

# FA 2024 IOT102 Attendance check

1<sup>st</sup>Phung Huu Thanh,

2<sup>nd</sup>Le Van Huy Hoang,

3<sup>rd</sup>Le Nhat Quang,

4<sup>th</sup>Phan Thanh Long Chau,

and Duc Ngoc Minh Dang.

FPT University, Ho Chi Minh Campus, Vietnam

{thanhphse170345,hoanglvhse170333,quanglnse170415,chaultpse172855}@fpt.edu.vn, and ducdnm2@fe.edu.vn

**Abstract**—The Attendance Check System utilizing WiFi-enabled ESP8266. This project combines hardware and cloud technology, featuring an ESP8266 microcontroller for WiFi connectivity. Through seamless integration, attendance data is efficiently collected and managed, offering real-time accessibility and analytics.

Users can easily monitor attendance records through a user-friendly interface, enabling swift and accurate tracking. The utilization of Firebase ensures secure and scalable data storage, facilitating effortless retrieval and analysis. Future enhancements for this project may involve incorporating machine learning algorithms for predictive attendance patterns, implementing additional security measures, and exploring options for integration with other educational systems. The Attendance Check System stands as a testament to the intersection of hardware, cloud computing, and educational innovation.

## I. INTRODUCTION

In an era marked by rapid technological advancements, attendance tracking undergoes a paradigm shift through innovative solutions. The Attendance Check System, utilizing the ESP8266 microcontroller, LCD display, and AS608 fingerprint sensor, emerges as a response to the demand for efficient and secure attendance management.

The ESP8266, serving as the system's central intelligence, facilitates seamless data communication and processing. Paired with the informative LCD display and the AS608 fingerprint sensor, this project aims to transform how attendance is monitored in various environments.

By harnessing cutting-edge technology and incorporating user-friendly features, such as a mode-switching button, the Attendance Check System seeks to redefine attendance tracking. This project not only addresses immediate challenges but also provides a foundation for future advancements in attendance management technology.

In the subsequent sections, we will explore the key components of the system, detailing the roles of the ESP8266 microcontroller, LCD display, AS608 fingerprint sensor, and other integral elements that contribute to the project's innovative approach to attendance tracking.

## II. MAIN PROPOSAL

### A. Objective

The primary goal of this project is to create a comprehensive Attendance Check System that seamlessly integrates cutting-

edge technologies to enhance accuracy, efficiency, and security in attendance tracking.

### B. Components

1) *ESP8266 Microcontroller*: The ESP8266 serves as the central processing unit, providing reliable WiFi connectivity for real-time data transmission and access. Its versatility and robust performance make it an ideal choice for seamless integration into the attendance management system.

2) *LCD Display*: An informative LCD display is incorporated to present essential information in a clear and user-friendly manner. This component acts as an interface for users to view attendance data, system status, and relevant prompts, contributing to a streamlined and efficient user experience.

3) *AS608 Fingerprint Sensor*: The AS608 fingerprint sensor adds an additional layer of security and personalization to the system. By utilizing fingerprint recognition technology, the system ensures accurate attendance verification, mitigating the risks associated with traditional methods such as card swiping or manual entry.

4) *Wiring Connections*: The inclusion of accessible wiring connections simplifies the system setup, making it user-friendly and easy to install. This feature enhances the system's adaptability to various environments and user skill levels.

5) *Mode-Switching Button*: A mode-switching button is integrated to facilitate a smooth transition between fingerprint registration, verification, checking and deletion modes. This dynamic feature enhances the system's flexibility, allowing users to effortlessly navigate through different functionalities.

### C. Benefits

- **Security**: Fingerprint recognition provides a secure and accurate method of attendance verification.
- **User-Friendly**: The LCD display and mode-switching button ensure a straightforward and intuitive user experience.
- **Adaptability**: Accessible wiring connections enhance the system's adaptability to diverse environments.

### D. Future Enhancements

Consideration for future developments includes potential integration with cloud-based storage for data backup and accessibility from multiple locations, as well as exploring

options for mobile app connectivity to offer users greater flexibility in attendance monitoring.

### III. RESULTS AND DISCUSSION

Initially, we prepared the necessary equipment for the Attendance check project from various sources, including lecturers and electronic marketplaces. We listed the main functions, then drew circuit diagrams, and subsequently wrote code to implement the project. We learned wiring techniques and circuit assembly by referencing reputable internet sources in Vietnam and internationally, including Coursera. We commenced project implementation in the second week and completed all tasks by the eighth week. The resulting product is an attendance checking model authenticated by fingerprint.

The outcomes demonstrate the principles of WiFi operation and biometric fingerprint functioning. We learned how to use code to connect devices and display information on the screen for users. In this project, we developed a fingerprint attendance model by adding, deleting, and checking users via the AS608 fingerprint sensor, saving data into a database through the ESP8266 WiFi connection, and displaying information on an LCD screen.

The resulting model is a simple mini model for fingerprint authentication of users. It is highly beneficial for companies seeking to authenticate members across different departments and high-security residential buildings. Enhanced security measures are crucial for larger projects as this data is essential for user identification.

### IV. CONCLUSION

Our project successfully demonstrated the feasibility of using ESP8266, AS608 fingerprint sensor. Using biometric fingerprint security in projects demonstrates significant convenience, accuracy, and security in daily life. It efficiently manages users in a precise manner, saving time for individuals or businesses. Security is a crucial concern in modern life, and fingerprint biometrics in projects clearly showcase this aspect. Enterprises can adopt and customize this model to suit their work environment and legal requirements in their respective jurisdictions.

#### CODE:

```
1 // pin #2 is IN from sensor
2 // pin #3 is OUT from arduino
3 #include <Adafruit_Fingerprint.h>
4 #include <SoftwareSerial.h>
5 #include <LiquidCrystal.h>
6 #include <Arduino.h>
7 #if defined(ESP32) ||
    defined(ARDUINO_RASPBERRY_PI_PICO_W)
8 #include <WiFi.h>
9 #elif defined(ESP8266)
10 #include <ESP8266WiFi.h>
11 #endif
12
13 #include <ESPAsyncWebServer.h>
14 #include <ESP8266mDNS.h>
```

```
15 MDNSResponder mdns;
16 AsyncWebServer server(80);
17
18 // #include <Firebase_ESP_Client.h>
19 // Provide the token generation process
20 // info.
21 // #include <addons/TokenHelper.h>
22 // Provide the RTDB payload printing
23 // info and other helper functions.
24 // #include <addons/RTDBHelper.h>
25
26 /* 1. Define the WiFi credentials */
27 #define WIFI_SSID "FPT_Student"
28 #define WIFI_PASSWORD "12345678"
29
30 // For the following credentials, see
31 // examples/Authentications/SignInAsUser/EmailPass
32
33 /* 2. Define the API Key */
34 #define API_KEY
    "AIzaSyAmSco6lJUfrpfU0Rq6LBh-DLkyunHKMjQ"
35
36 /* 3. Define the RTDB URL */
37 #define DATABASE_URL
    "https://fingerprint-53a7d-default-rtdb.firebaseio.com or
    //<databaseName>.firebaseio.com or
    <databaseName>.<region>.firebase.database.app"
38
39 /* 4. Define the user Email and
40 // password that already registered or
41 // added in your project */
42 #define USER_EMAIL
    "phuuthanh2003@gmail.com"
43 #define USER_PASSWORD "thanhthanh1"
44
45 // Define Firebase Data object
46 SoftwareSerial mySerial(5, 16);
47 Adafruit_Fingerprint finger =
    Adafruit_Fingerprint(&mySerial);
48 // FirebaseData fbdo;
49 // FirebaseAuth auth;
50 // FirebaseConfig config;
51
52 unsigned long sendDataPrevMillis = 0;
53
54 const char *input_parameter1 =
    "fingerid";
55 String inputString = "";
56 bool stringComplete = false; // whether
    the string is complete
57 int functionState = 0;
58 uint8_t id;
```

```

58 uint8_t checkTask = -1; //for check task
59 int previousButtonState = 0;
60 const int RS = D2, EN = D3, d4 = D5, d5
    = D6, d6 = D7, d7 = D8;
61 int numberFinger = 0;
62 LiquidCrystal lcd(RS, EN, d4, d5, d6,
    d7);
63 String input_message;
64
65 String enrollPage = {
66     "<!DOCTYPE html>"
67     "<html>"
68     "<head>"
69     "    <meta http-equiv='Content-Type'
        content='text/html; charset=utf-8'>"
70     "    <title> ng k v n
        tay</title>"
71     "    <meta name='viewport'
        content='width=device-width,
        initial-scale=1'>"
72     "    <style>"
73     "        .b{width: 120px;"
74     "            height: 40px;"
75     "            font-size: 21px;"
76     "            font-weight: bold;"
77     "            color: #fff;"
78     "            background-color:#4caf50;"
79     "            border-radius: 15px;"
80     "            border: none;"
81     "            padding: 0 10px;"
82     "            transition: 0,2s;"
83     "            cursor: pointer;"
84     "        }"
85     "        .b:hover {"
86     "            scale: 0.95;"
87     "            opacity: 0.9;"
88     "        }"
89     "        .t{width: 100px;height:
        40px;font-size: 21px;color:
        #FFF;background-color:#f44336;border-r
        10px;}"
90     "        .a {"
91     "            margin-top: 40px;"
92     "            display: inline-block;"
93     "            text-align: center;"
94     "            width: 150px;"
95     "            height: 30px;"
96     "            line-height: 30px;"
97     "            font-size: 21px;"
98     "            font-weight: bold;"
99     "            background-color: #6495ED;"
100    "            text-decoration: none;"
101    "            color: white;"
102    "            border-radius: 10px;"
103    "            padding: 10px;"
104    "            transition: 0.5s;"
105    "        }"
106    ""
107    "        .a:hover {"
108    "            scale: 0.95;"
109    "            opacity: 0.9;"
110    "            color: #FFBF00;"
111    "        }"
112    "</style>"
113    "</head>"
114    "<body>"
115    "<div style='height: auto; text-align:
        center; margin-top: 70px'>"
116    "    <h1 align='center'>Please type in
        the ID # (from 1 to 127) you want
        to save this finger as...</h1>"
117    "    <form action='/enrollFG'>"
118    "        <input style='height: 40px;
        width: 100px;' name='fingerprint'
        type='number' value='' min='1'
        max='127' />"
119    "        <button type='submit'
        class='b'> ng k </button>"
120    "    </form>"
121    "    <a class='a' href='\"/\">Back to
        home</a>"
122    "</div>"
123    "</body>"
124    "</html>"
125    };
126
127 String checkId[127];
128 String attend() {
129     Serial.println(checkId[1]);
130     String attendHTML = "";
131     for (int i = 1; i < 127; i++) {
132         if (checkId[i] == "im
            danh") {
133             attendHTML += "<p class='b'>User "
                + String(i) + ": " + checkId[i]
                + "</p>";
134         } else if (checkId[i] == "
            x a") {
135             continue;
136         } else if (checkId[i] == "Ch a
            im danh") {
137             attendHTML += "<p class='t'>User "
                + String(i) + ": " + checkId[i]
                + "</p>";
138         }
139     }
140 }
141 String attend = {
142     "<!DOCTYPE html>"
143     "<html>"
144     "<head>"
145     "    <meta http-equiv='Content-Type'

```

```

146     content='text/html;
147     charset=utf-8'>"
148 " <title>X c minh v n tay</title>"
149 " <meta name='viewport'
150     content='width=device-width,
151     initial-scale=1'>"
152 " <style> "
153 " h1 {"
154 "     font-size: 30px;"
155 "     font-weight: bold;"
156 "     text-transform: capitalize;"
157 " }"
158 " .b {"
159 "     text-align: center;"
160 "     max-width: 300px;"
161 "     height: 40px;"
162 "     font-size: 21px;"
163 "     font-weight: bold;"
164 "     line-height: 40px;"
165 "     color: #fff;"
166 "     margin-right:10px;"
167 "     background-color: #4caf50;"
168 "     border-radius: 15px;"
169 "     padding: 10px 5px;"
170 "     transition: 0.2s;"
171 "     cursor: pointer;"
172 " }"
173 " .b:hover {"
174 "     scale: 1.05;"
175 "     opacity: 0.9;"
176 "     color: #DBFF33;"
177 " }"
178 " .t {"
179 "     max-width: 300px;"
180 "     height: 40px;"
181 "     font-weight: bold;"
182 "     font-size: 21px;"
183 "     line-height: 40px;"
184 "     color: #fff;"
185 "     margin-right:10px;"
186 "     padding: 10px 5px;"
187 "     background-color: #f44336;"
188 "     border-radius: 15px;"
189 " }"
190 " .a {"
191 "     display: inline-block;"
192 "     text-align: center;"
193 "     width: 150px;"
194 "     height: 30px;"
195 "     line-height: 30px;"
196 "     font-size: 21px;"
197 "     font-weight: bold;"
198 "     background-color: #6495ED;"
199 "     text-decoration: none;"
200 "     color: white;"
201 "     border-radius: 10px;"

```

```

198 "     padding: 10px;"
199 "     transition: 0.5s;"
200 " }"
201 " .a:hover {"
202 "     scale: 0.95;"
203 "     opacity: 0.9;"
204 "     color: #FFBF00;"
205 " }"
206 " </style>"
207 "</head>"
208 "<body>"
209 "<div style='height: auto;
210     text-align: center; margin-top:
211     70px'>"
212 "<h1 align='center'>X c minh v n
213     tay</h1>"
214 "<a class='a' href='\"/\">Back to
215     home</a>"
216 "<div style='display: flex;
217     flex-wrap: wrap'>"
218 + attendHTML + "</div>"
219 " </div>"
220 " </body>"
221 " </html>"
222 };
223 return attend;
224 }
225
226 String deleteFG = {
227     "<!DOCTYPE html>"
228     "<html>"
229     "<head>"
230     "    <meta http-equiv='Content-Type'
231         content='text/html; charset=utf-8'>"
232     "    <title>X a v n tay</title>"
233     "    <meta name='viewport'
234         content='width=device-width,
235         initial-scale=1'>"
236     "    <style>"
237     "        .b{width: 100px;"
238     "            height: 40px;"
239     "            font-size: 21px;"
240     "            font-weight: bold;"
241     "            color: #fff;"
242     "            background-color:#E74C3C;"
243     "            border-radius: 15px;"
244     "            border: none;"
245     "            transition: 0,2s;"
246     "            cursor: pointer;"
247     "        }"
248     "        .b:hover {"
249     "            scale: 0.95;"
250     "            opacity: 0.9;"
251     "        }"
252     "        .t{width: 100px;height:
253         40px;font-size: 21px;color:

```

```

#FFF;background-color:#f44336;border-radius:
10px;}"
245 "      .a {"
246 "        margin-top: 40px;"
247 "        display: inline-block;"
248 "        text-align: center;"
249 "        width: 150px;"
250 "        height: 30px;"
251 "        line-height: 30px;"
252 "        font-size: 21px;"
253 "        font-weight: bold;"
254 "        background-color: #6495ED;"
255 "        text-decoration: none;"
256 "        color: white;"
257 "        border-radius: 10px;"
258 "        padding: 10px;"
259 "        transition: 0.5s;"
260 "      }"
261 ""
262 "      .a:hover {"
263 "        scale: 0.95;"
264 "        opacity: 0.9;"
265 "        color: #FFBF00;"
266 "      }"
267 "    </style>"
268 "</head>"
269 "<body>"
270 "  <div style='height: auto;
  text-align: center; margin-top:
  70px'>"
271 "    <h1 align='center'>X a v n
  tay (id = 0      x a h t )</h1>"
272 "    <form action='/deleteFG'>"
273 "      <input style='height:
  40px; width: 100px;'
  name='fingerprint' type='number'
  value='' min='0' max='127' />"
274 "      <button type='submit'
  class='b'>X a</button>"
275 "    </form>"
276 "    <a class='a' href='\"/\">Back
  to home</a>"
277 "  </div>"
278 "</body>"
279 "</html>"
280
281 };
282
283 String checkExist = {
284 "  <!DOCTYPE html>"
285 "  <html>"
286 "  <head>"
287 "    <meta http-equiv='Content-Type'
  content='text/html; charset=utf-8'>"
288 "    <title> K i m t r a v n
  tay</title>"

```

```

"    <meta name='viewport'
  content='width=device-width,
  initial-scale=1'>"
290 "    <style>"
291 "      .b{width: 120px;"
292 "        height: 40px;"
293 "        font-size: 21px;"
294 "        font-weight: bold;"
295 "        color: #fff;"
296 "        background-color:#4caf50;"
297 "        border-radius: 15px;"
298 "        border: none;"
299 "        padding: 0 10px;"
300 "        transition: 0,2s;"
301 "        cursor: pointer;"
302 "      }"
303 "      .b:hover {"
304 "        scale: 0.95;"
305 "        opacity: 0.9;"
306 "      }"
307 "      .t{width: 100px;height:
  40px;font-size: 21px;color:
  #FFF;background-color:#f44336;border-radius:
  10px;}"
308 "      .a {"
309 "        margin-top: 40px;"
310 "        display: inline-block;"
311 "        text-align: center;"
312 "        width: 150px;"
313 "        height: 30px;"
314 "        line-height: 30px;"
315 "        font-size: 21px;"
316 "        font-weight: bold;"
317 "        background-color: #6495ED;"
318 "        text-decoration: none;"
319 "        color: white;"
320 "        border-radius: 10px;"
321 "        padding: 10px;"
322 "        transition: 0.5s;"
323 "      }"
324 ""
325 "      .a:hover {"
326 "        scale: 0.95;"
327 "        opacity: 0.9;"
328 "        color: #FFBF00;"
329 "      }"
330 "    </style>"
331 "</head>"
332 "<body>"
333 "  <div style='height: auto;
  text-align: center; margin-top:
  70px'>"
334 "    <h1 align='center'> K i m t r a
  v n tay</h1>"
335 "    <form action='/checkFG'>"
336 "      <input style='height:

```

337	40px; width: 100px;'		10px 5px;"
	name='fingerid' type='number'	348	"
	value='' min='1' max='127' />"		transition: 0.2s;"
338	" <button type='submit'	349	" cursor:
	class='b'> K i m tra</button>"		pointer;"
339	" </form>"	350	" }
	" <a class='a' href=\""/\">Back	351	" .b:hover {"
	to home</a>"	352	" scale:
340	" </div>"		1.05;"
341	"</body>"	353	" opacity:
342	"</html>"		0.9;"
343		354	" color:
344	};		#DBFF33;"
345		355	" }
346	String webPage = "<!DOCTYPE html>"	356	" .t {"
347	"<html>"		" width:
348	"<head>"	357	100px;"
349	" <meta		" height:
	http-equiv='Content-Typ	358	40px;"
	content='text/html;	359	" font-size:
	charset=utf-8'>"		21px;"
350	" <title> im	360	" color:
	danh</title>"		#fff;"
351	" <meta	361	" background-color:
	name='viewport'		#f44336;"
	content='width=device-w	362	"
	initial-scale=1'>"		border-radius: 10px;"
352	" <style>"	363	" }
353	" h1 {"		" .a {"
354	" font-size:	364	" display:
	30px;"		inline-block;"
355	"	365	"
	font-weight: bold;"	366	text-align: center;"
356	"		" max-width:
	text-transform:	367	300px;"
	capitalize;"		height:
357	" }	368	30px;"
358	" .b {"	369	"
359	"		line-height: 30px;"
	text-align: center;"	370	" font-size:
360	" width:		21px;"
361	110px;"	371	"
	height:		font-weight: bold;"
362	40px;"	372	"
	font-size:		background-color:
363	18px;"		#229954 ;"
364	" font-weight: bold;"	373	"
	color:		text-decoration:
365	#fff;"	374	none;"
	"		color:
	background-color:	375	white;"
366	#4caf50;"		border-radius: 10px;"
	border-radius: 15px;"	376	" padding:
367	" padding:		10px;"

```

397         "        transition: 0.5s;"
398         "        }"
399         "        .a:hover {"
400         "            scale:
401         "            0.95;"
402         "            opacity:
403         "            0.9;"
404         "            color:
405         "            #FFBF00;"
406         "        }"
407         "    </style>"
408     "</head>"
409     "<body>"
410     "<div style='height:
411         auto;margin-top:
412         70px'>"
413     "    <h1
414         style='text-align:
415         center'> iu
416         k h i n t h i t b
417         qua WIFI</h1>"
418     "    <table
419         align='center'>"
420     "        <tr>"
421     "            <td>"
422     "                <a
423                 class='a'
424                 href='/attend'> i m
425                 danh</a>"
426     "            <a
427                 class='a'
428                 href='/enroll'> n g
429                 k v n tay</a>"
430     "            <a
431                 class='a'
432                 href='/checkexist'> K
433                 tra v n tay</a>"
434     "            <a
435                 class='a'
436                 href='/delete'> X a
437                 v n tay</a>"
438     "        </td>"
439     "    </tr>"
440     "    </table>"
441     "</div>"
442     "</body>"
443     "</html>";
444
445 String rsCheck(int idOk) {
446     String rs = {
447         "<!DOCTYPE html>"
448         "<html>"
449         "<head>"
450         "    <meta http-equiv='Content-Type'
451             content='text/html;
452             charset=utf-8'>"
453         "    <title> K i m t r a v n
454             tay</title>"
455         "    <meta name='viewport'
456             content='width=device-width,
457             initial-scale=1'>"
458         "    <style>"
459         "        .b{width: 100px;height:
460             40px;font-size: 21px;color:
461             #FFF;background-color:#4caf50;border-radius:
462             10px;}"
463         "        .t{width: 100px;height:
464             40px;font-size: 21px;color:
465             #FFF;background-color:#f44336;border-radius:
466             10px;}"
467         "        .a{display: inline-block;
468             text-align: center;width:
469             150px;height: 30px;font-size:
470             21px;background-color:
471             blue;text-decoration: none;color:
472             white;border-radius: 10px;border:
473             1px solid black;}"
474         "    </style>"
475     "</head>"
476     "<body>"
477     "<div style='height: auto;
478         text-align: center; margin-top:
479         70px'>"
480     "<h1 align='center'>V n tay v i
481         id "
482     + String(idOk) + "            t n
483         t i </h1>"
484     "            <a class='a'
485                 href='/checkexist'>Back
486                 to check</a>"
487     "</div>"
488     "</body>"
489     "</html>"
490     };
491     return rs;
492 }
493
494 String rsCheckNot(int idNotOk) {
495     String rs = {
496         "<!DOCTYPE html>"
497         "<html>"
498         "<head>"
499         "    <meta http-equiv='Content-Type'
500             content='text/html;
501             charset=utf-8'>"
502         "    <title> K i m t r a v n
503             tay</title>"
504         "    <meta name='viewport'
505             content='width=device-width,
506             initial-scale=1'>"
507         "    <style>"

```

```

458     ".b{width: 100px;height:
459     40px;font-size: 21px;color:
        #FFF;background-color:#4caf50;border
        10px;}"
460     ".t{width: 100px;height:
461     40px;font-size: 21px;color:
        #FFF;background-color:#f44336;border
        10px;}"
462     ".a{display: inline-block;
        text-align: center;width:
        150px;height: 30px;font-size:
        21px;background-color:
        blue;text-decoration: none;color:
        white;border-radius: 10px;border:
        1px solid black;}"
463     "</style>"
464     "</head>"
465     "<body>"
466     "<div style='height: auto;
        text-align: center; margin-top:
        70px'>"
467     "<h1 align='center'>V n tay v i
        id "
468     + String(idNotOk) + " kh ng t n
        t i </h1>"
469
470         "<a class='a'
471         href='/checkxis
472         to check</a>"
473         "</div>"
474         "</body>"
475         "</html>"
476
477     };
478     return rs;
479 }
480
481 String xoaThanhCong(int id) {
482     String rs = "";
483     if (id != 0) {
484         rs = {
485             "<!DOCTYPE html>"
486             "<html>"
487             "<head>"
488             "<meta
489             http-equiv='Content-Type'
490             content='text/html;
491             charset=utf-8'>"
492             "<title> K i m t r a v n
493             tay</title>"
494             "<meta name='viewport'
495             content='width=device-width,
496             initial-scale=1'>"
497             "<style>"
498             ".a{display: inline-block;
499             text-align: center;width:
500             150px;height: 30px;font-size:
501             21px;background-color:
502             blue;text-decoration:
503             none;color:
504             white;border-radius:
505             10px;border: 1px solid black;}"
506             "</style>"
507             "</head>"
508             "<body>"
509             "<div style='height: auto;
510             text-align: center; margin-top:
511             70px'>"
512             "<h1 align='center'>X a h t
513             v n tay th nh c ng!</h1>"
514             "<a class='a' href='/delete'>Back
515             to delete</a>"
516             "</div>"
517             "</body>"
518             "</html>"
519
520             blue;text-decoration:
521             none;color:
522             white;border-radius:
523             10px;border: 1px solid black;}"
524             "</style>"
525             "</head>"
526             "<body>"
527             "<div style='height: auto;
528             text-align: center; margin-top:
529             70px'>"
530             "<h1 align='center'>X a th nh
531             c ng v n tay v i id "
532             + String(id) + "</h1>"
533             "<a class='a'
534             href='/delete'>Back
535             to delete</a>"
536             "</div>"
537             "</body>"
538             "</html>"
539
540             };
541     } else {
542         rs = {
543             "<!DOCTYPE html>"
544             "<html>"
545             "<head>"
546             "<meta
547             http-equiv='Content-Type'
548             content='text/html;
549             charset=utf-8'>"
550             "<title> K i m t r a v n
551             tay</title>"
552             "<meta name='viewport'
553             content='width=device-width,
554             initial-scale=1'>"
555             "<style>"
556             ".a{display: inline-block;
557             text-align: center;width:
558             150px;height: 30px;font-size:
559             21px;background-color:
560             blue;text-decoration:
561             none;color:
562             white;border-radius:
563             10px;border: 1px solid black;}"
564             "</style>"
565             "</head>"
566             "<body>"
567             "<div style='height: auto;
568             text-align: center; margin-top:
569             70px'>"
570             "<h1 align='center'>X a h t
571             v n tay th nh c ng!</h1>"
572             "<a class='a' href='/delete'>Back
573             to delete</a>"
574             "</div>"
575             "</body>"
576             "</html>"
577
578             blue;text-decoration:
579             none;color:
580             white;border-radius:
581             10px;border: 1px solid black;}"
582             "</style>"
583             "</head>"
584             "<body>"
585             "<div style='height: auto;
586             text-align: center; margin-top:
587             70px'>"
588             "<h1 align='center'>X a th nh
589             c ng v n tay v i id "
590             + String(id) + "</h1>"
591             "<a class='a'
592             href='/delete'>Back
593             to delete</a>"
594             "</div>"
595             "</body>"
596             "</html>"
597
598             };
599     }
600 }

```



```

517     };
518 }
519 return rs;
520 }
521
522 String daTonTai(String inputmsg) {
523     String rs = {
524         "<!DOCTYPE html>"
525         "<html>"
526         "<head>"
527         "    <meta
528             http-equiv='Content-Type'
529             content='text/html;
530             charset=utf-8'>"
531         "    <title> K i m   t r a   v   n
532             t a y</title>"
533         "    <meta name='viewport'
534             content='width=device-width,
535             initial-scale=1'>"
536         "    <style>"
537         "        .a{display: inline-block;
538             text-align: center; width:
539             150px; height: 30px; font-size:
540             21px; background-color:
541             blue; text-decoration:
542             none; color:
543             white; border-radius:
544             10px; border: 1px solid black;} "
545         "    </style>"
546         "</head>"
547         "<body>"
548         "<div style='height: auto;
549             text-align: center; margin-top:
550             70px'>"
551         "<h1 align='center'>V n   t a y   v i
552             i d           t n       t i   "
553         + inputmsg + "</h1>"
554         "        <a class='a'
555             href='/enroll'>Back
556             to enroll</a>"
557         "    </div>"
558         "    </body>"
559         "    </html>"
560     };
561     return rs;
562 }
563
564 void setup() {
565     // put your setup code here, to run
566     // once:
567     Serial.begin(9600);
568     WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
569     Serial.print("Connecting to Wi-Fi");
570     while (WiFi.status() != WL_CONNECTED) {
571         Serial.print(".");
572         delay(300);
573     }
574 }

```

```

575 }
576 Serial.println();
577 Serial.print("Connected with IP: ");
578 Serial.println(WiFi.localIP());
579 Serial.println();
580
581 if (mdns.begin("esp8266",
582     WiFi.localIP()))
583     Serial.println("MDNS responder
584         started");
585
586 server.on("/", HTTP_GET,
587     [] (AsyncWebServerRequest *request) {
588         checkTask = -1;
589         lcd.clear();
590         lcd.setCursor(0, 0);
591         lcd.print("Vui long chon");
592         lcd.setCursor(0, 1);
593         lcd.print("che do");
594         request->send(200, "text/html",
595             webPage);
596     });
597
598 server.on("/attend", HTTP_GET,
599     [] (AsyncWebServerRequest *request) {
600         checkTask = 0;
601         request->send(200, "text/html",
602             attend());
603     });
604
605 server.on("/enroll", HTTP_GET,
606     [] (AsyncWebServerRequest *request) {
607         lcd.clear();
608         lcd.setCursor(0, 0);
609         lcd.print("Nhap id de");
610         lcd.setCursor(0, 1);
611         lcd.print("dang ki");
612         request->send(200, "text/html",
613             enrollPage);
614     });
615
616 server.on("/checkexist", HTTP_GET,
617     [] (AsyncWebServerRequest *request) {
618         lcd.clear();
619         lcd.setCursor(0, 0);
620         lcd.print("Nhap id de check");
621         request->send(200, "text/html",
622             checkExist);
623     });
624
625 server.on("/delete", HTTP_GET,
626     [] (AsyncWebServerRequest *request) {
627         lcd.clear();
628         lcd.setCursor(0, 0);
629         lcd.print("Nhap id de xoa");
630         request->send(200, "text/html",
631             deletePage);
632     });
633 }

```

```

deleteFG);
599 });
600
601 server.on("/enrollFG", HTTP_GET,
602     [] (AsyncWebServerRequest *request) {
603         String input_parameter;
604         if
605             (request->hasParam(input_parameter1)
606             {
607                 input_message =
608                     request->getParam(input_parameter1);
609                 input_parameter = input_parameter1;
610                 if (checkId[input_message.toInt()]
611                     == "Ch a im danh" ||
612                     checkId[input_message.toInt()]
613                     == " im danh") {
614                     request->send(200, "text/html",
615                         daTonTai(input_message));
616                 }
617                 checkTask = 1;
618             } else {
619                 input_message = "No message sent";
620                 input_parameter = "none";
621             }
622             request->send(200, "text/html",
623                 enrollPage);
624         });
625
626 server.on("/checkFG", HTTP_GET,
627     [] (AsyncWebServerRequest *request) {
628         String input_parameter;
629         if
630             (request->hasParam(input_parameter1)
631             {
632                 input_message =
633                     request->getParam(input_parameter1);
634                 input_parameter = input_parameter1;
635             } else {
636                 input_message = "No message sent";
637                 input_parameter = "none";
638             }
639             int isOk = checkFinger();
640             if (isOk) {
641                 request->send(200, "text/html",
642                     rsCheck(id));
643             } else {
644                 request->send(200, "text/html",
645                     rsCheckNot(id));
646             }
647         });
648
649 server.on("/deleteFG", HTTP_GET,
650     [] (AsyncWebServerRequest *request) {
651         String input_parameter;
652         if
653             (request->hasParam(input_parameter1)
654             {
655                 input_message =
656                     request->getParam(input_parameter1)->value;
657                 input_parameter = input_parameter1;
658             } else {
659                 input_message = "No message sent";
660                 input_parameter = "none";
661             }
662             int parse_id = input_message.toInt();
663             deleteFingerPrintWithId(input_message,
664                 parse_id);
665             request->send(200, "text/html",
666                 xoaThanhCong(id));
667         });
668
669 server.begin();
670
671 lcd.begin(16, 2);
672 while (!Serial)
673     ; // For Yun/Leo/Micro/Zero/...
674 delay(100);
675 Serial.println("\n\nAdafruit
676     Fingerprint sensor enrollment");
677 // set the data rate for the sensor
678 serial port
679 finger.begin(57600);
680 if (finger.verifyPassword()) {
681     Serial.println("Found fingerprint
682         sensor!");
683 } else {
684     Serial.println("Did not find
685         fingerprint sensor :(");
686     while (1) { delay(1); }
687 }
688 lcd.print("Hello there!");
689 Serial.println(F("Reading sensor
690     parameters"));
691 finger.getParameters();
692 Serial.print(F("Status: 0x"));
693 Serial.println(finger.status_reg, HEX);
694 Serial.print(F("Sys ID: 0x"));
695 Serial.println(finger.system_id, HEX);
696 Serial.print(F("Capacity: "));
697 Serial.println(finger.capacity);
698 Serial.print(F("Security level: "));
699 Serial.println(finger.security_level);
700 Serial.print(F("Device address: "));
701 Serial.println(finger.device_addr,
702     HEX);
703 Serial.print(F("Packet len: "));
704 Serial.println(finger.packet_len);
705 Serial.print(F("Baud rate: "));
706 Serial.println(finger.baud_rate);
707 finger.getTemplateCount();
708 Serial.print(F("Sensor contains: "));
709 Serial.println(finger.templateCount);

```

```

683 Serial.println(" templates");
684 }
685
686 uint8_t stringToUint8(String str) {
687     int intValue = str.toInt();
688     // Convert the string to an integer
689     if (intValue < 0 || intValue > 255) {
690         // Check if the integer value is
691         // within the range of uint8_t
692         // Handle the case where the value
693         // is out of range
694         Serial.println("Invalid value. Must
695         be between 0 and 255.");
696         return 0; // Return 0 or handle
697         appropriately based on your
698         requirements
699     }
700     return static_cast<uint8_t>(intValue);
701     // Convert the integer to uint8_t
702 }
703
704 void loop() {
705     if (checkTask == -1) {
706         lcd.clear();
707         lcd.setCursor(0, 0);
708         lcd.print("Vui long chon");
709         lcd.setCursor(0, 1);
710         lcd.print("che do");
711     } else if (checkTask == 0) {
712         getFingerprintID();
713     } else if (checkTask == 1) //Add
714         template fingerprint
715     {
716         Serial.println("Ready to enroll a
717         fingerprint!");
718         Serial.println("Please type in the
719         ID # (from 1 to 127) you want to
720         save this finger as...");
721         lcd.clear();
722         lcd.setCursor(0, 0);
723         lcd.print("Nhap ID");
724         int parse_id = input_message.toInt();
725         id = stringToUint8(input_message);
726         Serial.println(input_message);
727         //
728         Serial.println(getUser(parse_id));
729         if (parse_id == 0) { // ID #0 not
730             allowed, try again!
731             lcd.clear();
732             lcd.setCursor(0, 0);
733             lcd.print("Khong hop le");
734             delay(1000);
735             lcd.clear();
736             lcd.setCursor(0, 0);
737             lcd.print("Thu lai");
738         }
739     }
740 }

```

```

725 Serial.print("Enrolling ID #");
726 Serial.println(id);
727 while
728     (!getFingerprintEnroll(parse_id))
729     {
730         checkTask = -1;
731     } else if (checkTask == 2) //check
732         template
733     {
734         checkFinger();
735     } else if (checkTask == 3) //delete
736         template
737     {
738         id = stringToUint8(input_message);
739         int parse_id = input_message.toInt();
740         if (id == 0) //delete all template
741         {
742             finger.emptyDatabase();
743             lcd.print("Empty database");
744             Serial.println("Now database is
745             empty");
746         } else {
747             deleteFingerprint(id, parse_id);
748             lcd.print("Delete user id " +
749             input_message);
750             Serial.println("Delete user id " +
751             input_message);
752         }
753         checkTask = -1;
754     }
755 }
756
757 bool checkFinger() {
758     bool isOk = false;
759     // Parse input_message to uint8_t
760     id = stringToUint8(input_message);
761     // Load fingerprint model by id
762     uint8_t p = finger.loadModel(id);
763     // Display result based on the loaded
764     model
765     Serial.print("ID ");
766     Serial.print(id);
767     if (p == FINGERPRINT_OK) {
768         isOk = true;
769         Serial.println(" is exist");
770     } else {
771         Serial.println(" is NOT exist");
772     }
773     // Reset checkTask flag
774     Serial.println(isOk);
775     return isOk;
776 }

```

```

773
774 void deleteFingerPrintWithId(String
      input_message, int parse_id) {
775     id = stringToUint8(input_message);
776     if (id == 0) //delete all template
777     {
778         finger.emptyDatabase();
779         for (int i = 1; i < 127; i++) {
780             checkId[i] = "    x a";
781         }
782         lcd.print("Empty database");
783         Serial.println("Now database is
              empty");
784     } else {
785         deleteFingerprint(id, parse_id);
786         lcd.print("Delete user id " +
              input_message);
787         Serial.println("Delete user id " +
              input_message);
788     }
789     checkTask = -1;
790 }
791
792 uint8_t getFingerprintID() {
793     uint8_t p = finger.getImage();
794     switch (p) {
795         case FINGERPRINT_OK:
796             Serial.println("Image taken");
797             break;
798         case FINGERPRINT_NOFINGER:
799             Serial.println("No finger
              detected");
800             lcd.clear();
801             lcd.setCursor(0, 0);
802             lcd.print("Diem danh");
803             delay(500);
804             return p;
805         case FINGERPRINT_PACKETRECIEVEERR:
806             Serial.println("Communication
              error");
807             return p;
808         case FINGERPRINT_IMAGEFAIL:
809             Serial.println("Imaging error");
810             return p;
811         default:
812             Serial.println("Unknown error");
813             return p;
814     }
815     // OK success!
816     p = finger.image2Tz();
817     switch (p) {
818         case FINGERPRINT_OK:
819             Serial.println("Image converted");
820             break;
821         case FINGERPRINT_IMAGEMESS:
822             Serial.println("Image too messy");

```

```

823         return p;
824     case FINGERPRINT_PACKETRECIEVEERR:
825         Serial.println("Communication
              error");
826         return p;
827     case FINGERPRINT_FEATUREFAIL:
828         Serial.println("Could not find
              fingerprint features");
829         return p;
830     case FINGERPRINT_INVALIDIMAGE:
831         Serial.println("Could not find
              fingerprint features");
832         return p;
833     default:
834         Serial.println("Unknown error");
835         return p;
836 }
837 // OK converted!
838 p = finger.fingerSearch();
839 if (p == FINGERPRINT_OK) {
840     Serial.println("Found a print
              match!");
841 } else if (p ==
      FINGERPRINT_PACKETRECIEVEERR) {
842     Serial.println("Communication
              error");
843     return p;
844 } else if (p == FINGERPRINT_NOTFOUND) {
845     Serial.println("Did not find a
              match");
846     lcd.clear();
847     lcd.setCursor(0, 0);
848     lcd.print("Thu lai");
849     delay(2000);
850     return p;
851 } else {
852     Serial.println("Unknown error");
853     return p;
854 }
855
856 // found a match!
857 Serial.print("Found ID #");
858 Serial.print(finger.fingerID);
859 Serial.print(" with confidence of ");
860 Serial.println(finger.confidence);
861 // updateUser(finger.fingerID, "
      im danh");
862 lcd.clear();
863 lcd.setCursor(0, 0);
864 lcd.print("Da diem danh");
865 delay(2000);
866 checkId[finger.fingerID] = "
      im danh";
867 return finger.fingerID;
868 }
869

```

```

870 uint8_t getFingerprintEnroll(int
      parse_id) {
871     int p = -1;
872     Serial.print("Waiting for valid finger
      to enroll as #");
873     lcd.clear();
874     lcd.setCursor(0, 0);
875     lcd.print("Dau van tay");
876     Serial.println(id);
877     while (p != FINGERPRINT_OK) {
878         p = finger.getImage();
879         switch (p) {
880             case FINGERPRINT_OK:
881                 Serial.println("Image taken");
882                 break;
883             case FINGERPRINT_NOFINGER:
884                 Serial.print(".");
885                 delay(500);
886                 break;
887             case FINGERPRINT_PACKETRECEIVEERR:
888                 Serial.println("Communication
      error");
889                 break;
890             case FINGERPRINT_IMAGEFAIL:
891                 Serial.println("Imaging error");
892                 break;
893             default:
894                 Serial.println("Unknown error");
895                 break;
896         }
897     }
898
899     // OK success!
900     p = finger.image2Tz(1);
901     switch (p) {
902         case FINGERPRINT_OK:
903             Serial.println("Image converted");
904             break;
905         case FINGERPRINT_IMAGEMESS:
906             Serial.println("Image too messy");
907             return p;
908         case FINGERPRINT_PACKETRECEIVEERR:
909             Serial.println("Communication
      error");
910             return p;
911         case FINGERPRINT_FEATUREFAIL:
912             Serial.println("Could not find
      fingerprint features");
913             return p;
914         case FINGERPRINT_INVALIDIMAGE:
915             Serial.println("Could not find
      fingerprint features");
916             return p;
917         default:
918             Serial.println("Unknown error");
919             return p;

```

```

920     }
921
922     Serial.println("Remove finger");
923     lcd.clear();
924     lcd.setCursor(0, 0);
925     lcd.print("Bo tay");
926     delay(2000);
927     p = 0;
928     while (p != FINGERPRINT_NOFINGER) {
929         p = finger.getImage();
930     }
931     Serial.print("ID ");
932     Serial.println(id);
933     p = -1;
934     Serial.println("Place same finger
      again");
935     lcd.clear();
936     lcd.setCursor(0, 0);
937     lcd.print("Xac nhan");
938     lcd.setCursor(0, 1);
939     lcd.print("Dau van tay");
940     while (p != FINGERPRINT_OK) {
941         p = finger.getImage();
942         switch (p) {
943             case FINGERPRINT_OK:
944                 Serial.println("Image taken");
945                 break;
946             case FINGERPRINT_NOFINGER:
947                 Serial.print(".");
948                 break;
949             case FINGERPRINT_PACKETRECEIVEERR:
950                 Serial.println("Communication
      error");
951                 break;
952             case FINGERPRINT_IMAGEFAIL:
953                 Serial.println("Imaging error");
954                 break;
955             default:
956                 Serial.println("Unknown error");
957                 break;
958         }
959     }
960
961     // OK success!
962     p = finger.image2Tz(2);
963     switch (p) {
964         case FINGERPRINT_OK:
965             Serial.println("Image converted");
966             break;
967         case FINGERPRINT_IMAGEMESS:
968             Serial.println("Image too messy");
969             return p;
970         case FINGERPRINT_PACKETRECEIVEERR:
971             Serial.println("Communication
      error");
972             return p;

```

```

973 case FINGERPRINT_FEATUREFAIL:
974     Serial.println("Could not find
          fingerprint features");
975     return p;
976 case FINGERPRINT_INVALIDIMAGE:
977     Serial.println("Could not find
          fingerprint features");
978     return p;
979 default:
980     Serial.println("Unknown error");
981     return p;
982 }
983
984 // OK converted!
985 Serial.print("Creating model for #");
986 Serial.println(id);
987 p = finger.createModel();
988 if (p == FINGERPRINT_OK) {
989     Serial.println("Prints matched!");
990 } else if (p ==
          FINGERPRINT_PACKETRECEIVEERR) {
991     Serial.println("Communication
          error");
992     return p;
993 } else if (p ==
          FINGERPRINT_ENROLLMISMATCH) {
994     Serial.println("Fingerprints did not
          match");
995     lcd.setCursor(0, 0);
996     lcd.print("Van tay");
997     lcd.setCursor(0, 1);
998     lcd.print("Khong khop");
999     return p;
1000 } else {
1001     Serial.println("Unknown error");
1002     return p;
1003 }
1004
1005 Serial.print("ID ");
1006 Serial.println(id);
1007 p = finger.storeModel(id);
1008 if (p == FINGERPRINT_OK) {
1009     Serial.println("Stored!");
1010     checkId[parse_id] = "Ch a im
          danh";
1011     // updateUser(id, "Ch a im
          danh");
1012     lcd.clear();
1013     lcd.setCursor(0, 0);
1014     lcd.print("Luu van tay");
1015     lcd.setCursor(0, 1);
1016     lcd.print("Thanh cong");
1017     delay(5000);

```

```

1018 } else if (p ==
          FINGERPRINT_PACKETRECEIVEERR) {
1019     Serial.println("Communication
          error");
1020     return p;
1021 } else if (p ==
          FINGERPRINT_BADLOCATION) {
1022     Serial.println("Could not store in
          that location");
1023     return p;
1024 } else if (p == FINGERPRINT_FLASHERR) {
1025     Serial.println("Error writing to
          flash");
1026     return p;
1027 } else {
1028     Serial.println("Unknown error");
1029     return p;
1030 }
1031 return true;
1032 }
1033
1034 uint8_t deleteFingerprint(uint8_t id,
          int parse_id) {
1035     uint8_t p = -1;
1036     p = finger.deleteModel(id);
1037     if (p == FINGERPRINT_OK) {
1038         // int parse_id =
          input_message.toInt();
1039         // updateUser(id, " x a");
1040         checkId[parse_id] = " x a";
1041         Serial.println("Deleted!");
1042     } else if (p ==
          FINGERPRINT_PACKETRECEIVEERR) {
1043         Serial.println("Communication
          error");
1044     } else if (p ==
          FINGERPRINT_BADLOCATION) {
1045         Serial.println("Could not delete in
          that location");
1046     } else if (p == FINGERPRINT_FLASHERR) {
1047         Serial.println("Error writing to
          flash");
1048     } else {
1049         Serial.print("Unknown error: 0x");
1050         Serial.println(p, HEX);
1051     }
1052     return p;
1053 }

```

Listing 1. Your Arduino Code

## REFERENCES

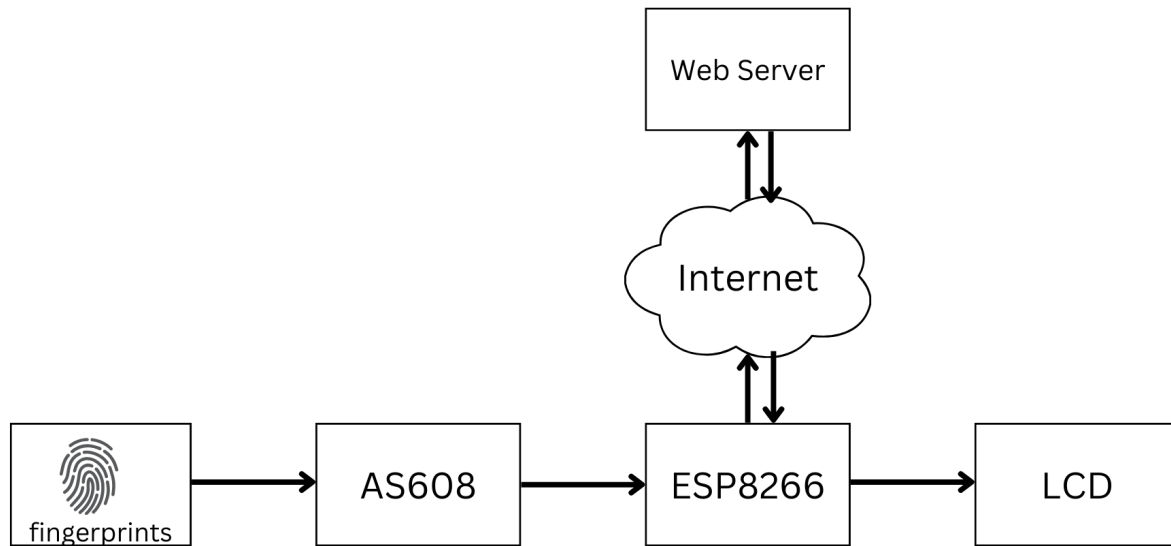


Figure 1. Block diagram of the developed system.

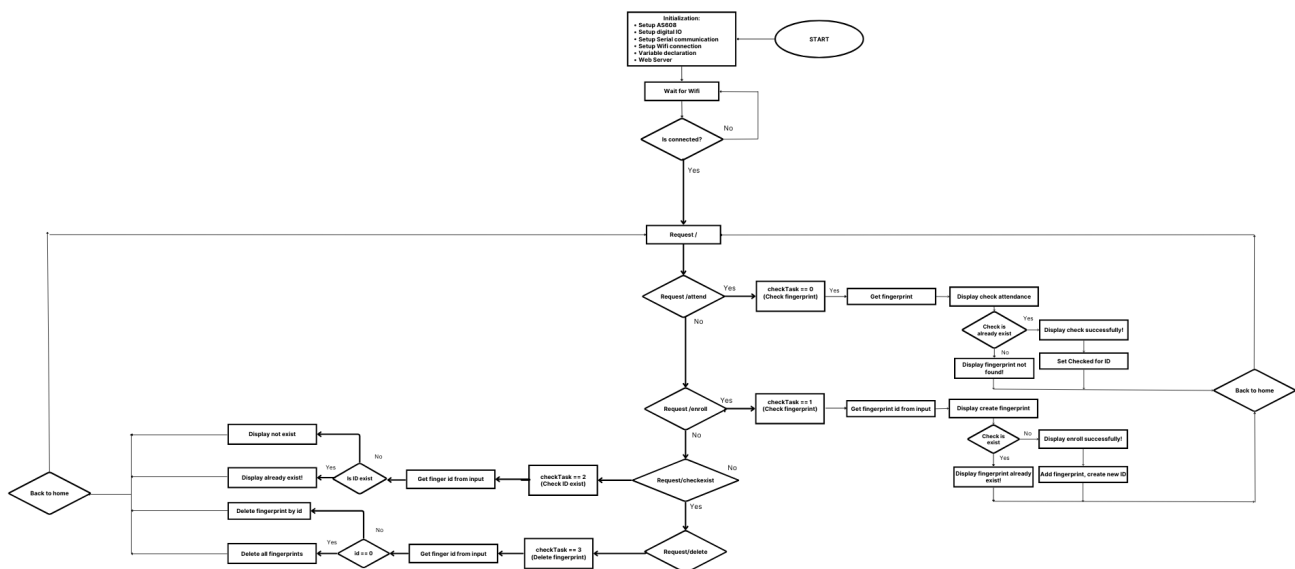


Figure 2. Flow chart of the developed system.



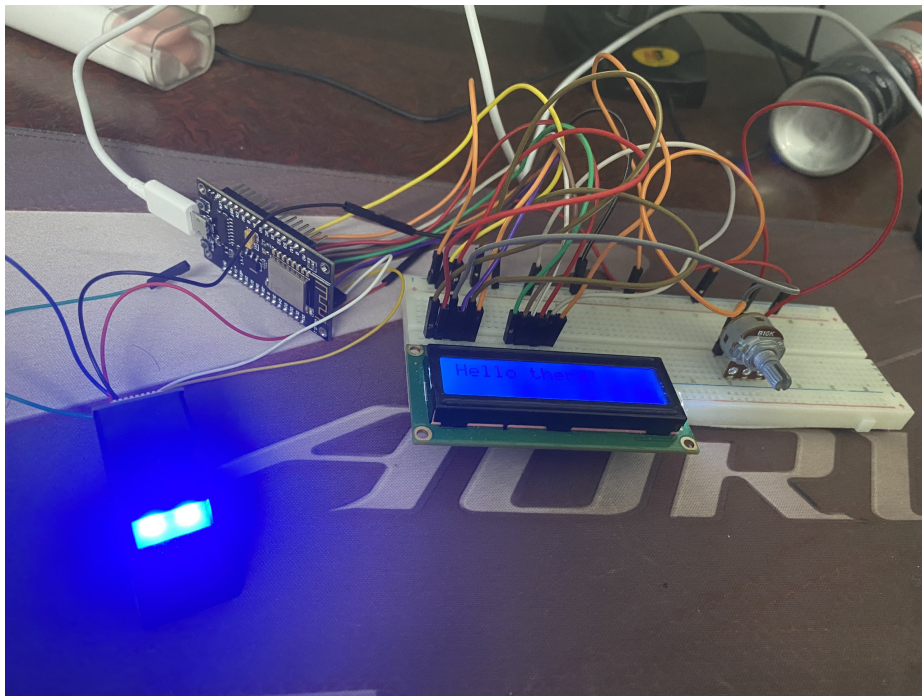


Figure 3. Attendance check process.