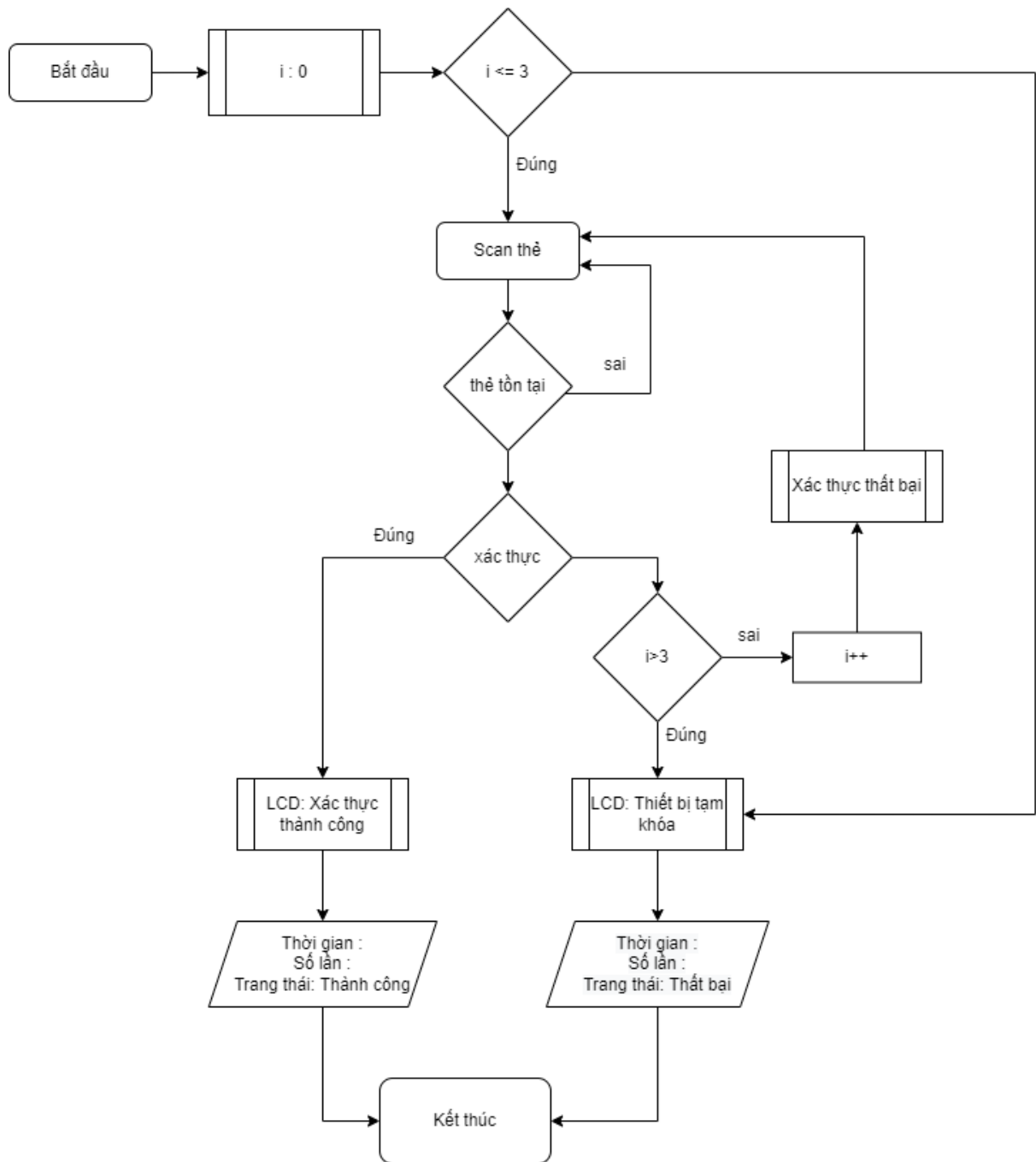
3.2 Lợi ích của hệ thống

Hệ thống là biện pháp thay thế con người giám sát tài sản nhằm làm giảm thời gian cùng sự lo lắng về mức độ an toàn của tài sản. Đi kèm đó là khả năng chủ động rất cao khi có sự cố xảy ra. Ngoài ra có thể theo dõi được hệ thống từ bất kỳ đâu thông qua thiết bị di động

3.3 Nguyên lí hoạt động và sơ đồ khối.

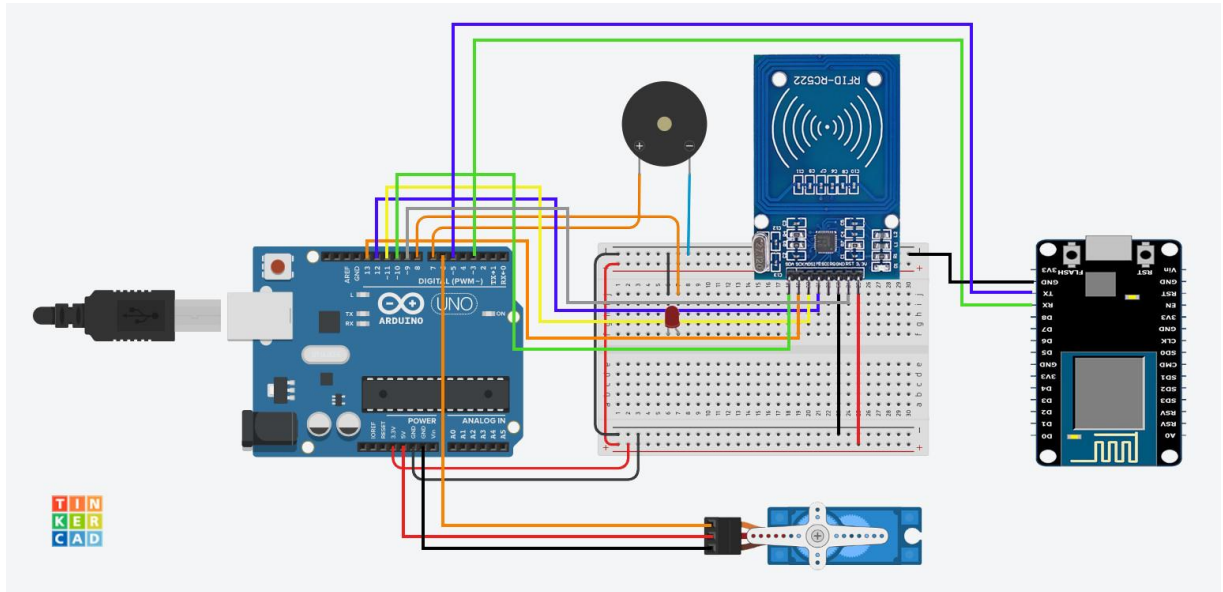


Hình 2. Sơ đồ nguyên lí hoạt động cơ bản



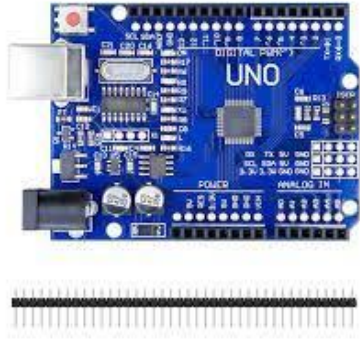

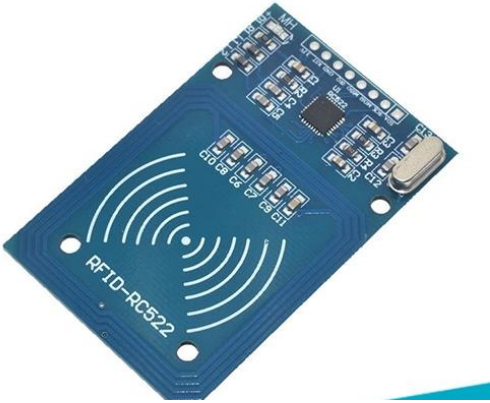
Hình 3. Sơ đồ thuật toán hoạt động chi tiết




3.4 Sơ đồ lắp đặt và nối dây.



Hình 4. Sơ đồ lắp đặt và nối dây thiết bị

3.5 Mô tả thiết bị có trong hệ thống

STT	Tên thiết bị	Hình ảnh	Mô tả	Số lượng
1	Arduino uno R3		Mạch điều khiển chính	1
2	Module wireless ESP8266		Kết nối internet, nhận tín hiệu từ arduino	1
3	RFID – RC522		Xử lý dữ liệu từ thẻ từ	1

4	Thẻ từ	 <p>Các loại thẻ thanh máy hiện nay</p>	Trả về UID	1
5	Còi báo hiệu		Báo hiệu khi thẻ sai UID	1
6	Servo		Động cơ chính để đóng mở chốt cửa	1

Bảng 1. Thiết bị

3.6 Thông tin kết nối port

ARDUINO	D10	D13	D11	D12	#N/A	GND	D9	3.3V
RC522	SS	SCK	MOSI	MISO	IRQ	GND	RST	VCC

Bảng 2. Kết nối mạch Arduino và mạch RC522

ARDUINO	GND	5	3
ESP8266	GND	TX	RX

Bảng 3. Kết nối mạch Arduino và mạch ESP8266

ARDUINO	GND	D7
BUZZER	GND	+

Bảng 4. Kết nối mạch Arduino và còi Buzzer

ARDUINO	GND	D8
LED	GND	+

Bảng 5. Kết nối mạch Arduino và đèn LED

3.7 Code nạp cho mạch Arduino Uno R3

```

1  #include <SPI.h>
2  #include <MFRC522.h>
3  #include <SoftwareSerial.h>
4
5  #define BAUD_RATE 115200
6  #define SS_PIN 10
7  #define RST_PIN 9
8  #define SIGNAL_LED 8
9  #define BUZZER 7
10 #define SERVO 6
11 #define ESP_RX_PIN 3
12 #define ESP_TX_PIN 5
13
14 SoftwareSerial espSerial(ESP_TX_PIN, ESP_RX_PIN);
15 MFRC522 rfid(SS_PIN, RST_PIN); // Instance of the class
16
17 MFRC522::MIFARE_Key key;
18
19 // Init array that will store new NUID
20 String UIDPICC = "";
21 byte UID[4];
22 // Init KEY_CARD
23 int KEY_CARD[4] = {163, 35, 126, 163};
24 //Init access temp
25 int access_tmp = 0;
26

```

Hình 5. Khai báo thư viện và các biến

```

27 void setup() {
28     access_tmp = 0;
29     Serial.begin(BAUD_RATE);
30     espSerial.begin(BAUD_RATE);
31     pinMode(SIGNAL_LED, OUTPUT);
32     pinMode(BUZZER, OUTPUT);
33     pinMode(SERVO, OUTPUT);
34     SPI.begin(); // Init SPI bus
35     rfid.PCD_Init(); // Init MFRC522
36 }
37

```

Hình 6. Hàm khởi tạo các giá trị ban đầu

```

38 void LedSignal(){
39     digitalWrite(SIGNAL_LED, HIGH);
40     delay(300);
41     digitalWrite(SIGNAL_LED, LOW);
42 }

```

Hình 7. Hàm sử dụng cho đèn LED tín hiệu

```

51 ∨ void BUZZER_ON(){
52     digitalWrite(BUZZER, HIGH);
53 }
54
55 ∨ void BUZZER_OFF(){
56     digitalWrite(BUZZER, LOW);
57 }

```

Hình 8. Hàm bật tắt còi báo hiệu

```

59 ∨ void checkValidUID(){
60 ∨ if (UID[0] == KEY_CARD[0] && UID[1] == KEY_CARD[1] && UID[2] == KEY_CARD[2] && UID[3] == KEY_CARD[3]) {
61     access_tmp = 0;
62     espSerial.print("?uid="+UIDPICC+"&status=1");
63     ServoTrigger();
64 ∨ } else {
65     espSerial.print("?uid="+UIDPICC+"&status=0");
66     access_tmp ++;
67 };
68 Serial.print("Access times: ");
69 Serial.println(access_tmp);
70 }

```

Hình 9. Hàm kiểm tra UID của thẻ từ (thẻ tag)

```

80 void RFIDReader(){
81     UIDPICC = "";
82     // Reset the loop if no new card present on the sensor/reader.
83     // This saves the entire process when idle.
84     if ( ! rfid.PICC_IsNewCardPresent()) return;
85     // Verify if the NUID has been readed
86     if ( ! rfid.PICC_ReadCardSerial()) return;
87
88     // Store NUID into nuidPICC array
89     for (byte i = 0; i < 4; i++) {
90         UID[i] = rfid.uid.uidByte[i];
91         UIDPICC.concat(String(rfid.uid.uidByte[i],HEX));
92     };
93
94     // Turn LED Signal on and print UID on Serial
95     LedSignal();
96     printDec(rfid.uid.uidByte, rfid.uid.size);
97     Serial.println();
98     Serial.println(UIDPICC);
99
100    // Check whether an UID is valid or not
101    checkValidUID();
102
103    // Halt PICC
104    rfid.PICC_HaltA();
105    // Stop encryption on PCD
106    rfid.PCD_StopCrypto1();
107 }

```

Hình 10. Hàm xử lý việc đọc tín hiệu từ thẻ từ (thẻ tag)

```

72 void checkAccess(){
73     if (access_tmp >= 3) {
74         BUZZER_ON();
75     } else {
76         BUZZER_OFF();
77     }
78 }

```

Hình 11. Hàm kiểm tra số lần truy cập sai quá 3 lần và bật còi cảnh báo

```

109 // LOOP main
110 void loop() {
111     RFIDReader();
112     // Check time access dennied
113     checkAccess();
114 }

```

Hình 12. Vòng lặp chính

3.8 Code nạp cho mạch ESP8266

```

1  #include <ESP8266WiFi.h>
2  #include <ESP8266WebServer.h>
3  #include <ESP8266HTTPClient.h>
4
5  #define WEB_SERVER_PORT 80
6  #define BAUD_RATE 115200
7
8  ESP8266WebServer webServer(WEB_SERVER_PORT);
9  WiFiClient client;
10 HTTPClient http;
11
12 String arduinoData = "0";
13 const char * SSID = "Hog1_2.4G";
14 const char * PSW = "0903701132";
15 const char * NODEJS_SERVER = "192.168.1.25";
16 const int NODEJS_PORT = 3000;

```

Hình 13. Khai báo thư viện và các biến

```

18 void wifi_connection(){
19     WiFi.disconnect();
20     WiFi.mode(WIFI_STA);
21     WiFi.begin(SSID, PSW);
22     Serial.begin(BAUD_RATE);
23     Serial.println("Wait for connecting to wifi...");
24     while (WiFi.status() != WL_CONNECTED){
25         delay(500);
26         Serial.print("Connecting to ");
27         Serial.print(SSID);
28         Serial.println("...");
29     };
30     Serial.println("Wifi connected successfully!");
31     Serial.print("ESP Local IP address: ");
32     Serial.println(WiFi.localIP());
33 }
34
35 void wifi_listening_connection(){
36     while (WiFi.status() != WL_CONNECTED){
37         delay(500);
38         Serial.print("Disconnected to ");
39         Serial.print(SSID);
40         Serial.println("...");
41     };
42 }

```

Hình 14. Hàm xử lý và kết nối với wifi

```

56 void sendDataToServer(){
57     client.print(
58         String("GET /api/v1/new-data/") +
59         String(arduinoData) + " HTTP/1.1\r\n" +
60         "Host: " + NODEJS_SERVER + "\r\n" +
61         "Connection: close\r\n\r\n"
62     );
63 }
64
65 void readDataSerial(){
66     if (Serial.available()) {
67         arduinoData = Serial.readString();
68         Serial.println(arduinoData);
69         sendDataToServer();
70     };
71 }

```

Hình 15. Hàm đọc nhận dữ liệu từ Arduino và gửi lên server

```

73 void NODEJS_listening(){
74     if (!client.connect(NODEJS_SERVER, NODEJS_PORT)) {
75         Serial.println("Connection to NodeJS server failed!");
76         return;
77     };
78 }

```

Hình 16. Hàm kết nối với server Node JS

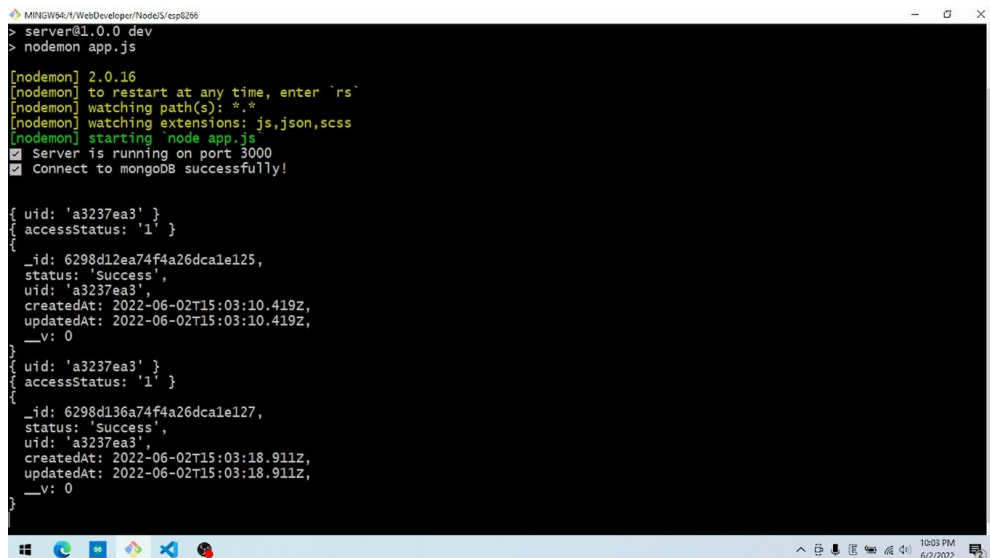
```

80 void loop() {
81     WiFiClient client;
82     while(client.available()){
83         String line = client.readStringUntil('\r');
84         Serial.print(line);
85     };
86     webServer.handleClient();
87     wifi_listening_connection();
88     NODEJS_listening();
89     readDataSerial();
90 }

```

Hình 17. Vòng lặp chính

3.9 Server nhận dữ liệu



```

MINGW64 ~/WebDeveloper/NodeJS/csp0256
> server@1.0.0 dev
> nodemon app.js

[nodemon] 2.0.16
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,json,scss
[nodemon] starting node app.js
Server is running on port 3000
Connect to mongoDB successfully!

{
  uid: 'a3237ea3'
}
{
  accessStatus: '1'
}
{
  _id: 6298d12ea74f4a26dca1e125,
  status: 'Success',
  uid: 'a3237ea3',
  createdAt: 2022-06-02T15:03:10.419Z,
  updatedAt: 2022-06-02T15:03:10.419Z,
  __v: 0
}
{
  uid: 'a3237ea3'
}
{
  accessStatus: '1'
}
{
  _id: 6298d136a74f4a26dca1e127,
  status: 'Success',
  uid: 'a3237ea3',
  createdAt: 2022-06-02T15:03:18.911Z,
  updatedAt: 2022-06-02T15:03:18.911Z,
  __v: 0
}

```

Hình 18. Server nhận và xử lý dữ liệu lên MongoDB

3.10 Link github source code bài đồ án

<https://github.com/truonggiangit793/iot-final>

CHƯƠNG 4 – KẾT LUẬN

Tổng quan, nhóm chúng em đã thiết kế và xây dựng hệ thống dựa trên mô hình thực tế và lắp đặt các thiết bị thật.

Những mục tiêu mà nhóm đã đạt được:

- Chúng em đã thiết kế và lắp đặt thành công hệ thống đóng/mở khóa cửa bằng thẻ từ thông qua công nghệ đọc thẻ RFID. Có thể áp dụng vào thực tế để thay thế ổ khoá cửa truyền thống như “Thay thế ổ khoá truyền thống ở phòng trọ, nhà ở, nhà kho...”
- Hệ thống mà nhóm chúng em đã xây dựng được tích hợp sẵn còi báo hiệu nhằm trường hợp mở khoá sai thẻ quá số lần để cảnh báo tới người dùng.

Tuy nhiên vẫn còn nhiều tính năng và quy mô chưa được tối ưu hoá và còn nhiều sai sót trong quá trình làm bài, dưới đây là những hạn chế mà chúng em chưa hoàn thiện:

- Chúng em chưa hoàn thiện được ứng dụng để người dùng có thể nhận được thông báo.

Để có thể dễ dàng ứng dụng hệ thống này vào thực tế hơn trong tương lai, nhóm chúng em sẽ tiếp tục phát triển một ứng dụng di động để người dùng có thể dễ dàng xem và nhận các thông báo từ hệ thống.

BẢNG PHÂN CÔNG CÔNG VIỆC

Stt	Mssv	Họ Tên	Công Việc	Đánh Giá	
				Hoàn Thành	Không H. Thành
1	51900793	Phạm Trường Giang	<ul style="list-style-type: none"> - Lên danh sách các thiết bị cần mua. - Mua thiết bị và linh kiện. - Lắp đặt phần cứng - Viết báo cáo chương 3, 4 và hoàn thiện báo cáo. - Lập trình và test nạp code cho mạch. - Khảo sát thực tế. 	X	
2	51900419	Phạm Nguyễn Hoàng Quân	<ul style="list-style-type: none"> - Viết báo cáo chương 1, 2. - Tìm tài liệu tham khảo. - Thiết kế powepoint. - Video demo. - Hỗ trợ lắp đặt linh kiện. - Lên kế hoạch cho nhóm thực hiện. - Khảo sát thực tế. 	X	

TÀI LIỆU THAM KHẢO

Tiếng Việt

1. <http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi>
2. <http://arduino.vn/bai-viet/77-bai-2-cach-lam-den-led-nhap-nhay-theo-yeu-cau>
3. <https://hshop.vn/products/kit-rf-thu-phat-wifi-esp8266-nodemcu>
4. <https://arduino.esp8266.vn/>
5. <https://www.arduino.cc/reference/en/language/functions/communication/serial/>
6. <https://bloglaptrinh.info/esp8266-thu-vien-wifi-manager-quan-ly-ket-noi-wifi/>
7. <https://randomnerdtutorials.com/wifimanager-with-esp8266-autoconnect-custom-parameter-and-manage-your-ssid-and-password/>
8. <https://hocarm.org/tu-dong-ket-noi-cau-hinh-esp8266-voi-wifimanager/>
9. <http://iot47.com/iot-bai-5-tao-webserver-voi-esp8266-va-lap-trinh-cho-esp8266-bang-arduino-ide/>
10. <https://arduino.vn/dieu-khien-led-bang-webserver-su-dung-nodemcu-esp8266/>
11. <https://esp8266.vn/nonos-sdk/http-server/http-server/>

PHỤ LỤC

Phần này bao gồm những nội dung cần thiết nhằm minh họa hoặc hỗ trợ cho nội dung luận văn như số liệu, biểu mẫu, tranh ảnh. . . . nếu sử dụng những câu trả lời cho một *bảng câu hỏi* thì *bảng câu hỏi mẫu* này phải được đưa vào phần Phụ lục ở dạng *nguyên bản* đã dùng để điều tra, thăm dò ý kiến; **không được tóm tắt hoặc sửa đổi**. Các tính toán mẫu trình bày tóm tắt trong các biểu mẫu cũng cần nêu trong Phụ lục của luận văn. Phụ lục không được dày hơn phần chính của luận văn.