

GROUP 9

LOGIC

DESIGN PROJECT

DECEMBER, 2024



# DANH SÁCH THÀNH VIÊN

Phan Huy Trung

2213709

Nguyễn Kim Thuận

2213357

Lê Nguyễn Phúc Thịnh

2213279

Đề tài

Xây dựng hệ thống giao tiếp, điều  
khiển module RFID, LED RGB và  
WiFi ESP32 sử dụng vi điều khiển  
STM32





## Giới thiệu

1

Tìm hiểu thực trạng và đưa ra các giải pháp

## Giải pháp

2

Trình bày giải pháp và hướng đi

## Thiết kế và hiện thực

3

Giai đoạn hiện thực và kết quả thực hiện

## Demo sản phẩm

4

Video mô phỏng code và đánh giá

## Tổng kết

5

Đưa ra phương hướng phát triển dựa trên đánh giá những điểm tốt và điều còn chưa làm được



# Giới thiệu

1

# THỰC TRẠNG

- Lĩnh vực điện tử, Internet vạn vật (IoT) và trí tuệ nhân tạo (AI) phát triển vượt bậc
- Nhu cầu bảo mật người dùng tăng cao

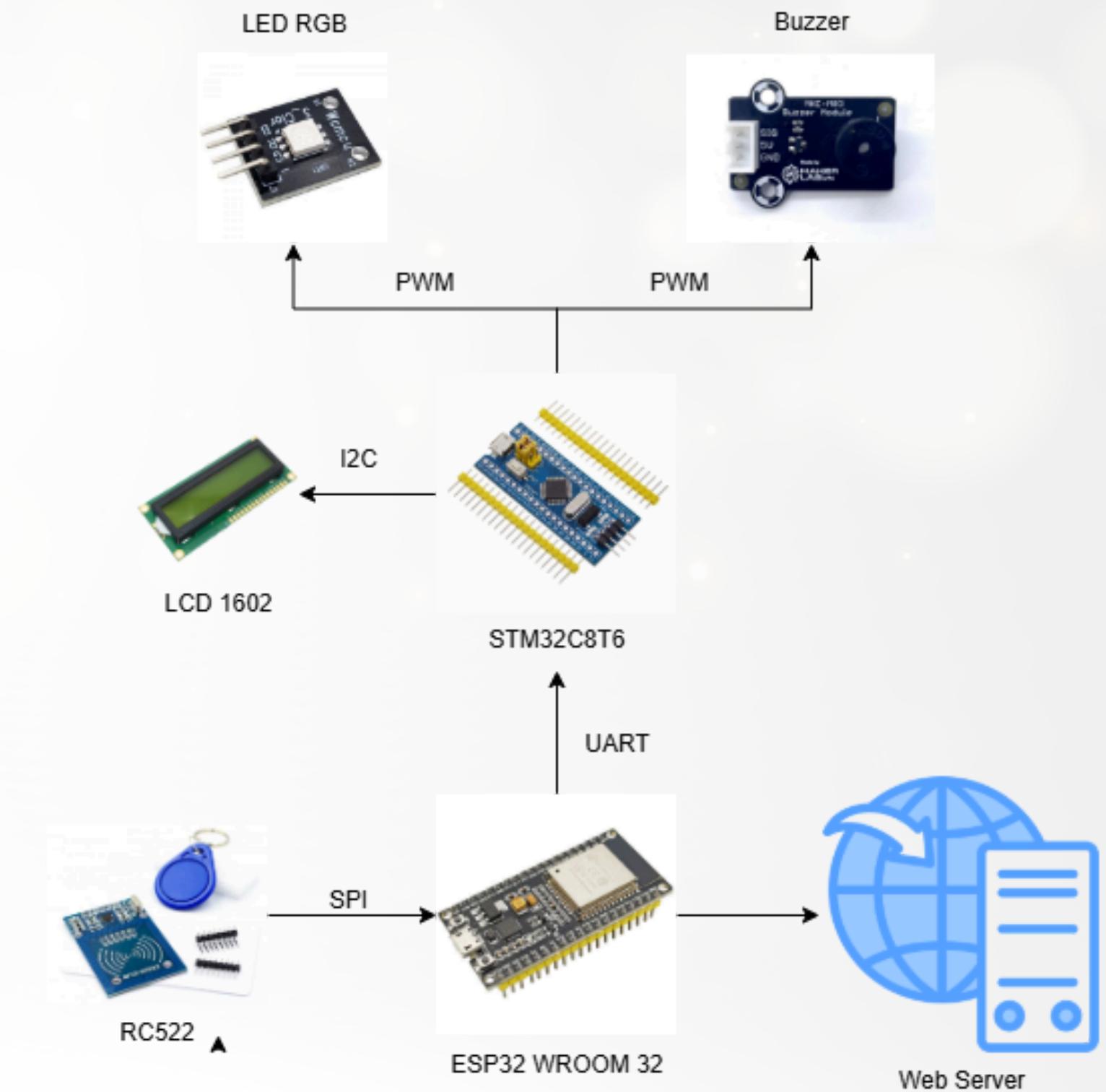
→ Ứng dụng các thiết bị điện tử thông minh đã trở thành xu thế tất yếu



# GIỚI THIỆU

Hệ thống bao gồm các module RFID RC522, ESP32, và STM32 là một giải pháp tiêu biểu nhằm đáp ứng những yêu cầu trên. Với khả năng quét thẻ RFID để nhận diện đối tượng, truyền thông tin đến ESP32, đèn LED RGB để báo hiệu tình trạng quét thẻ, và cuối cùng gửi dữ liệu lên Web Server.

Module bổ sung như màn hình LCD 16x2, và loa buzzer nhằm cải thiện và tối ưu hóa trải nghiệm người dùng.





# Giải pháp

2

# GIẢI PHÁP

## PHẦN MỀM

STM32CubeIDE là môi trường phát triển tích hợp (IDE) mạnh mẽ của STMicroelectronics, cung cấp các công cụ lập trình, biên dịch, và debug trực tiếp trên một nền tảng duy nhất



Arduino IDE là môi trường lập trình đơn giản, phù hợp cả với người mới bắt đầu và chuyên gia. Giao diện thân thiện và cú pháp lập trình đơn giản giúp tiết kiệm thời gian học tập và phát triển.

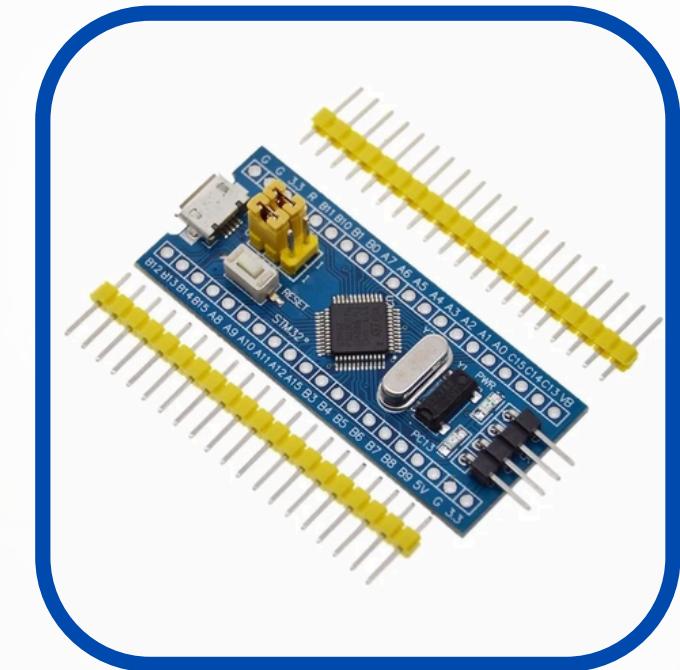


# GIẢI PHÁP

## PHẦN CỨNG



**RFID RC522**  
Quét và xác nhận  
danh tính của người  
dùng hoặc đối  
tượng.



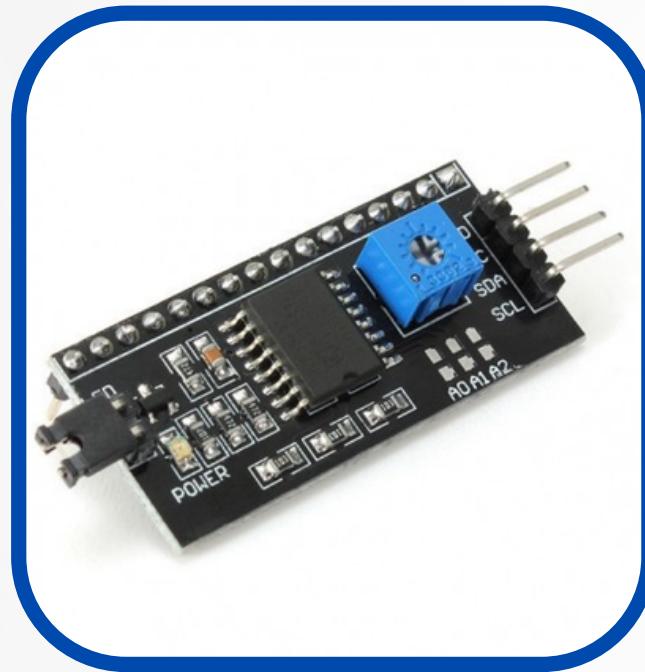
**STM32F103C8T6**  
Là cốt lõi xử lý dữ  
liệu và điều phối hoạt  
động của toàn bộ hệ  
thống.



**WIFI ESP32**  
Truyền dữ liệu lên Web  
Server, cho phép quản lý  
từ xa và tích hợp với  
các hệ thống IoT.

# GIẢI PHÁP

## PHẦN CỨNG



**I2C-Converter**

Hỗ trợ truyền tín hiệu cho LCD.

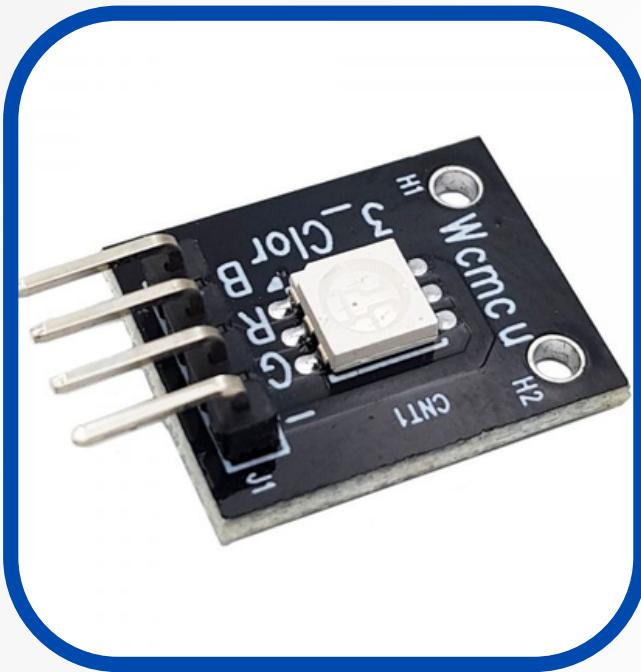


**LCD 16X2**

Hiện thông tin người dùng.

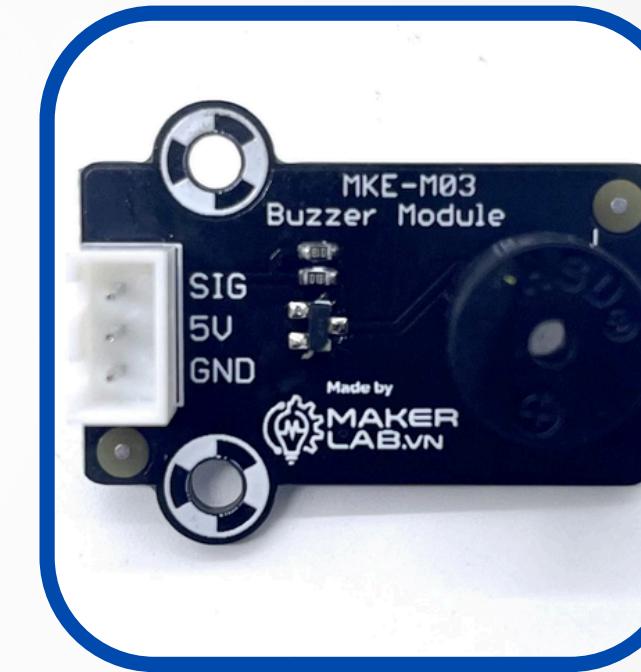
# GIẢI PHÁP

## PHẦN CỨNG



### LED\_RGB

Phát ra tín hiệu  
quét thẻ thành công  
hay chưa.

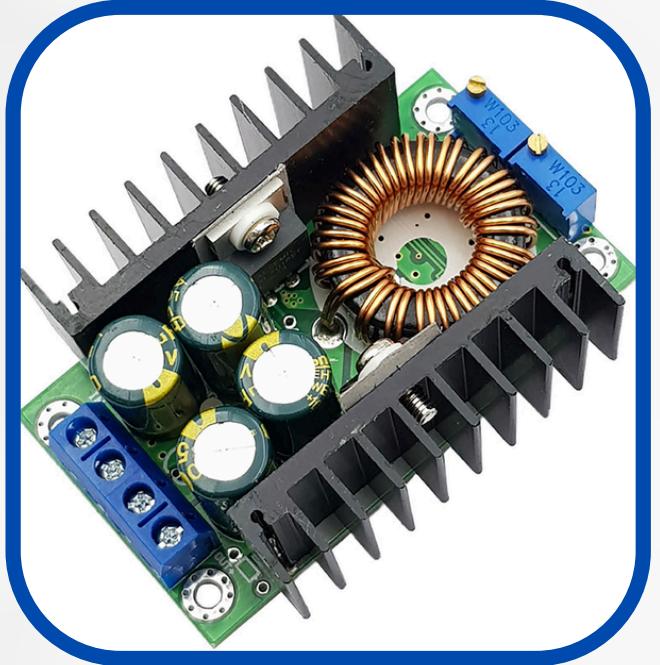


### Buzzer

Phát ra âm thanh nếu  
quét thẻ thành công  
và âm khác khi quét  
sai.

# GIẢI PHÁP

## BỘ NGUỒN



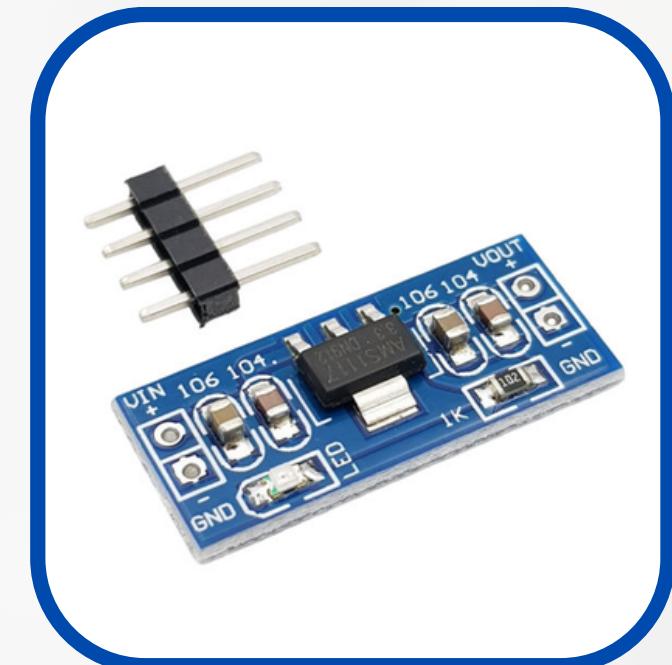
### Mạch giảm áp

Chuyển dòng điện xoay chiều 220V sang dòng một chiều 3,3V để cấp nguồn cho hệ thống.



### Nguồn tổ ong 12V-5A

Cấp nguồn điện cho hệ thống.



### Mạch giảm áp

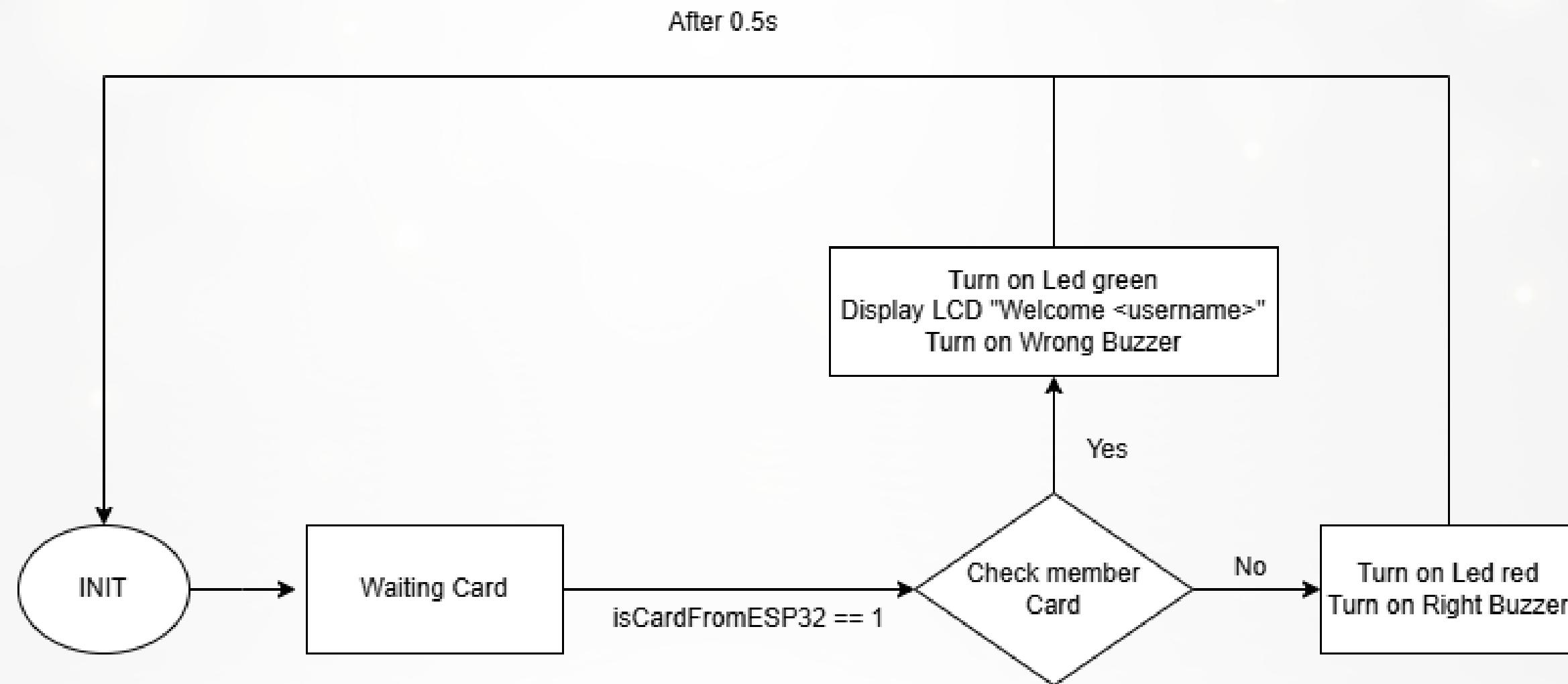
Giảm áp từ 5V xuống 3,3V để cấp cho STM32 và RC522



# Thiết kế và hiện thực

3

# THIẾT KẾ VÀ HIỆN THỰC



## Máy trạng thái



# THIẾT KẾ VÀ HIỆN THỰC

## Hiện thực với Scheduler

Kiến trúc Scheduler sẽ được sử dụng cho STM32, một kiến trúc giúp tổ chức và quản lý các tác vụ rất hiệu quả tích hợp thêm cấu trúc dữ liệu Linked List để giảm độ phức tạp thuật toán.

```
1 typedef struct TaskNode
2 {
3     void (*pTask)(void);
4     uint32_t Delay;
5     uint32_t Period;
6     struct TaskNode* next; //Next pointer
7 } TaskNode;
8
9 typedef struct
10 {
11     TaskNode* head; //Head Node
12 } LinkedList;
13
14 LinkedList SCH_LL = {
15     .head = NULL
16 }; //Init Scheduler Linked List
17
18 void SCH_Init(void); //Init Sheduler
19
20 void SCH_Add_Task(
21     void (*pFunction)(void), //Task
22     uint32_t DELAY, //Task's Delay
23     uint32_t PERIOD //Task's Period
24 ); //Add Task Node to Scheduler Linked List
25
26 void SCH_Update(void); //Update Delay of Task
27
28 void SCH_Dispatch_Tasks(void); //Perform the task when it's time !
29
30 void SCH_Delete_Task(void); //Delete the task after performing it
```

Scheduler áp dụng linked list

Độ phức tạp khi sử dụng Linked List → O(1)



# THIẾT KẾ VÀ HIỆN THỰC

## Database ESP32

### Front-end

Khởi động giao diện dùng các ngôn ngữ  
là: HTML, CSS và JavaScript.

```
1 // Route for serving the main page
2 server.on("/", HTTP_GET, []() {
3     serveFile("/index.html", "text/html");
4 });
5
6 // Route for serving the CSS file
7 server.on("/styles.css", HTTP_GET, []() {
8     serveFile("/styles.css", "text/css");
9 });
10
11 // Route for serving the JavaScript file
12 server.on("/scripts.js", HTTP_GET, []() {
13     serveFile("/scripts.js", "application/javascript");
14 });
15
16 server.on("/images/logo.png", HTTP_GET, []() {
17     serveFile("/images/logo.png", "image/png");
18 });
19
```



# THIẾT KẾ VÀ HIỆN THỰC

## Database ESP32

### Back-end

Khai báo mảng array dùng để lưu trữ các tên người dùng, mảng array lưu trữ các ID card, một struct CardInfo chứa thông tin thẻ, tên người dùng cuối cùng là một mảng chứa lịch sử của các thẻ.

```
1 String memberDatabase[] = {"51293B2", "34211E74", "849F2D74"};
2 String memberNames[] = {
3     "Lê Nguyễn Phúc Thịnh",
4     "Nguyễn Kim Thuận",
5     "Phan Huy Trung"
6 };
7
8 struct CardInfo {
9     String cardID;
10    String name;
11    String date;
12    String time;
13    bool isMember;
14 };
15 CardInfo scanHistory[10];
```

Database của hệ thống



# THIẾT KẾ VÀ HIỆN THỰC

## Database ESP32

### Back-end

Hiện thực hàm fetchTable() trong mã nguồn JavaScript. Hàm này có nhiệm vụ khi có tín hiệu thẻ từ module RFID - RC522, đưa thông tin thẻ lên Web Server.

```
1  async function fetchTable() {
2    try {
3      const response = await fetch("/table");
4      const tableContent = await response.text();
5      document.getElementById("tableBody").innerHTML = tableContent;
6    } catch (error) {
7      console.error("Error fetching table data:", error);
8    }
9  }
10 setInterval(fetchTable, 2000);
11 fetchTable();
```

Hàm đưa thông tin lên webServer



# THIẾT KẾ VÀ HIỆN THỰC

The screenshot shows a web browser window titled "ESP32 Web Server" with the URL "172.20.10.12". The page features a blue header with the logo of "ĐẠI HỌC QUỐC GIA TP.HCM TRƯỜNG ĐẠI HỌC BÁCH KHOA" and an envelope icon. Below the header, the title "RFID Card Management" is displayed, followed by "Card Reading History". A table lists four card readings:

No.	ID Card	Name	Date	Time	Is Member?
1	51293B2	Lê Nguyễn Phúc Thịnh	2024-12-18	17:50:39	Yes
2	34211E74	Nguyễn Kim Thuận	2024-12-18	17:50:42	Yes
3	849F2D74	Phan Huy Trung	2024-12-18	17:50:44	Yes
4	954E1812	Unknown	2024-12-18	17:51:03	No

The footer contains copyright information "© Logic Design Project Group 9. All Rights Reserved." and a weather widget showing "30°C Mostly cloudy". The bottom right corner shows the date and time "5:51 PM 12/18/2024".

## Giao diện của Web Server

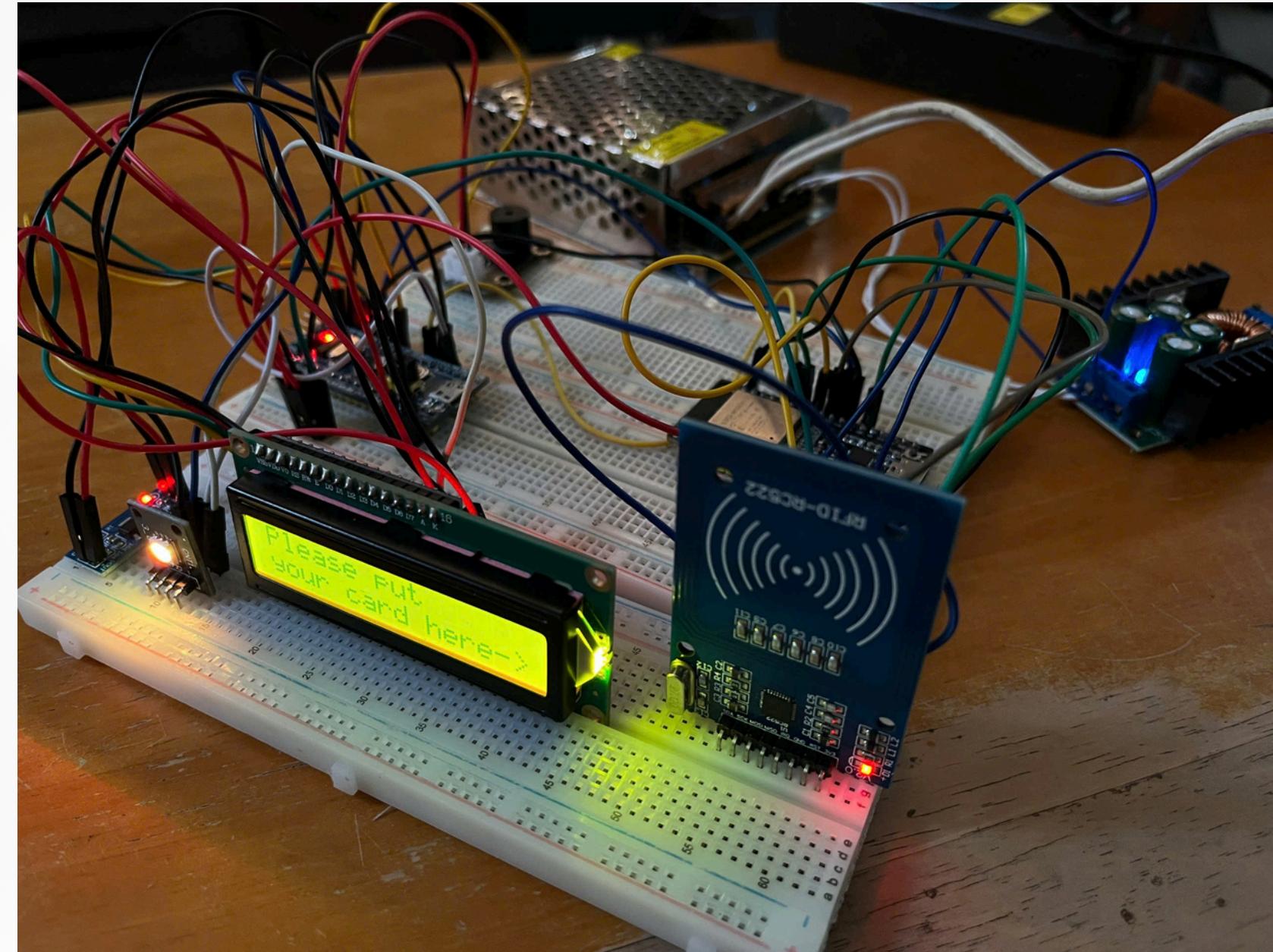




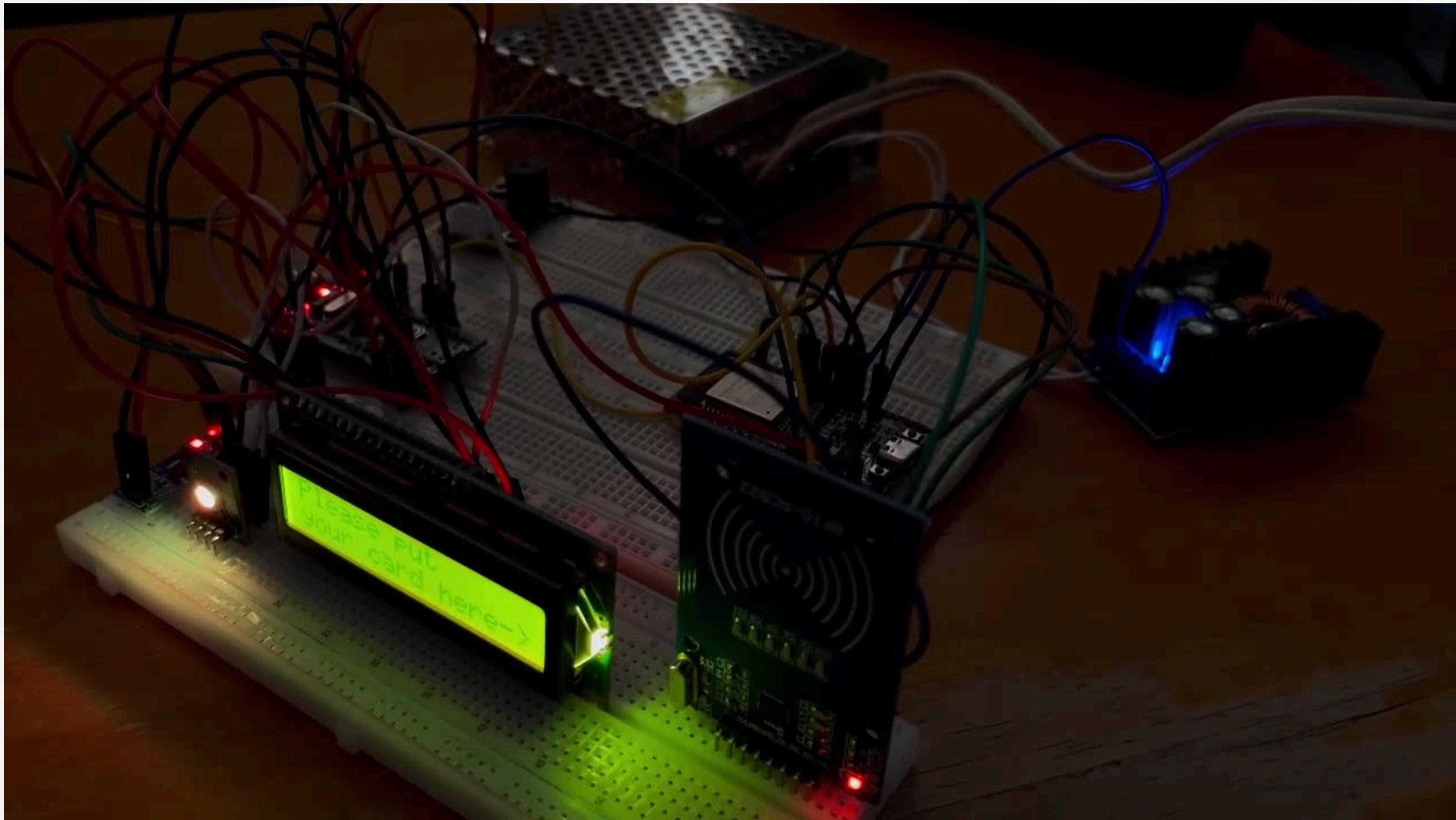
# Demo sản phẩm

4

# DEMO SẢN PHẨM



Hệ thống phần cứng





Tổng kết

5

# ƯU ĐIỂM

- Tính linh hoạt
- Khả năng kết nối mạng tốt
- Phản hồi nhanh và trực quan
- Chi phí thấp
- Tính mở rộng cao

# NHƯỢC ĐIỂM

- Giới hạn băng thông
- Khả năng bảo mật kém
- Khả năng chống nhiễu
- Hiển thị hạn chế
- Hệ thống database thô sơ



# PHƯƠNG HƯỚNG PHÁT TRIỂN HỆ THỐNG

Tăng cường bảo mật

Cải thiện tốc độ và hiệu suất

Nâng cấp giao diện người dùng

Mở rộng tính năng

Tăng tính thực tế và ứng dụng

Nâng cấp và cải thiện hệ thống Database

Đảm bảo khả năng phát triển dài hạn, phù hợp  
với các yêu cầu ngày càng cao của ứng dụng  
thực tế trong môi trường công nghiệp.



GROUP 9

THANK YOU

DECEMBER, 2024

