

ĐẠI HỌC QUỐC GIA TP. HCM  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



**BÁO CÁO TỔNG KẾT**  
**ĐỀ TÀI KHOA HỌC VÀ CÔNG NGHỆ SINH VIÊN NĂM 2023**

*Tên đề tài tiếng Việt:*

***THIẾT KẾ HỆ THỐNG TƯỚI NƯỚC VÀ THEO DÕI TĂNG TRƯỞNG CHIỀU CAO CÂY TRỒNG TỰ ĐỘNG ỨNG DỤNG CHO NHÀ ƯƠM THÔNG MINH***

*Tên đề tài tiếng Anh:*

***DESIGN AN AUTOMATIC WATERING AND HEIGHT TRACKING SYSTEM FOR SMART GREEN HOUSE***

Khoa/ Bộ môn: **KỸ THUẬT MÁY TÍNH**

Thời gian thực hiện: **6 THÁNG**

Cán bộ hướng dẫn: **TS. ĐOÀN DUY**

Tham gia thực hiện:

TT	Họ và tên, MSSV	Chịu trách nhiệm	Số điện thoại	Gmail
1	Nguyễn Minh Trí (19522388)	Chủ nhiệm	0905163201	19522388@gm.uit.edu.vn
2	Nguyễn Hữu Thành (19522236)	Thành viên	0867242901	19522236@gm.uit.edu.vn



ĐẠI HỌC QUỐC GIA TP. HCM  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Ngày nhận hồ sơ

Mã số đề tài

(Do CQ quản lý ghi)

## BÁO CÁO TỔNG KẾT

Tên đề tài tiếng Việt:

**THIẾT KẾ HỆ THỐNG TƯỚI NƯỚC VÀ THEO DÕI TĂNG TRƯỞNG CHIỀU CAO CÂY TRỒNG TỰ ĐỘNG ỨNG DỤNG CHO NHÀ ƯƠM THÔNG MINH**

Tên đề tài tiếng Anh:

**DESIGN AN AUTOMATIC WATERING AND HEIGHT TRACKING SYSTEM FOR SMART GREEN HOUSE**

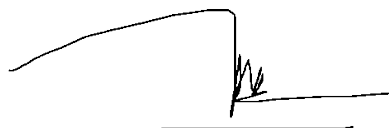
Ngày 01 tháng 01 năm 2024

**Cán bộ hướng dẫn**  
(Họ tên và chữ ký)

  
Đoàn Duy

Ngày 01 tháng 01 năm 2024

**Sinh viên chủ nhiệm đề tài**  
(Họ tên và chữ ký)



.....  
**NGUYỄN MINH TRÍ**

# MỤC LỤC

<b>Chương 1. TỔNG QUAN ĐỀ TÀI</b>	<b>2</b>
1.1 Giới thiệu về OpenCV	3
1.2 Firebase Database	4
<b>Chương 2. QUÁ TRÌNH THỰC HIỆN HỆ THỐNG</b>	<b>5</b>
2.1 Sơ đồ khối	5
2.2 Lưu đồ giải thuật	6
2.3 Phần cứng hệ thống	6
2.4 Nhận diện chiều cao cây	9
2.4.1 Thu thập dữ liệu	9
2.4.2 Lọc màu detect đỉnh cây	12
2.4.3 So sánh với bộ dữ liệu trước đó	15
2.4.4 Xuất chiều cao cây	15
2.5 Đăng thông tin lên Firebase Web	17
2.5.2 Realtime Database	18
2.5.3 Flutter	21
2.5.4. Giao diện App	25
<b>Chương 3: THỰC NGHIỆM ĐỀ TÀI</b>	<b>31</b>
3.1 Thực nghiệm nhiệt độ, độ ẩm không khí, độ ẩm đất	31
3.2 Thực nghiệm đo chiều cao	31
<b>Chương 4. TỔNG KẾT ĐỀ TÀI</b>	<b>38</b>
4.1 Kết quả đạt được	38
4.1.1 Ưu điểm	38
4.1.2 Hạn chế	38
4.1.3 Hướng phát triển	39

# Chương 1. TỔNG QUAN ĐỀ TÀI

## LỜI MỞ ĐẦU

Trong những năm gần đây, thiết bị không dây ứng dụng công nghệ IoT và trí tuệ nhân tạo đang phát thu hút nhiều sự quan tâm, nghiên cứu của các nhà khoa học trên thế giới và chúng đã có những ứng dụng mạnh mẽ trong ngành công nghiệp điện tử nói riêng và trong cuộc sống của con người nói chung. Nhưng về cơ bản sự phát triển vượt bậc của công nghệ ngày nay đều hướng đến trải nghiệm của người dùng, giúp nâng cao và phục vụ những nhu cầu của con người. Và trong nền công nghiệp 4.0 thì các nhà phát triển đều hướng đến sự tự động hóa và kết nối mọi vạn vật với nhau mục tiêu là giảm thiểu khối lượng công việc của con người và nâng cao năng suất, sự chính xác

Tự động hóa thống trị thế giới ngày nay. Nó là một kỹ thuật sử dụng máy tính hoặc điện thoại di động để theo dõi và kiểm soát các thông số đơn giản của cuộc sống hàng ngày. Tiêu chuẩn cuộc sống của chúng ta sẽ được nuôi dưỡng bằng cách sử dụng tự động hóa cho những việc đơn giản. Sử dụng khái niệm IOT, chúng tôi tạo ra các cảm biến để giao tiếp với nhau, vốn rất mạnh trong tự động hóa.

Khi mọi người cố gắng trồng và xây dựng khu vườn của riêng mình, họ chỉ thận trọng trong việc bảo trì ở giai đoạn đầu. Nhiều ngày trôi qua do không được bảo dưỡng, khu vườn sẽ bị phá hủy. Sản phẩm này sẽ giúp mọi người tự động theo dõi các thông số và đảm bảo duy trì vườn cây. Ngoài ra, thông qua chiều cao của cây ở các giai đoạn bất kỳ hệ thống còn có thể cung cấp cho người dùng thời điểm sinh trưởng của cây, qua đó người dùng có thể nắm bắt được sự sinh trưởng của cây để có biện pháp chăm sóc. Nó đóng một vai trò quan trọng và phục vụ như một người bạn đồng hành tốt cho cây trồng. IOT cung cấp các giải pháp cho các vấn đề khác

nhau và nó cho phép mọi thứ cảm nhận hoặc điều khiển từ xa trong cơ sở hạ tầng mạng.

## **1.1 Giới thiệu về OpenCV**

OpenCV (Open Source Computer Vision) là một thư viện mã nguồn mở cho ngôn ngữ lập trình Python và các ngôn ngữ khác, nó cung cấp các thư viện và công cụ để xử lý và phân tích ảnh và video. Nó được sử dụng rộng rãi trong các ứng dụng như xử lý ảnh, nhận dạng khuôn mặt, phát hiện đối tượng, tạo ra ảnh 3D và nhiều ứng dụng khác trong lĩnh vực xử lý ảnh và video.

OpenCV có thể được cài đặt và sử dụng với nhiều loại ngôn ngữ lập trình như Python, C/C++ và Java, v. Ngoài ra, nó cũng dễ dàng tương thích với các hệ điều hành khác nhau, bao gồm từ Windows, Linux, Mac OS, iOS cho đến Android.

OpenCV cung cấp rất nhiều tính năng khác nhau cho phép bạn xử lý và phân tích ảnh và video. Một số tính năng chính của OpenCV bao gồm:

- Xử lý ảnh: OpenCV cung cấp các hàm để thay đổi kích thước của ảnh, làm mờ hoặc làm sáng ảnh, đổi màu sắc của ảnh, v.v.
- Nhận dạng đối tượng: OpenCV có thể sử dụng các tập lệnh đã được huấn luyện sẵn để phát hiện các đối tượng như khuôn mặt, xe hơi, v.v. trong một ảnh hoặc video.
- Phát hiện biên: OpenCV có thể sử dụng các thuật toán để tìm các biên của đối tượng trong một ảnh hoặc video.
- Phát hiện điểm: OpenCV có thể sử dụng các thuật toán để tìm các điểm đặc trưng trong một ảnh hoặc video, ví dụ như các điểm Harris corner hoặc SIFT.
- Xử lý video: OpenCV có thể đọc và ghi các tập tin video, và cũng có thể xử lý các khung hình trong một video theo từng khung hình.

Để cài gói thư viện OpenCV trong Pycharm ta thực hiện lệnh trên terminal :  
pip install opencv-python.

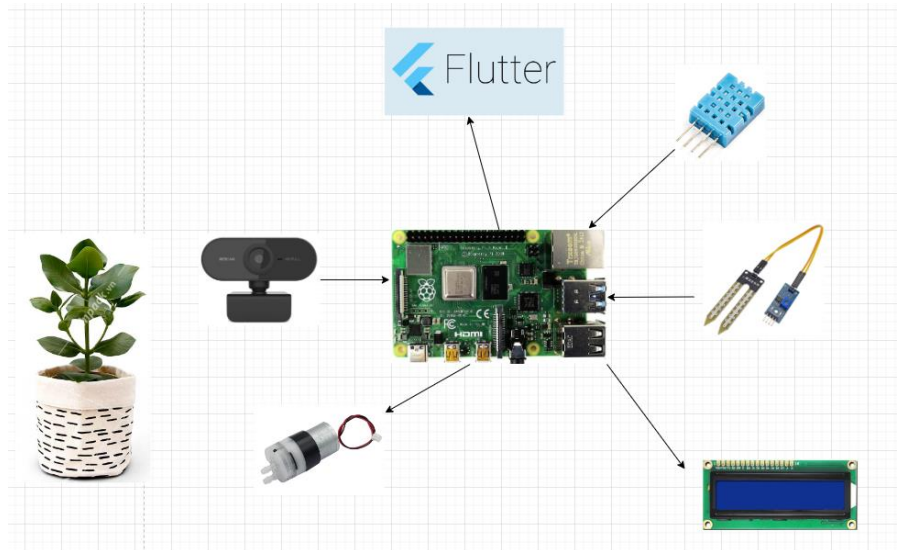
## **1.2 Firebase Database**

Firebase là dịch vụ cơ sở dữ liệu dựa trên đám mây do Google cung cấp. Nó cung cấp khả năng đồng bộ hóa dữ liệu theo thời gian thực và cập nhật dữ liệu tự động trên các thiết bị, làm cho nó trở thành lựa chọn lý tưởng để xây dựng các ứng dụng hợp tác và tương tác.

Firebase cung cấp nhiều tùy chọn cơ sở dữ liệu, bao gồm Cloud Firestore, cơ sở dữ liệu tài liệu NoSQL và Cơ sở dữ liệu thời gian thực, cơ sở dữ liệu NoSQL JSON. Bạn có thể chọn cơ sở dữ liệu phù hợp nhất với nhu cầu của mình và sử dụng SDK Firebase để dễ dàng truy cập và thao tác dữ liệu của bạn từ ứng dụng di động, web hoặc ứng dụng phụ trợ.

## Chương 2. QUÁ TRÌNH THỰC HIỆN HỆ THỐNG

### 2.1 Sơ đồ khối

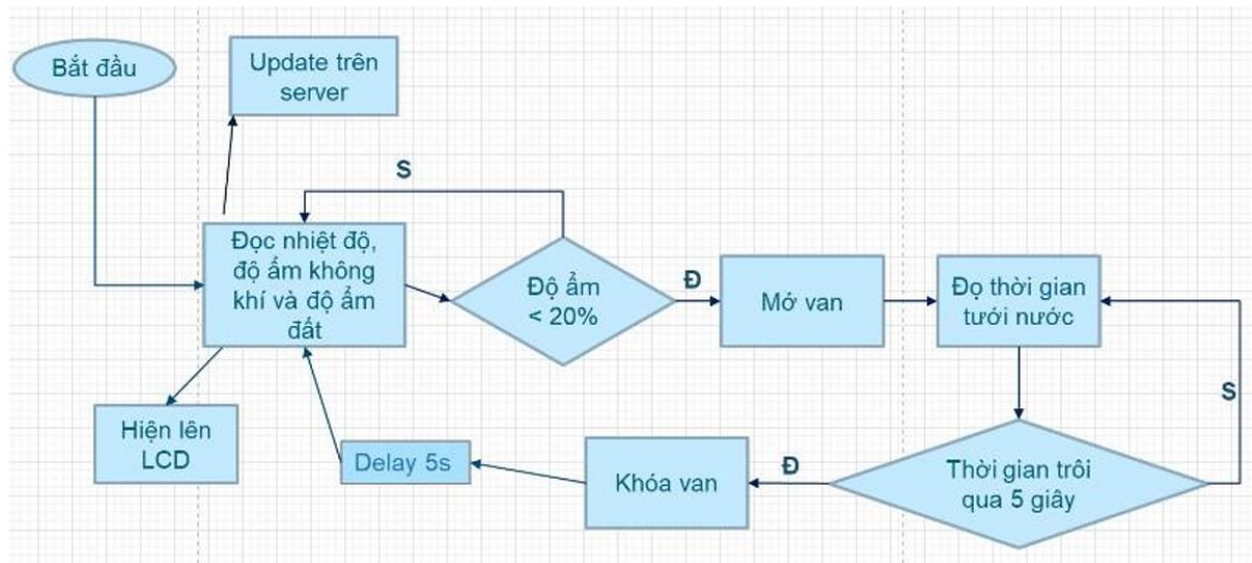


Hình 2.1 Sơ đồ khối của hệ thống

Phần cứng điều khiển là ESP32, nhận dữ liệu nhiệt độ, độ ẩm không khí từ DHT11, độ ẩm đất từ cảm biến độ ẩm đất, 1 màn hình LCD để hiển thị các thông tin môi trường xung quanh, và thông số chiều cao cây. Máy bơm sẽ mở ra để hệ thống tưới nước.

Một camera để nhận diện chiều cao cây. Phần mềm bao gồm Firebase, phần mềm này sẽ nhận dữ liệu mà ESP32 đã thu thập được, và dữ liệu chiều cao thu thập được từ camera. Những dữ liệu đó sẽ được hiển thị lên Firebase.

## 2.2 Lưu đồ giải thuật



Hình 2.2 Lưu đồ giải thuật của hệ thống tưới nước

Ban đầu cảm biến sẽ đo độ ẩm đất, nếu độ ẩm dưới 20% thì sẽ tiến hành cho mở van và tưới nước theo thời gian được chỉ định. Sau khi tưới nước xong trong 5 giây, hệ thống sẽ khóa van và đợi thêm 5 giây, sau đó sẽ tiến hành đo lại độ ẩm đất, và vòng lặp đó cứ tiếp tục tiếp diễn. Trong lúc đo nhiệt độ, độ ẩm không khí, độ ẩm đất, hệ thống cũng sẽ hiện những thông tin đó lên màn hình LCD và Firebase web.

## 2.3 Phần cứng hệ thống

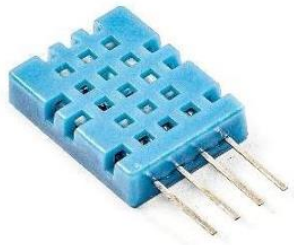
Bo mạch điều khiển: ESP 32 Node MCU 32S (hình 2.3)



Hình 2.3 ESP 32

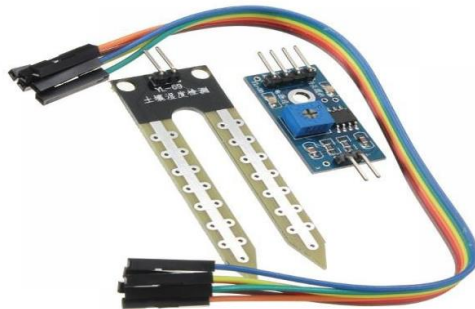
Cảm biến nhiệt độ độ ẩm môi trường : DHT11 (hình 2.4)





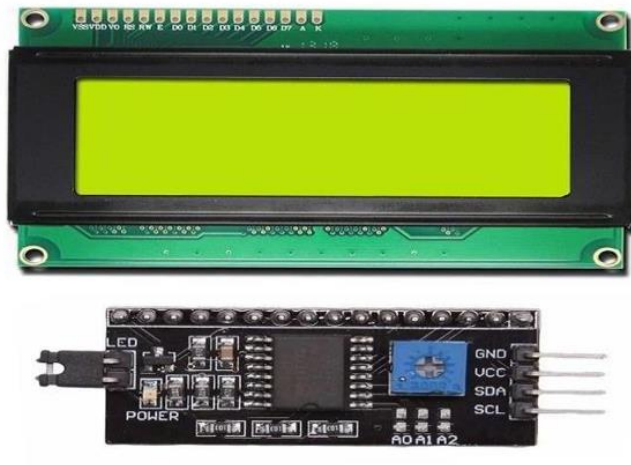
Hình 2.4 Cảm biến nhiệt độ, độ ẩm không khí

Cảm biến độ ẩm đất (hình 2.5)



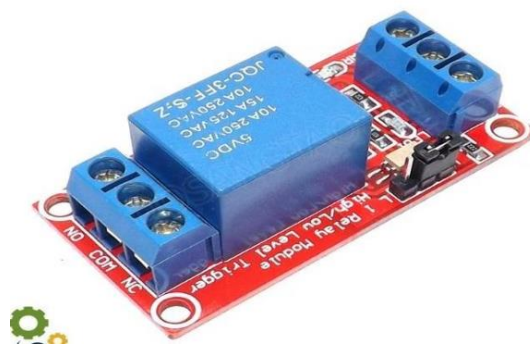
Hình 2.5 Cảm biến độ ẩm đất

LCD 2004A + I2C (hình 2.6):



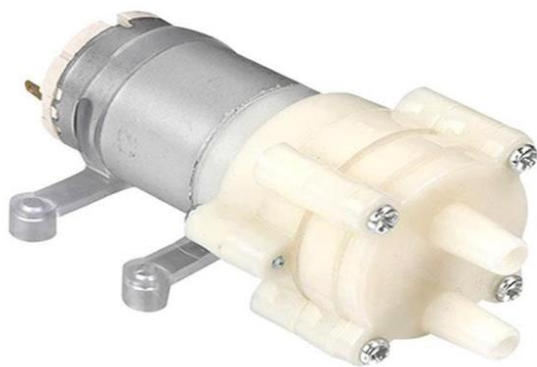
Hình 2.6 Màn hình LCD

Relay 5V (hình 2.7)



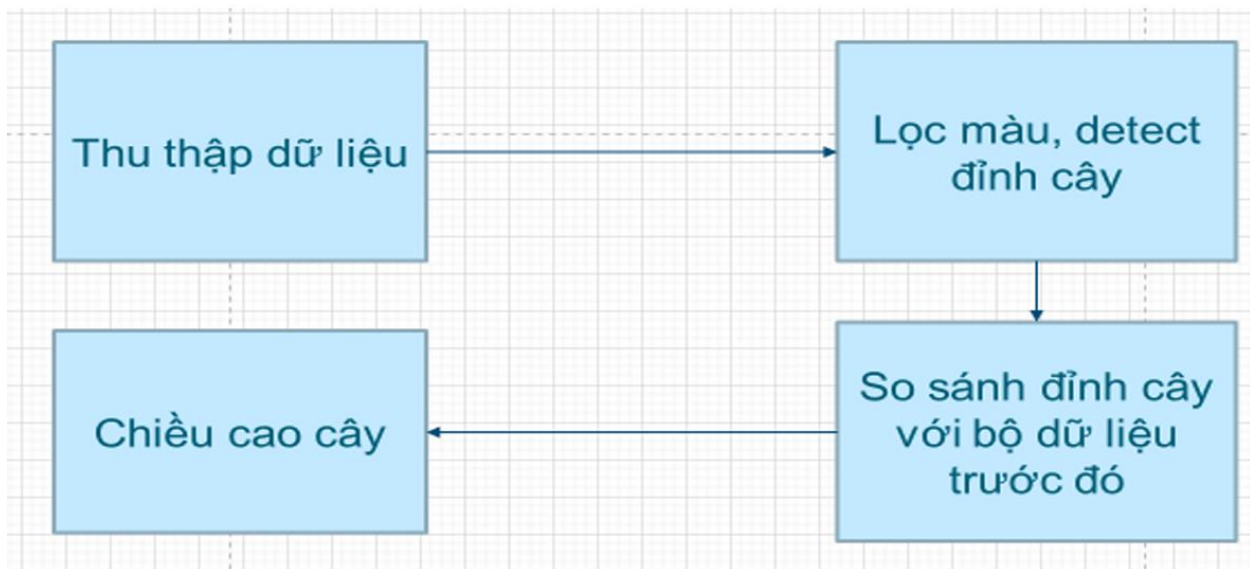
Hình 2.7 Relay 5V

Máy bơm 12V (hình 2.8)



Hình 2.8 Máy bơm

## 2.4 Nhận diện chiều cao cây



Hình 2.9 Các bước đo chiều cao cây

### 2.4.1 Thu thập dữ liệu

Như trên hình 2.9, ở phần thu thập dữ liệu, nhóm sẽ tiến hành set up một góc cam cố định, với khoảng cách từ camera đến vật thể là 141cm, chiều cao từ cam đến mặt đất là 48,75cm.

Nhóm sẽ tiến hành đánh các dấu mốc trên tường, mỗi mốc tương ứng với 1 chiều cao cố định, từng mốc sẽ cách nhau 2cm.

Tiến hành chụp ảnh từng mốc, lọc màu, dùng thư viện matplotlib để lấy được thông số trục y của từng mốc. Từ đó thu thập được tập dữ liệu.

Phía dưới là những bức ảnh của từng mốc 14,16,18,20 - Bộ dữ liệu bao gồm đến mốc thứ 86, nhưng do giới hạn của bài báo cáo, nên nhóm sẽ up lấy ví dụ của các mốc từ 14-20cm.



Hình 2.10

a/ Mốc 14cm đời thực

b/ Mốc 16cm đời thực

Những bức ảnh như hình 2.10a và 2.10b sẽ tiến hành được lọc màu và dùng matplotlib để lấy thông số từ trực y.



Hình 2.11 Thông số trực y của mốc 14cm

Như ta có thể thấy ở hình 2.11, ảnh mốc 14cm được lọc màu, và lấy được thông số trực y tương ứng là 827. Làm tương tự với các mốc khác, ta thu được bộ dữ liệu như ở bảng 1

Height	Pixel	Height	Pixel	Height	Pixel	Height	Pixel
14	827	34	611	54	391	74	169
16	808	36	589	56	364	76	142
18	788	38	568	58	341	78	120
20	767	40	542	60	320	80	98
22	745	42	521	62	297	82	77
24	723	44	502	64	274	84	None
26	701	46	477	66	259	86	34
28	678	48	454	68	229		
30	656	50	430	70	209		
32	630	52	413	72	187		

Bảng 1 Thông số trực y so với chiều cao thực

## 2.4.2 Lọc màu detect đỉnh cây

```
img2 = cv2.imread("/content/WIN_20221228_13_58_21_Pro.jpg")
image2 = cv2.cvtColor(img2, cv2.COLOR_BGR2HSV)
lower_green1 = np.array([30,120,111])
upper_green1 = np.array([42,255,255])
lower_green2 = np.array([42,80,111])
upper_green2 = np.array([63,255,255])
lower_green3 = np.array([63,120,111])
upper_green3 = np.array([70,255,255])
lower_green4 = np.array([70,150,111])
upper_green4 = np.array([75,255,255])
lower_green5 = np.array([75,200,111])
upper_green5 = np.array([80,255,255])
mask_green1 = cv2.inRange(image2,lower_green1,upper_green1)
mask_green2 = cv2.inRange(image2,lower_green2,upper_green2)
mask_green3 = cv2.inRange(image2,lower_green3,upper_green3)
mask_green4 = cv2.inRange(image2,lower_green4,upper_green4)
mask_green5 = cv2.inRange(image2,lower_green5,upper_green5)
mask_green = mask_green1+ mask_green2+ mask_green3+ mask_green4+ mask_green5

cv2.imshow(mask_green)
```

Hình 2.12 Code lọc màu xanh lá cây

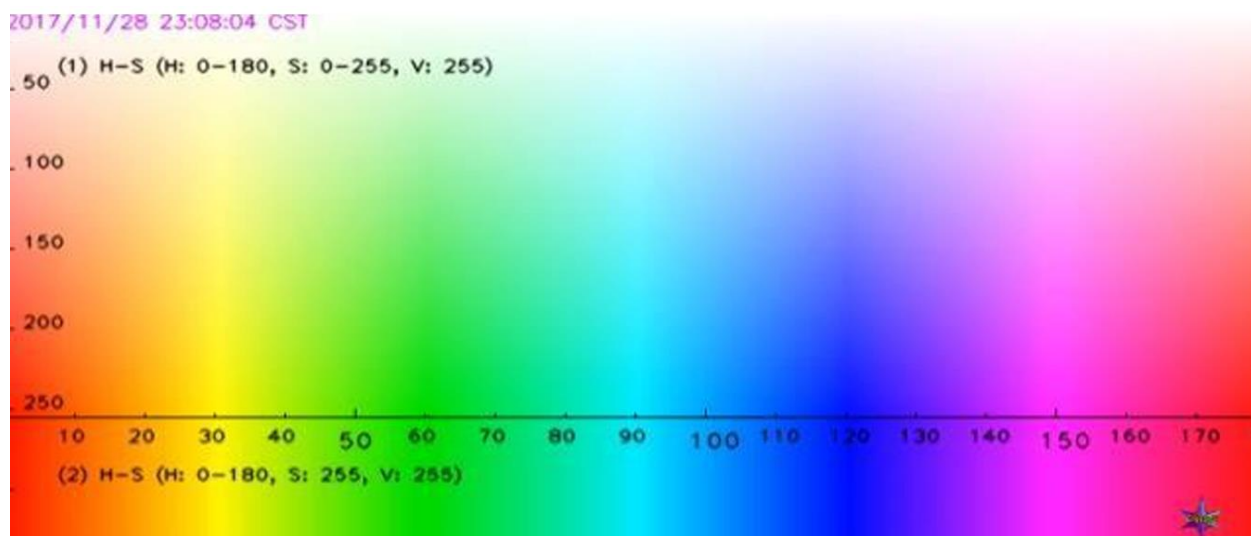


Ở đây nhóm đã tiến hành đọc cái cây vừa chụp được ở hình 2.13 bằng code như trong hình 2.12



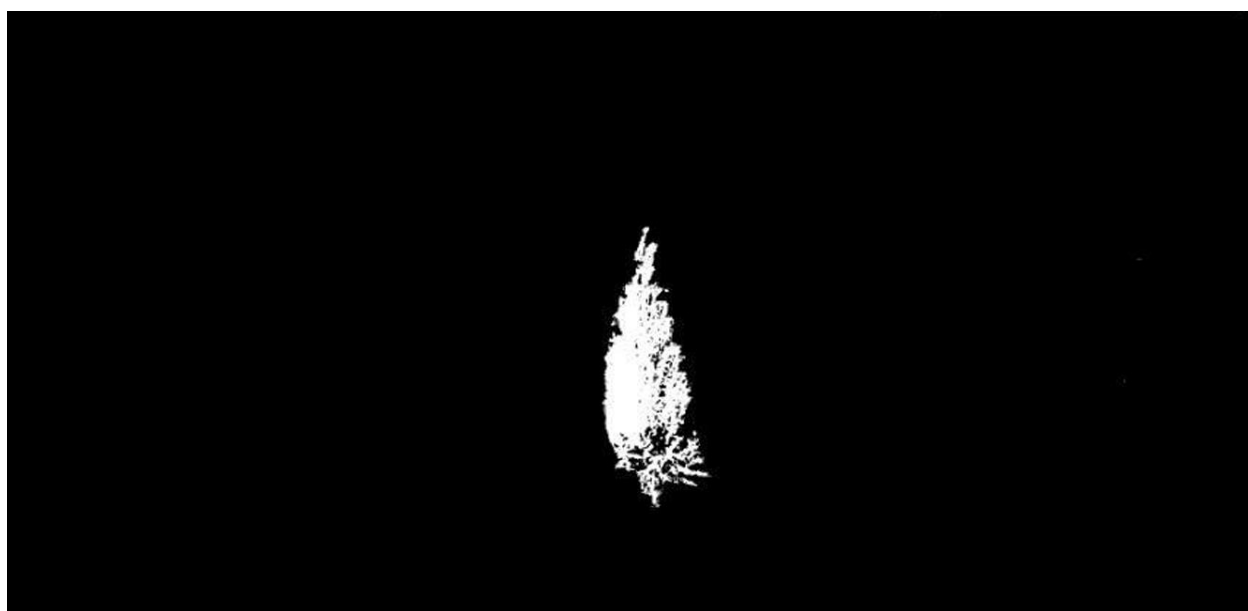
Hình 2.13 Cây mẫu

Các biến `lower_green1` đến `upper_green5` trong code hình 2.12 là các khoảng cao thấp nhằm xác định hết các khoảng màu xanh trong bảng màu sắc (hình 2.14) dưới đây.



Hình 2.14 Bảng màu HSV

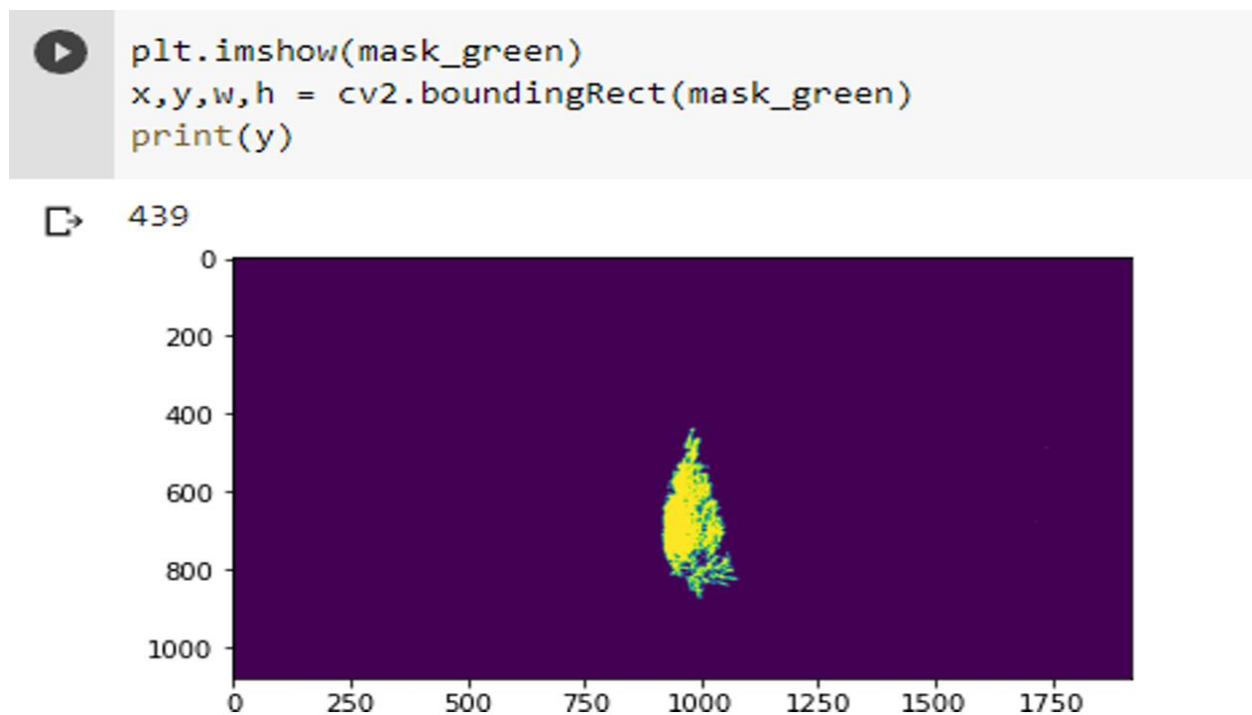
Sau khi xác định được khoảng màu xanh cần thiết, ta tiến hành lọc màu cái cây và cho ra mask của cái cây như hình 2.15.



Hình 2.15 Mask của cây



### 2.4.3 So sánh với bộ dữ liệu trước đó



Hình 2.16 Lấy thông số trục y của đỉnh cây

Sau khi đã lọc được màu, ta tiến hành dùng matplotlib (hình 2.16) để lấy được thông số y của bức ảnh này.

Khi đã có được thông số y, tiến hành so sánh với bộ dữ liệu trước đó và xuất ra chiều cao của cây.

### 2.4.4 Xuất chiều cao cây

```

if y > 808 and y < 827:
    h= (y-808)/(827-808)*2 + 14
elif y> 788 and y<808:
    h= (y-788)/(808-788)*2 + 16
elif y> 767and y< 788:
    h= (y-767)/(788-767)*2 + 18
elif y> 745 and y< 767:
    h= (y-745)/(767-745)*2 + 20
elif y> 723 and y< 745:
    h= (y-723)/(745-723)*2 + 22
elif y> 701and y< 723:
    h= (y-701)/(723-701)*2 + 24
elif y> 678 and y< 701:
    h= (y-678)/(701-678)*2 + 26
elif y> 656and y< 678:
    h= (y-656)/(678-656)*2 + 28
elif y> 630and y< 656:
    h= (y-630)/(656-630)*2 + 30
elif y>611 and y< 630:
    h= (y-611)/(630-611)*2 + 32
elif y> 589 and y< 611:
    h= (y-589)/(611-589)*2 + 34
elif y> 568and y< 589:
    h= (y-568)/(589-568)*2 + 36
elif y> 542and y< 568:
    h= (y-542)/(568-542)*2 + 38
elif y>521 and y< 542:
    h= (y-521)/(542-521)*2 + 40
elif y> 502 and y< 521:
    h= (y-502)/(521-502)*2 + 42
elif y> 477and y< 502:
    h= (y-477)/(502-477)*2 + 44
elif y> 454and y< 477:
    h= (y-454)/(477-454)*2 + 46
elif y>430 and y< 454:
    h= (y-430)/(454-430)*2 + 48
elif y> 413 and y< 430:
    h= (y-413)/(430-413)*2 + 50
elif y> 391and y< 413:
    h= (y-391)/(413-391)*2 + 52
elif y> 364and y< 391:
    h= (y-364)/(391-364)*2 + 54
elif y>341 and y< 364:
    h= (y-341)/(364-341)*2 + 56
elif y> 274and y< 297:
    h= (y-274)/(297-274)*2 + 62
elif y>259 and y< 274:
    h= (y-259)/(274-259)*2 + 64
elif y> 229 and y< 259:
    h= (y-229)/(259-229)*2 + 66
elif y> 209and y< 229:
    h= (y-209)/(229-209)*2 + 68
elif y> 187and y< 209:
    h= (y-187)/(209-187)*2 + 70
elif y>169 and y< 187:
    h= (y-169)/(187-169)*2 + 72
elif y> 142 and y< 169:
    h= (y-142)/(169-142)*2 + 74
elif y> 120and y< 142:
    h= (y-120)/(142-120)*2 + 76
elif y> 98and y< 120:
    h= (y-98)/(120-98)*2 + 78
elif y>77 and y< 98:
    h= (y-77)/(98-77)*2 + 80
elif y>34 and y< 77:
    h= (y-34)/(77-34)*2 + 82

```

Hình 2.17 Code để lấy được chiều cao cây từ thông số trực y

Sau khi so sánh được bộ dữ liệu như code trên hình 2.17, nhóm tiến hành xuất ra chiều cao của cây như trong hình 2.18

```

print(h)

```

48.75

Hình 2.18 Kết quả đo chiều cao cây

## 2.5 Đăng thông tin lên Firebase Web

### 2.5.1 Authentication

Firebase cũng cung cấp một số tùy chọn khác để xác thực người dùng, chẳng hạn như xác thực số điện thoại, nơi người dùng có thể đăng nhập bằng số điện thoại của họ và mã một lần được gửi qua SMS; và xác thực ẩn danh, nơi người dùng có thể đăng nhập mà không cần tạo tài khoản cố định.

Để sử dụng xác thực Firebase trong ứng dụng của mình, bạn có thể sử dụng SDK xác thực Firebase, cung cấp thư viện phía máy khách và thư viện phía máy chủ mà bạn có thể sử dụng để xác thực người dùng và quản lý tài khoản của họ. Bạn cũng có thể sử dụng bảng điều khiển Firebase để quản lý người dùng và trạng thái xác thực của họ. Đề tài nhóm thực hiện xác thực bằng Email.

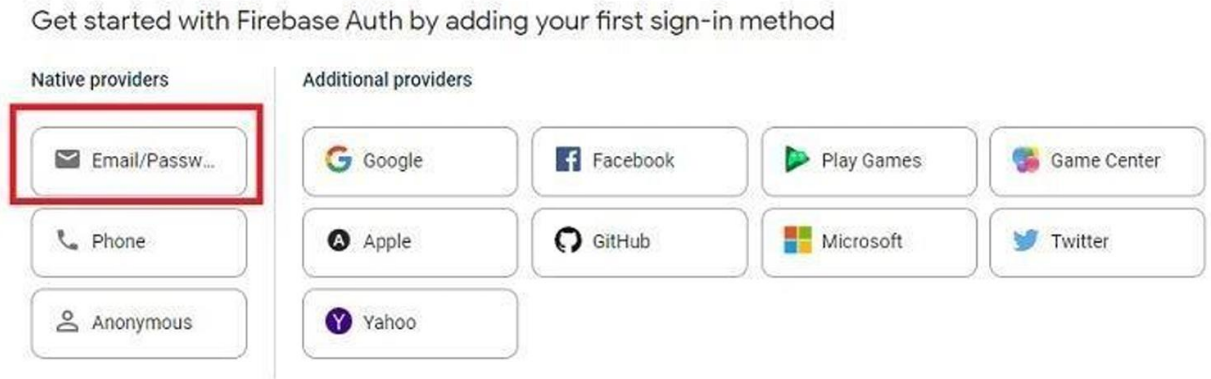
Để tạo xác thực Email trên Firebase ta thực hiện như sau:

1/ Click **Authentication** và ấn **Get Started** như ở hình 2.19



Hình 2.19 Bắt đầu project trên Firebase

2/ Select Email/ Password như ở hình 2.20



Hình 2.20 Chọn email để đăng nhập

3/ Add User là Email và Password của mình như trong hình 2.21.



Hình 2.21 Thêm email

## 2.5.2 Realtime Database

Một trong những tính năng chính của Cơ sở dữ liệu thời gian thực là khả năng đồng bộ hóa dữ liệu trong thời gian thực. Điều này có nghĩa là mọi thay đổi được thực hiện đối với dữ liệu sẽ tự động được truyền tới tất cả các máy khách được kết nối, làm cho nó trở thành lựa chọn lý tưởng để xây dựng các ứng dụng hợp tác và tương tác.

Cơ sở dữ liệu thời gian thực cũng được thiết kế để dễ sử dụng, với API đơn giản cho phép bạn lưu trữ và truy xuất dữ liệu bằng các thao tác JavaScript tiêu chuẩn. Bạn có thể sử dụng SDK Firebase để truy cập Cơ sở dữ liệu thời gian thực từ các ứng dụng di động, web hoặc phụ trợ của mình.

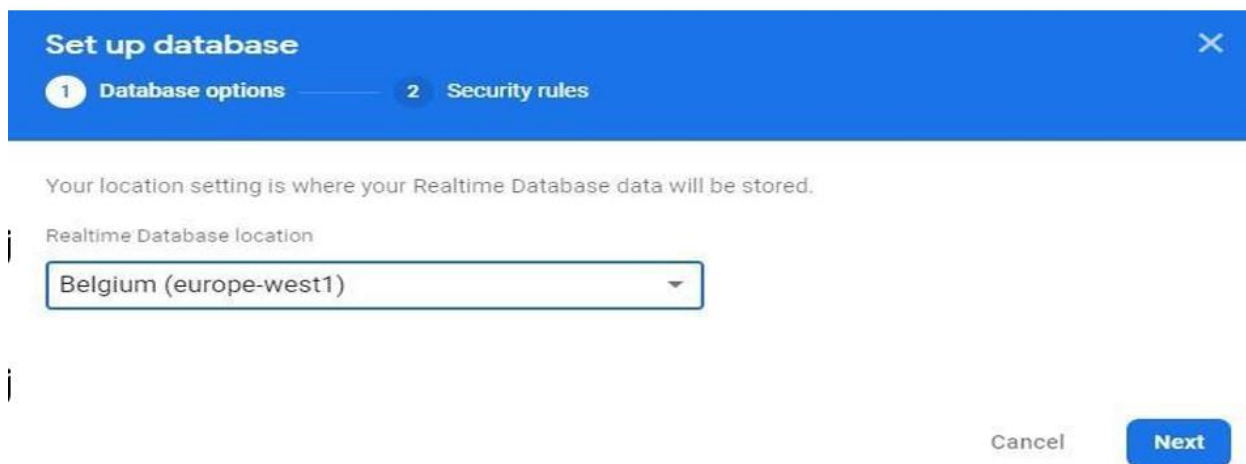
Để tạo Realtime Database trên Firebase, ta sẽ có những bước sau:

1/ Click vào **Realtime Database** và ấn vào **Get Started** như ở hình 2.22



Hình 2.22 Khởi tạo Realtime Database

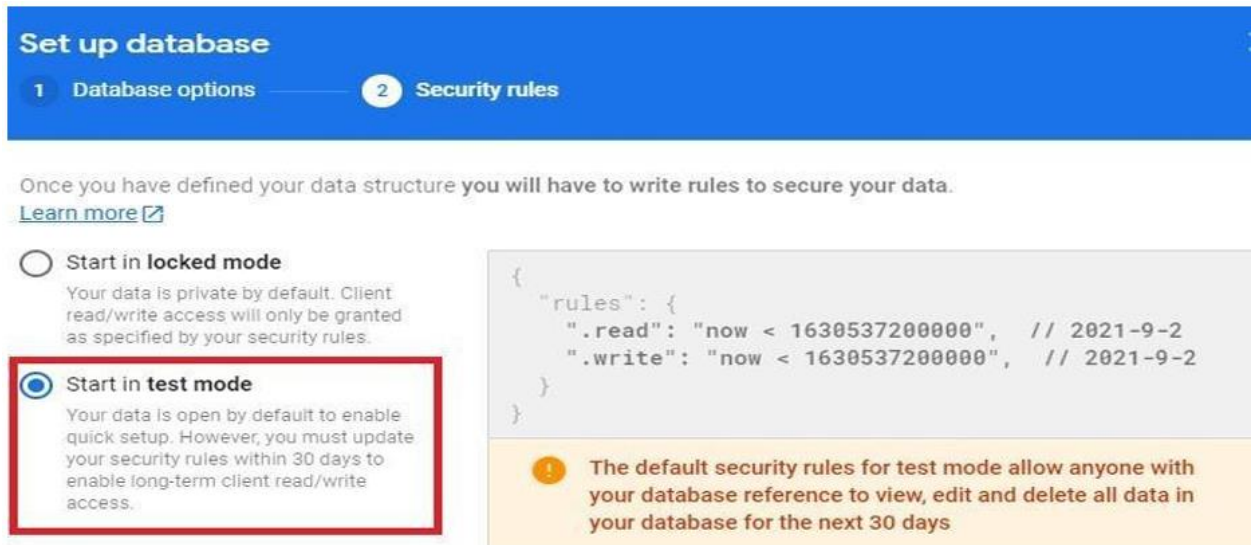
2/ Lựa chọn Database location ở gần chỗ bạn như ở hình 2.23



Hình 2.23 Chọn Location cho Database

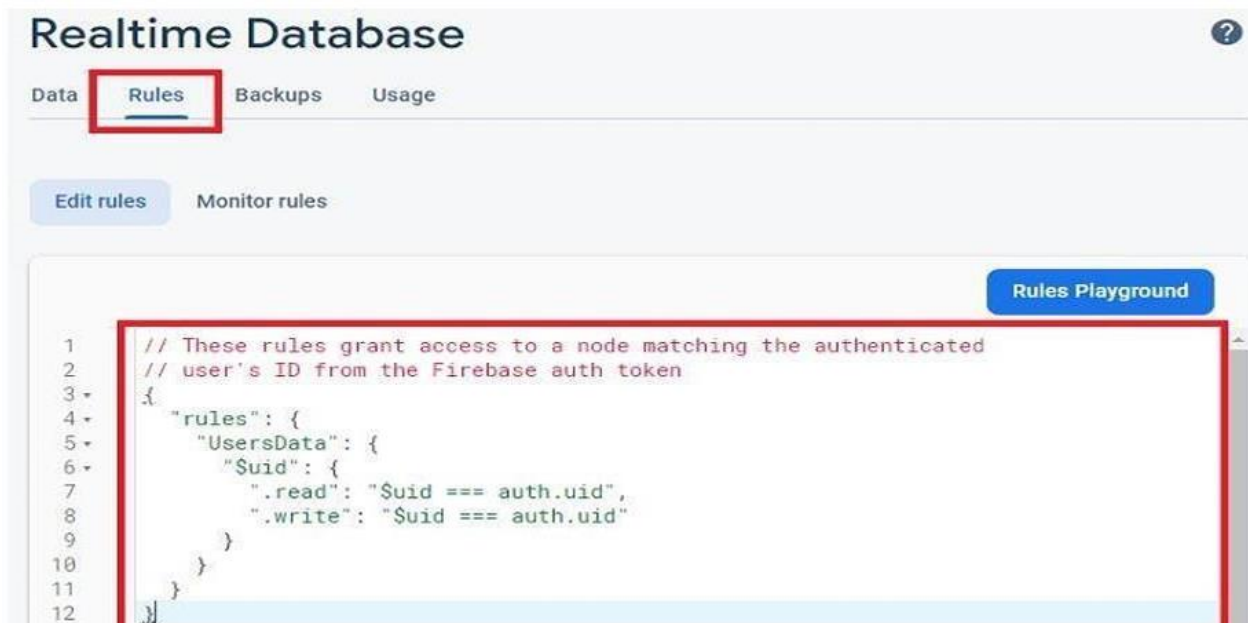
3/ Chọn vào **Start in Test mode** như trong hình 2.24





Hình 2.24 Chọn mode cho database

#### 4/ Cài đặt Rules theo mẫu như trong hình 2.25



Hình 2.25 Cài đặt Rules

Các quy tắc này cấp quyền truy cập vào một nút khớp với UID “O5JGsNcLDkNztsdbFJwLH72JGCZ2” mà nhóm lấy được từ xác thực email ở trên. Điều này có nghĩa là người dùng chỉ có thể truy cập các nút nằm dưới một nút

có UID người dùng tương ứng. Nếu có dữ liệu khác được xuất bản trên cơ sở dữ liệu, không phải trong nút có UID của người dùng, thì người dùng đó không thể truy cập dữ liệu đó.

### 2.5.3 Flutter

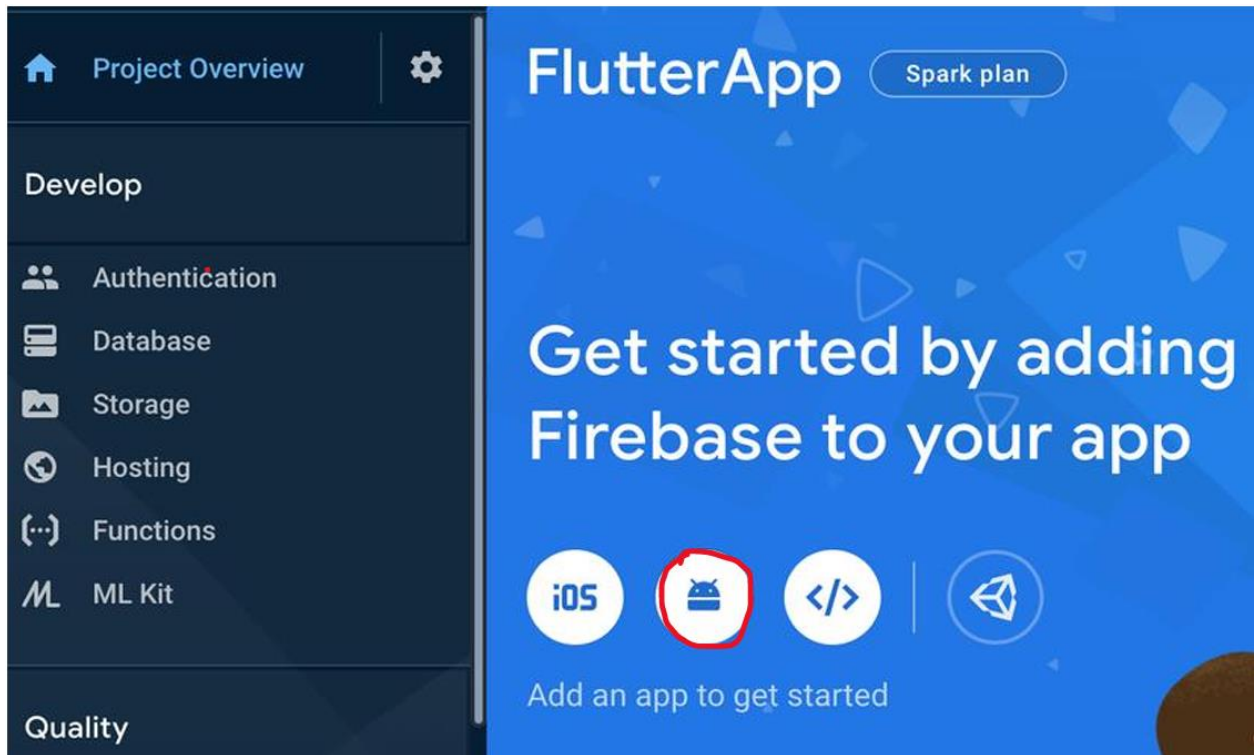


*Hình 2.26 Flutter*

Flutter (hình 2.26) được Google giới thiệu là một người mới trong thế giới ứng dụng di động. Là một SDK mới của Google dành cho các thiết bị di động giúp developers và designers xây dựng nhanh chóng ứng dụng dành cho các thiết bị di động (Android, iOS). Flutter là dự án mã nguồn mở đang trong giai đoạn thử nghiệm. Flutter bao gồm Reactive framework và công nghệ hiển thị 2D (2D rendering engine) và các công cụ phát triển (development tool). Các thành phần này làm việc cùng nhau giúp ta thiết kế, xây dựng, test, debug ứng dụng. Không có gì ngạc nhiên khi Flutter giúp các nhà phát triển tạo ra các ứng dụng native đẹp mắt và giúp họ phát triển các ứng dụng đa nền tảng một cách dễ dàng.

#### 2.5.3.1. Cấu hình Firebase cho Flutter

Sau khi tạo xong project Firebase. Ta chọn vào icon Android (hình 2.27) để thêm Android vào project Firebase mới.



Hình 2.27 Khởi tạo Flutter

Sau đó điền các thông tin lấy ra từ project Flutter. Bao gồm các thông tin về package, SHA1 và nick name của Project sau khi hoàn thành thì click vào Register app. Bước tiếp theo thì mình sẽ phải tải file google-services.json và import vào trong project Flutter như hình 2.28 và 2.29 bên dưới.

The image shows a 'Register app' form. It has a title '1 Register app' with a blue circle containing the number 1. The form contains three input fields: 'Android package name' with a red border and the value 'com.company.appname', 'App nickname (optional)' with the value 'My Android App', and 'Debug signing certificate SHA-1 (optional)' with a long hexadecimal string. Below the fields is a 'Register app' button. A small note at the bottom states: 'Required for Dynamic Links, Invites, and Google Sign-In or phone number support in Auth. Edit SHA-1s in Settings.'

Hình 2.28 Register App



✓

Register app

Android package name: com.xnn.demo, App nickname: Demo

2

Download config file

Instructions for Android Studio below | [Unity](#) [C++](#)

Download google-services.json

Switch to the **Project** view in Android Studio to see your project root directory.

Move the google-services.json file you just downloaded into your Android app module root directory.

google-services.json

Previous

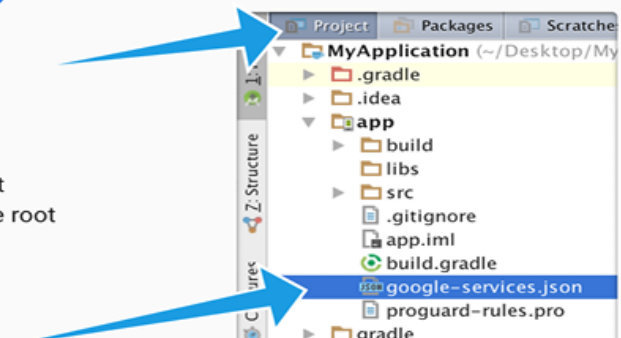
Next

3

Add Firebase SDK

4

Read the Get Started Guide for Android



Hình 2.29 Download config file

Tiếp đến là cấu hình build.gradle theo hướng dẫn như hình 2.30 bên dưới.

Project-level build.gradle (<project>/build.gradle):

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.3'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
        ...
    }
}
```

App-level build.gradle (<project>/<app-module>/build.gradle):

```
apply plugin: 'com.android.application'
// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // add SDKs for desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

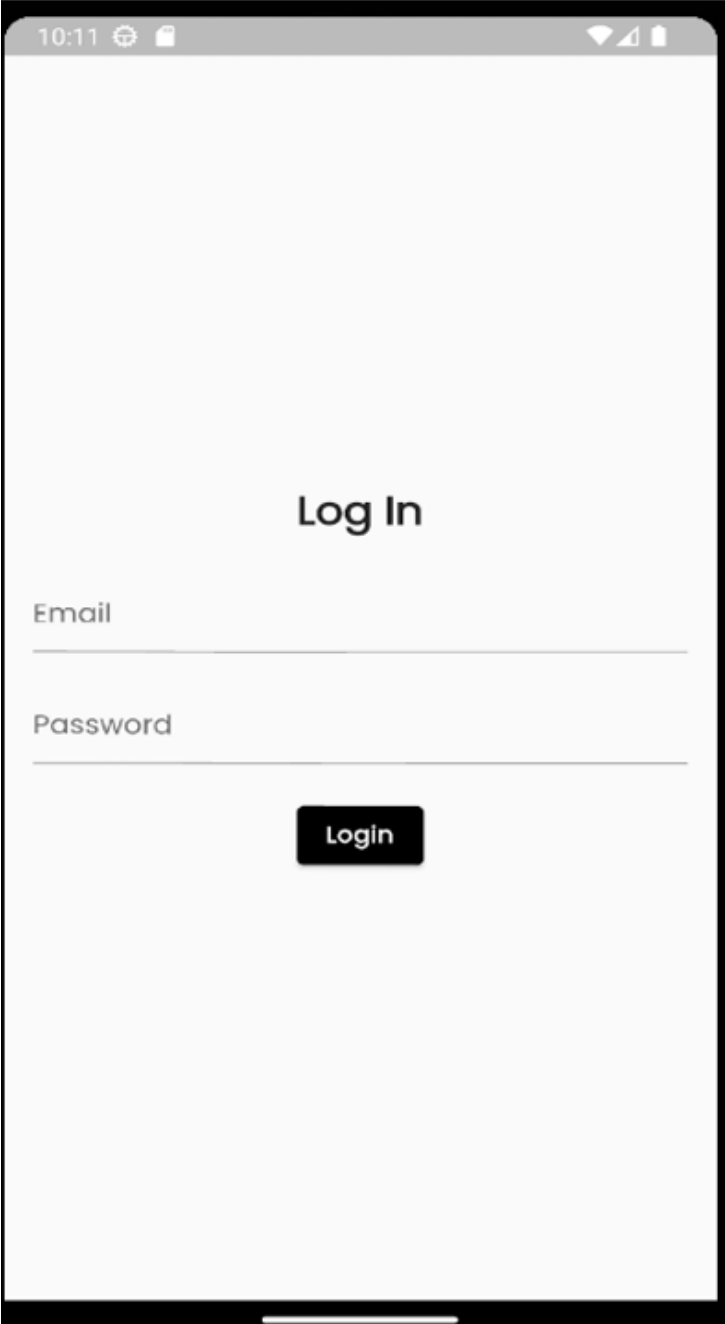
Finally, press "Sync now" in the bar that appears in the IDE:

Gradle files have changed since last sync. [Sync now](#)

Hình 2.30 Cấu hình cho build.gradle

## 2.5.4. Giao diện App

### 2.5.4.1. Giao diện đăng nhập



10:11

Log In

Email

Password

Login

Hình 2.31 Giao diện đăng nhập

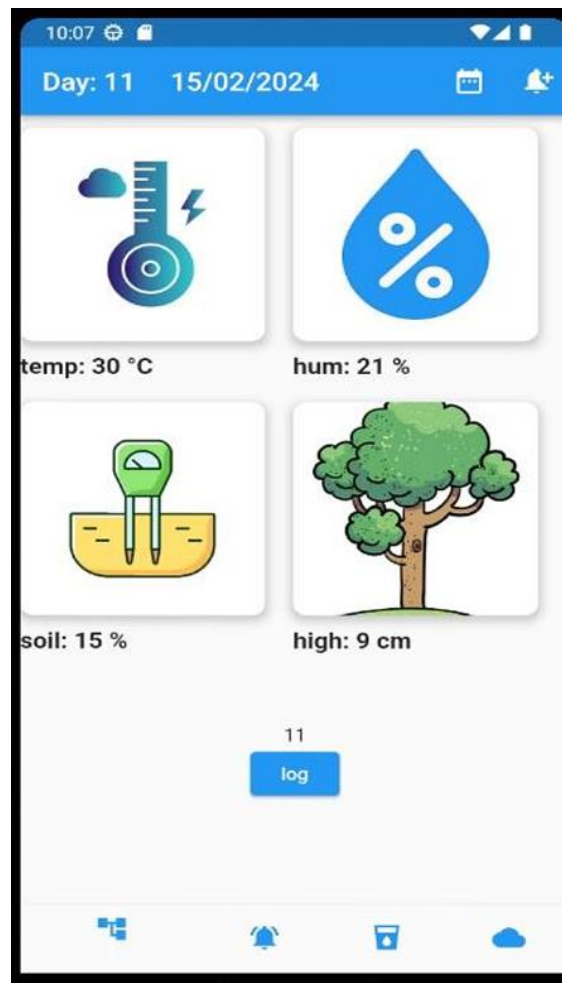
Với giao diện đăng nhập như ở hình 2.31 trên, ta dùng tài khoản Authentication của firebase đã tạo ở trên hình 2.21 để đăng nhập.

#### 2.5.4.2 Giao diện trang chủ

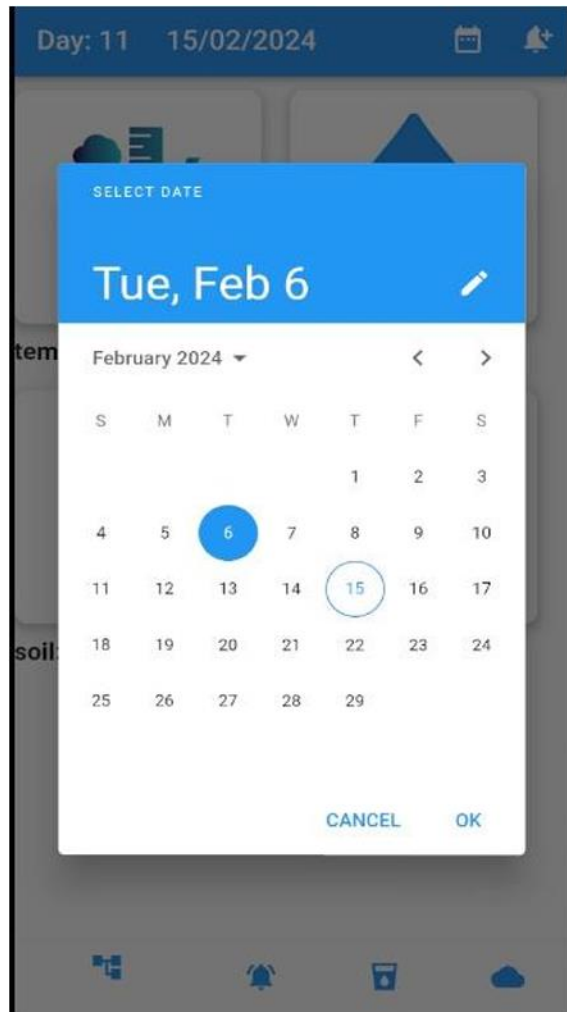
Dữ liệu từ cảm biến sẽ được đẩy lên Database realtime Firebase mà nhóm đã tạo trước đó ở mục 2.5.2, và từ Database đó, dữ liệu sẽ được cập nhật vào App. Tại giao diện trang chủ của App (hình 2.33), những thông số về nhiệt độ, độ ẩm không khí, độ ẩm đất, chiều cao cây sẽ được hiện ra. Database realtime sẽ như ở hình 2.32 dưới.



Hình 2.32 Realtime Database ở Firebase



Hình 2.33 Giao diện trang chủ

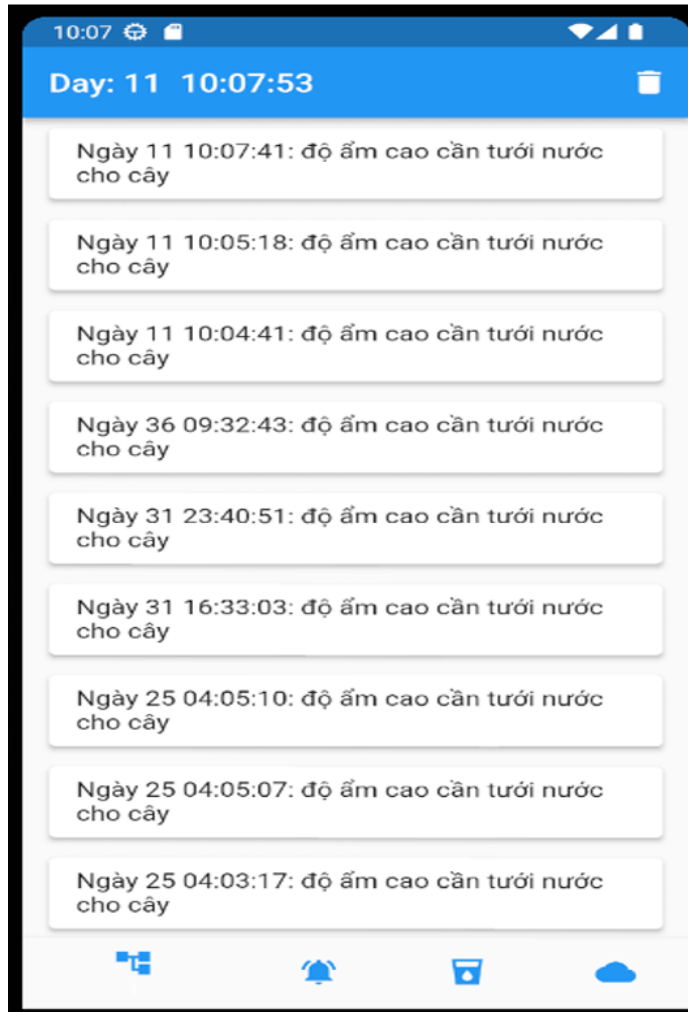


Hình 2.34 Lịch trong App

Bên cạnh đó, App cũng tích hợp lịch (hình 2.34) để người dùng có thể chọn ngày và hỗ trợ trong việc quan sát cây trồng.

#### **2.5.4.3. Giao diện thông báo.**

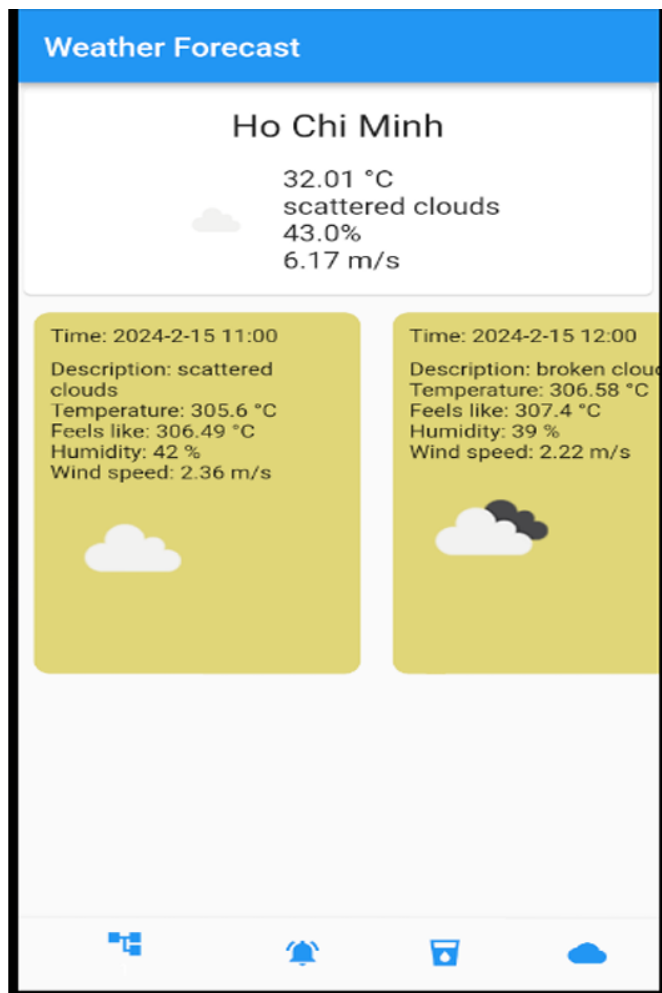
Thông báo (hình 2.35) sẽ được xuất hiện dựa theo kịch bản sinh trưởng của cây. Sẽ thông khi cây thiếu nước hoặc tới ngày thì bón phân và chăm sóc cây theo đúng kịch bản sinh trưởng.



Hình 2.35 Giao diện thông báo

#### 2.5.4.4. Giao diện thông tin dự báo thời tiết.

Dữ liệu dự báo thời tiết được lấy bằng call api tới Openweathermap. Dữ liệu được lấy dự báo trong vòng 5 ngày và cách 1 giờ. Giao diện dự báo thời tiết giống như trong hình 2.36.



Hình 2.36 Giao diện dự báo thời tiết



## Chương 3: THỰC NGHIỆM ĐỀ TÀI

### 3.1 Thực nghiệm nhiệt độ, độ ẩm không khí, độ ẩm đất

Nhiệt độ ngoài trời lúc này đang ở tầm 27 độ C, và nhóm sẽ tiến hành đo nhiệt độ không khí bằng DHT11 trong vòng 30'. Cứ 5p sẽ đo 1 lần. Kết quả được cho ra như bảng 2.

0'	5'	10'	15'	20'	25'	30'
27.3	27.2	27	27.2	27.5	27	27.1

Bảng 2 Nhiệt độ đo được từ DHT11 theo thời gian

Độ ẩm không khí ngoài trời đang tầm 62%, nhóm sẽ tiến hành đo như cách trên, kết quả được cho ra như ở bảng 3.

0'	5'	10'	15'	20'	25'	30'
61	61	62	60	61	63	62

Bảng 3 Độ ẩm không khí được đo theo thời gian

Độ ẩm đất đang là 54%, nhóm cũng sẽ tiến hành đo như cách trên. Kết quả được cho ra như ở bảng 4.

0'	5'	10'	15'	20'	25'	30'
52	53	54	55	54	55	54

Bảng 4 Độ ẩm đất đo theo thời gian

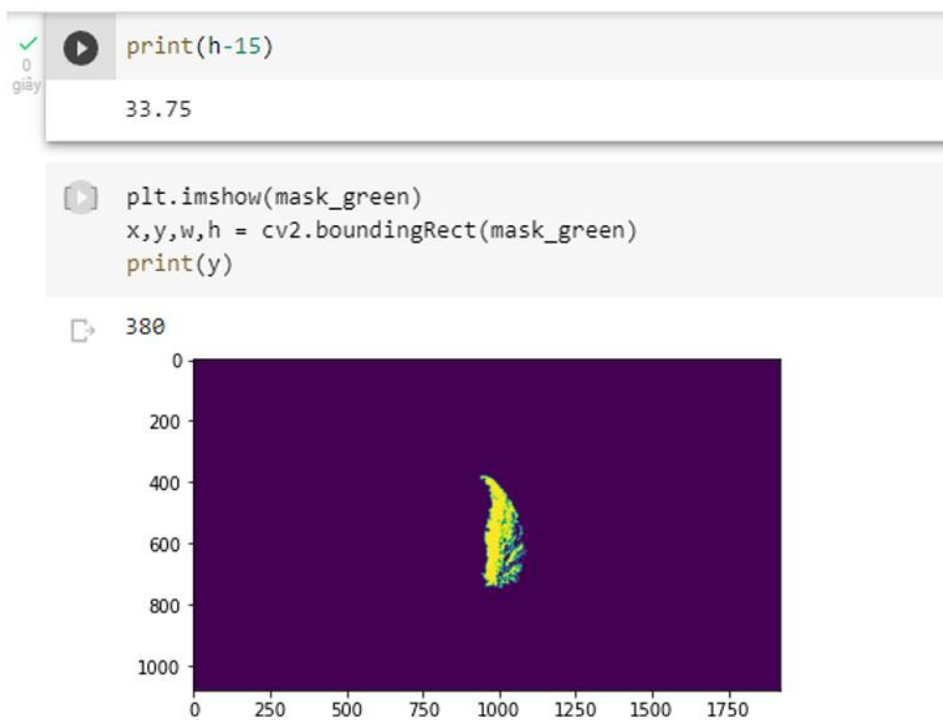
Các cảm biến hoạt động khá tốt, khi không xảy ra quá nhiều sai số trong quá trình đo.

### 3.2 Thực nghiệm đo chiều cao

Ở đây nhóm sẽ tiến hành chụp và đo chiều cao cây ở các khung giờ khác nhau. Chiều cao thực của cây sau khi đã trừ đi 15cm là 40cm



Hình 3.1 Cây lúc 12h

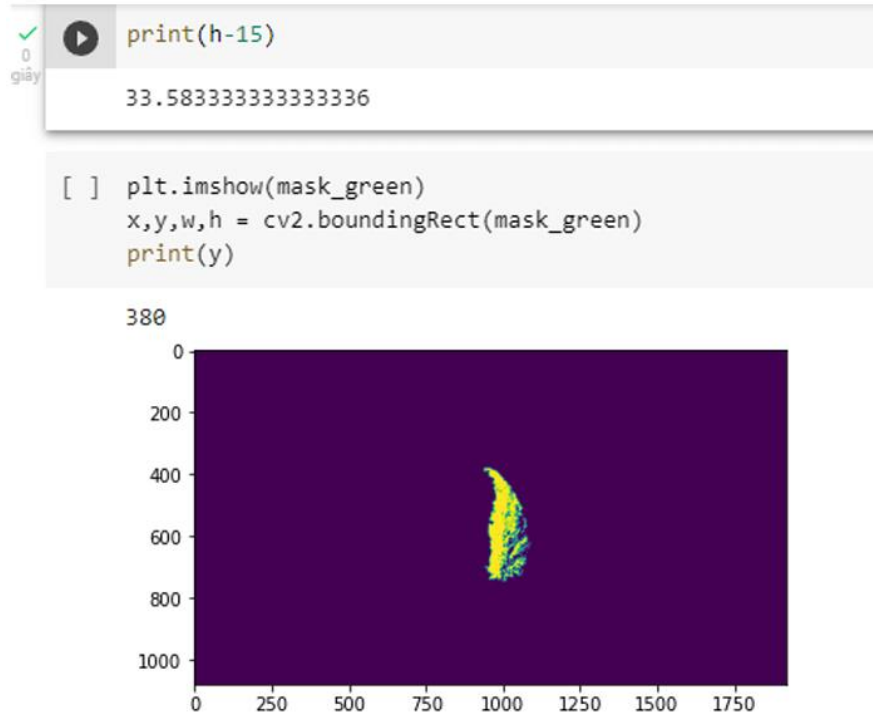


Hình 3.2 Kết quả chiều cao cây lúc 12h

Hình 3.1 và 3.2 là ảnh cái cây được chụp vào lúc 12h và kết quả của nó là 33.75cm.



Hình 3.3 Cây lúc 13h



Hình 3.4 Kết quả chiều cao cây lúc 13h

Hình 3.3 và 3.4 là ảnh cái cây được chụp vào lúc 13h và kết quả là 33.58cm



Hình 3.5 Cây lúc 15h

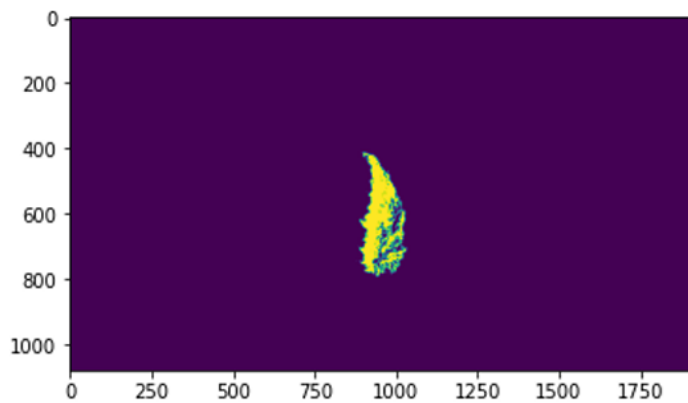
```
print(h-15)
```

```
38.90909090909091
```



```
plt.imshow(mask_green)
x,y,w,h = cv2.boundingRect(mask_green)
print(y)
```

```
412
```

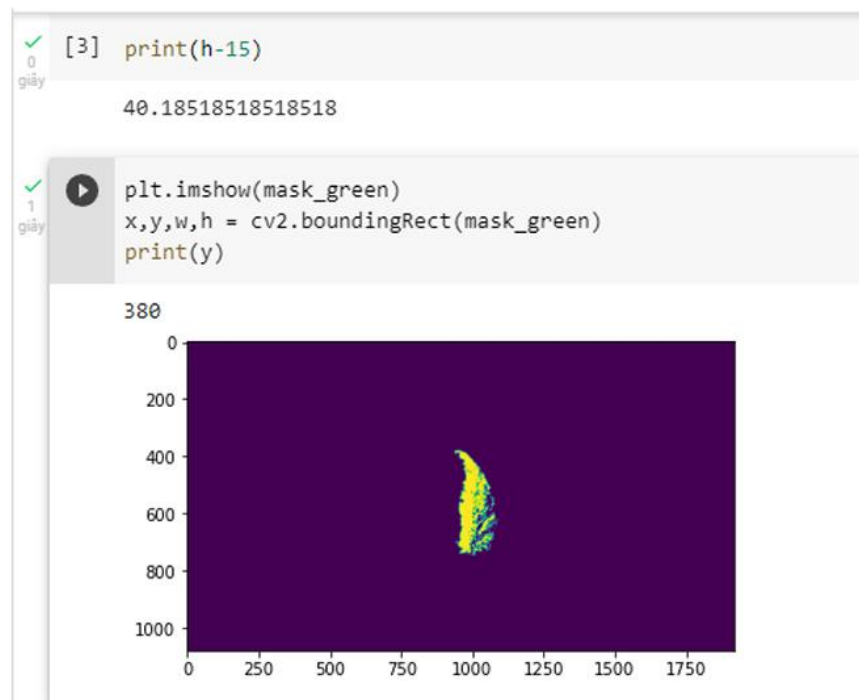


Hình 3.6 Kết quả chiều cao cây lúc 15h

Hình 3.5 và 3.6 là ảnh cái cây được chụp vào lúc 15h và kết quả là 38.9cm.



Hình 3.7 Cây lúc 16h



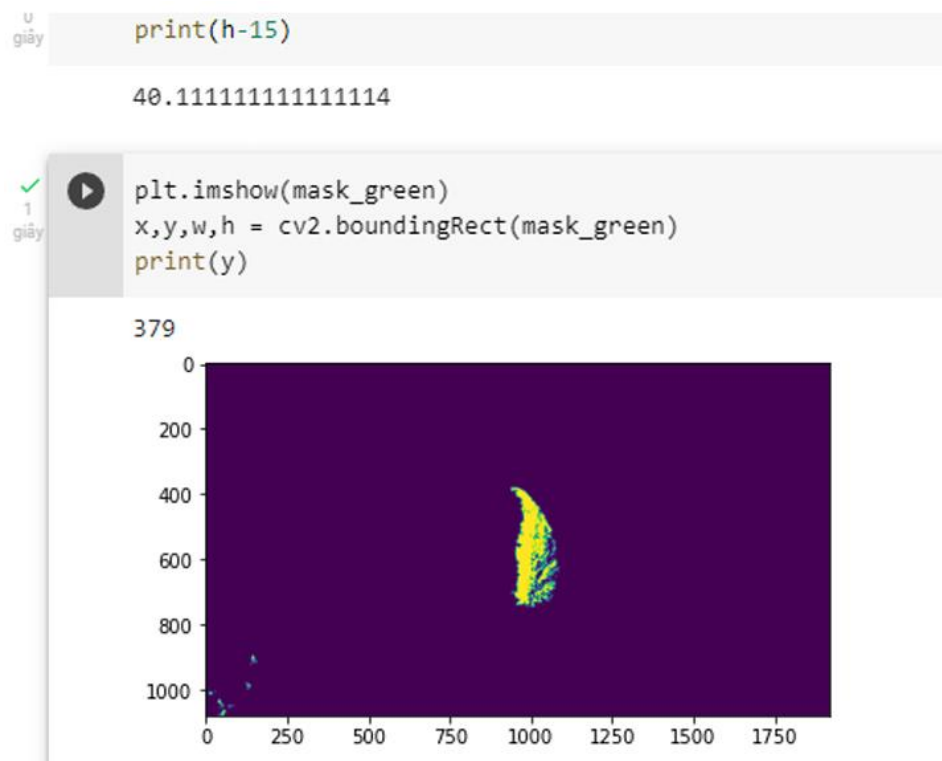
Hình 3.8 Kết quả chiều cao cây lúc 16h

Hình 3.7 và 3.8 là ảnh cái cây được chụp vào lúc 16h và kết quả là 40.18cm.





Hình 3.9 Cây lúc 17h



Hình 3.10 Kết quả chiều cao cây lúc 17h

Hình 3.9 và 3.10 là ảnh cái cây được chụp vào lúc 17h và kết quả là 40.11cm

Theo như những gì kết quả thực nghiệm vừa cho ra được, thì có thể thấy vào lúc 12h và 13h, kết quả cho ra được khá là chênh lệch so với kết quả thực. Còn vào lúc 15h, 16h, 17h thì kết quả cho ra gần đúng và đúng với chiều cao thực. Sở dĩ có những sai số như vậy, là do lúc đặt cam, góc cam bị lệch, nên các ảnh ở 12h và 13h sai số nhiều đến thế.

## Chương 4. TỔNG KẾT ĐỀ TÀI

### 4.1 Kết quả đạt được

#### 4.1.1 Ưu điểm

- Mạch có cấu tạo và hoạt động đơn giản không phức tạp nhưng tính chính xác và khả năng áp dụng thực tế cao, giảm bớt công sức của con người trong việc chăm sóc vườn.
- Dễ lắp ráp đối với người mới tập làm mạch điện tử.
- Các con cảm biến nhiệt độ, độ ẩm đất hoạt động với sai số nhỏ, máy bơm nước hoạt động ổn
- Dữ liệu được cập nhật lên màn hình LCD và Firebase với độ trễ thấp
- Chiều cao cây detect được với sai số  $\leq 1\text{cm}$  trong điều kiện lý tưởng
- Có thể mở rộng và phát triển dễ dàng.

#### 4.1.2 Hạn chế

- Trong điều kiện thiếu ánh sáng detect chiều cao sẽ bị ảnh hưởng nhiều, Bộ dữ liệu được train theo góc cam cố định nên nếu cây ở vị trí khác thì sẽ detect ra chiều cao bị sai. Nếu thay đổi loại cây có chậu cây khác thì phải thay đổi lại code trừ chiều cao chậu cây. Chiều cao cây lớn nhất 90cm nên nếu như cây cao hơn thì phải train thêm dataset



### 4.1.3 Hướng phát triển

- Thêm nhiều tương tác hơn trên Firebase Web chứ không phải chỉ dừng ở mức hiển thị.
- Cải thiện độ chính xác của thuật toán detect chiều cao, tìm cách để có thể cải thiện độ chính xác trong điều kiện thiếu ánh sáng.

### TÀI LIỆU THAM KHẢO

- [1] Thamaraimanalan, T., Vivekk, S. P., Satheeshkumar, G., & Saravanan, P. (2018). Smart garden monitoring system using IoT. Asian Journal of Applied Science and Technology (AJAST), 2(2), 186-192.
- [2] Sambath, M., Prasant, M., Raghava, N. B., & Jagadeesh, S. (2019, November). IoT based garden monitoring system. In Journal of Physics: Conference Series (Vol. 1362, No. 1, p. 012069). IOP Publishing.
- [3] Karunakanth, M., Venkatesan, R., & Kathrine, G. J. W. (2018). IOT based smart irrigation system for home based organic garden. International Journal of Pure and Applied Mathematics, 119(12), 16193-16199.
- [4] Kurniawan, D., Putra, R. J., Bella, A., Ashar, M., & Dedes, K. (2021, October). Smart Garden with IoT Based Real Time Communication using MQTT Protocol. In 2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE) (pp. 1-5). IEEE.
- [5] Nath, S. D., Hossain, M. S., Chowdhury, I. A., Tasneem, S., Hasan, M., & Chakma, R. (2021). Design and implementation of an IoT based greenhouse monitoring and controlling system. Journal of Computer Science and Technology Studies, 3(1), 01- 06.

[6] Sowmiya, M., & Prabavathi, S. (2019). Smart agriculture using IoT and cloud computing. *Int. J. Recent Technol. Eng*, 7(6), 251-255.

[7] Singh, S., Iqbal, A., Singh, J., Kumar, R., Yadav, A. K., & Pandey, Y. (2020). Smart garden with iot based plant monitoring system. *Solid State Technology*, 63(4), 2780- 2787.

[8] Wiyono, A. T. (2021). Design of Smart Garden Based On The Internet of Things (IoT). *BEST: Journal of Applied Electrical, Science, & Technology*, 3(2), 36-40.

[9] Khan, R. U., Khan, K., Albattah, W., & Qamar, A. M. (2021). Image-based detection of plant diseases: from classical machine learning to deep learning journey. *Wireless Communications and Mobile Computing*, 2021, 1-13.

[10] Criminisi, A., Zisserman, A., Van Gool, L. J., Bramble, S. K., & Compton, D. (1999, February). New approach to obtain height measurements from video. In *Investigation and Forensic Science Technologies* (Vol. 3576, pp. 227-238). SPIE.

**Cơ quan Chủ trì**

*(ký, họ và tên, đóng dấu)*

**Chủ nhiệm đề tài**

*(ký, họ và tên)*



Nguyễn Minh Trí

## THÔNG TIN KẾT QUẢ NGHIÊN CỨU

### 1. Thông tin chung:

- Tên đề tài: Thiết kế hệ thống tưới nước và theo dõi tăng trưởng chiều cao cây trồng tự động ứng dụng cho nhà ươm thông minh.
- Mã số: 19522388
- Chủ nhiệm: Nguyễn Minh Trí
- Thành viên tham gia: Nguyễn Hữu Thành (MSSV: 19522236)
- Cơ quan chủ trì: Trường Đại học Công nghệ Thông tin.
- Thời gian thực hiện: 6 tháng

### 2. Mục tiêu:

- Lấy được thông tin về nhiệt độ không khí với sai số không quá 2 độ C
- Lấy được thông tin về độ ẩm không khí với sai số không quá 2%
- Lấy được thông tin về độ ẩm đất không qua 2%
- Tưới nước cho cây khi độ ẩm đất giảm qua mức ngưỡng
- Đo được chiều cao cây với sai số không quá 5%

### 3. Tính mới và sáng tạo:

- Đo được chiều cao của cây với sai số không quá 5%

### 4. Tóm tắt kết quả nghiên cứu:

Ở nghiên cứu này, nhóm đã tạo ra 1 bộ dataset về chiều cao cây. Dựa vào dataset đã có sẵn, nhóm đã chụp hình cây thật, và đối chiếu với bộ dataset trước đó để lấy được chiều cao cây thực tế.

Bên cạnh đó, những thông tin về nhiệt độ, độ ẩm không khí, độ ẩm đất, nhóm đã lấy được những thông tin đó với sai số không quá cao

Nhóm cũng đã đưa được những thông tin về nhiệt độ, độ ẩm không khí, độ ẩm đất, chiều cao cây lên giao diện Firebase web

## **5. Tên sản phẩm:**

Thiết kế hệ thống tưới nước và theo dõi tăng trưởng chiều cao cây trồng tự động ứng dụng cho nhà ươm thông minh.

## **6. Hiệu quả, phương thức chuyển giao kết quả nghiên cứu và khả năng áp dụng:**

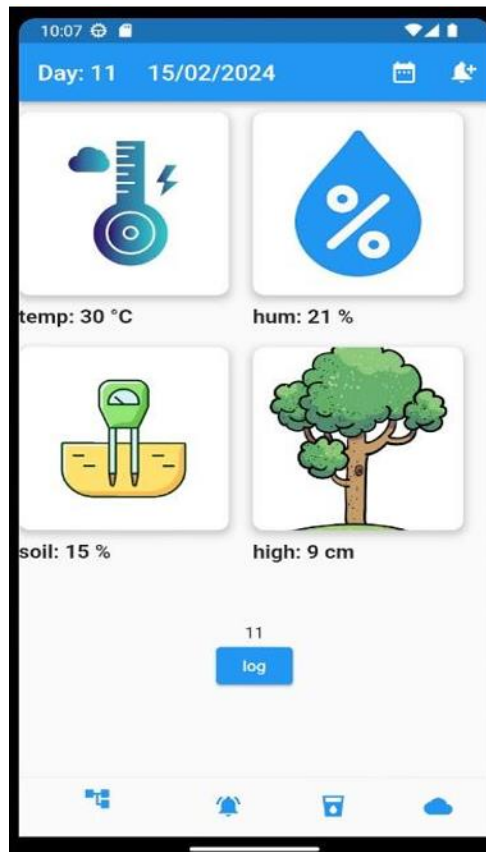
Nghiên cứu này có thể áp dụng trong các nhà ươm thông minh, khi nó có thể tự động tưới nước khi độ ẩm đất giảm qua mức ngưỡng.

Nghiên cứu này cập nhật được thông tin về môi trường xung quanh cây trồng như nhiệt độ, độ ẩm không khí, độ ẩm đất. Những thông tin này được hiển thị lên App, giúp người dùng tiện hơn trong việc nắm được thông tin môi trường xung quanh cây.

Nghiên cứu này đo được chiều cao cây trồng, qua đó giúp cho người dùng nắm được quá trình sinh trưởng của cây. Dữ liệu này cũng được cập nhật trên app, giúp người dùng nắm thông tin một cách tiện lợi hơn.

## **7. Hình ảnh, sơ đồ minh họa chính**

### **7.1 Giao diện App**



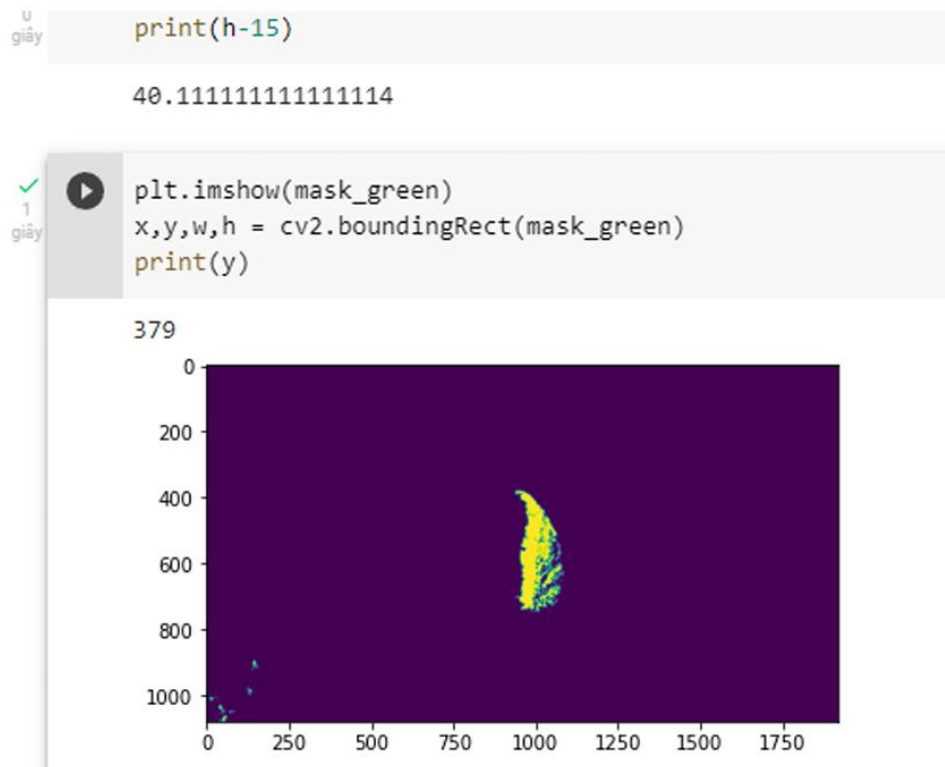
*Hình 1 Giao diện Flutter App*

Hình 1 phía trên là giao diện của App mà nhóm đã làm ra. Nó chứa những thông tin về chiều cao cây, nhiệt độ, độ ẩm không khí, độ ẩm đất. Những dữ liệu đó sẽ được cập nhật theo thời gian.

## 7.2 Đo chiều cao cây



*Hình 2 Ảnh cây gốc*



Hình 3 Kết quả đo chiều cao cây

Hai hình 7.2 và 7.3 phía trên lần lượt là ảnh gốc của cây và kết quả sau khi đo được chiều cao cây.

Ở hình 7.3, sau khi cây được chụp, nó sẽ được lọc màu, sau đó detect đỉnh cây, lấy được thông số của đỉnh cây đó. Sau khi có được thông số, nhóm sẽ so sánh thông số đó với bộ dataset mà nhóm đã chuẩn bị trước đó. Sau khi so sánh xong, nó sẽ cho ra chiều cao cây. Kết quả chiều cao cây sau khi trừ đi chiều cao của bình là 40.11 cm.

**Cơ quan Chủ trì**  
(ký, họ và tên, đóng dấu)

**Chủ nhiệm đề tài**  
(ký, họ và tên)

Nguyễn Minh Trí





## THUYẾT MINH ĐỀ TÀI KHOA HỌC VÀ CÔNG NGHỆ CẤP SINH VIÊN 2021

### A. THÔNG TIN CHUNG

#### A1. Tên đề tài

- Tên tiếng Việt (IN HOA):

“THIẾT KẾ HỆ THỐNG TƯỚI NƯỚC VÀ THEO DÕI TĂNG TRƯỞNG CHIỀU CAO CÂY TRỒNG TỰ ĐỘNG ỨNG DỤNG CHO NHÀ ƯƠM THÔNG MINH”

- Tên tiếng Anh (IN HOA):

“DESIGN AN AUTOMATIC WATERING AND HEIGHT TRACKING SYSTEM FOR SMART GREEN HOUSE”

#### A2. Thời gian thực hiện

06 tháng (kể từ khi được duyệt).

#### A3. Tổng kinh phí

Tổng kinh phí: 6 triệu đồng, gồm

- Kinh phí từ Trường Đại học Công nghệ Thông tin: 6 triệu đồng

#### A4. Chủ nhiệm

Họ và tên: Nguyễn Minh Trí

Ngày, tháng, năm sinh: 17/10/2001

. Giới tính (Nam/Nữ): Nam.

Số CMND: 048201006101 ; Ngày cấp: 10/8/2021 ; Nơi cấp: Đà Nẵng

Mã số sinh viên: 19522388

Số điện thoại liên lạc: 0905163201

Đơn vị (Khoa): Kỹ thuật máy tính

Số tài khoản: 103869367650

Ngân hàng: Vietinbank

#### A5. Thành viên đề tài

TT	Họ tên	MSSV	Khoa
1	Nguyễn Hữu Thành	19522236	Kỹ thuật máy tính
2			



## **B. MÔ TẢ NGHIÊN CỨU**

### **B1. Giới thiệu về đề tài**

#### **B1.1 Lời nói đầu**

Trong bối cảnh thay đổi khí hậu và quá trình đô thị hóa ngày càng gia tăng, việc trồng cây trở nên ngày càng khó khăn hơn. Để giải quyết vấn đề này, nhiều nỗ lực đã được đưa ra để phát triển các giải pháp trồng cây thông minh nhằm tăng cường năng suất và giảm tác động tiêu cực đến môi trường.

Trong đó, hệ thống Smart Garden được xem là một giải pháp hiệu quả để giúp người trồng cây quản lý và chăm sóc cây trồng một cách thông minh. Hệ thống này sử dụng các công nghệ mới như cảm biến, hệ thống tự động và kết nối mạng để giám sát và điều khiển các yếu tố môi trường cần thiết cho việc trồng cây. Điều này giúp người trồng cây dễ dàng điều chỉnh và tối ưu hóa các thông số như nhiệt độ, độ ẩm, ánh sáng và nước, từ đó giúp cây trồng phát triển một cách tốt nhất.

Một số công trình nghiên cứu về Smart Garden:

➤ Nghiên cứu của tác giả Thamaraيمانalan T. và các cộng sự giới thiệu hệ thống giám sát vườn thông minh sử dụng công nghệ Internet of Things (IoT) để giám sát và duy trì sức khỏe của các loại cây trong vườn [1].

➤ Tác giả Sambath M. và các cộng sự giới thiệu một hệ thống giám sát vườn ươm sử dụng công nghệ Internet of Things (IoT) để giám sát và điều khiển các thông số khác nhau trong vườn, chẳng hạn như nhiệt độ, độ ẩm, độ ẩm đất và độ sáng [2].

➤ Nghiên cứu [3] mô tả một hệ thống tưới thông minh sử dụng công nghệ IoT (Internet of Things) cho vườn rau hữu cơ gia đình. Hệ thống được thiết kế để cải thiện hiệu quả sử dụng nước và giảm lãng phí nước. Bài báo cung cấp một mô tả chi tiết về kiến trúc, các thành phần và hoạt động của hệ thống.

➤ Nghiên cứu [4] mô tả về Ứng dụng IoT vào 1 khu vườn nhỏ tại đô thị. Hệ thống quan sát và ghi nhận môi trường của vườn, giúp cho người dùng tiện trong việc theo dõi.

Trong đề tài này, chúng tôi sẽ tập trung vào việc phát triển hệ thống Smart Garden kết hợp theo dõi quá trình sinh trưởng của cây thông qua chiều cao, một yếu tố quan trọng trong việc đánh giá và quản lý sức khỏe của cây trồng. Chúng tôi hy vọng rằng nghiên cứu của chúng tôi sẽ giúp tăng cường hiệu quả của việc trồng cây thông minh và đóng góp tích cực vào sự phát triển bền vững của ngành nông nghiệp.

#### **B1.2 Tổng quan tình hình nghiên cứu liên quan đến đề tài**

Hiện nay, có rất nhiều nghiên cứu được thực hiện liên quan đến đề tài Smart Garden. Dưới đây là một số nghiên cứu được thực hiện trong lĩnh vực này:

1. Tích hợp hệ thống tưới thông minh vào Smart Garden: Nghiên cứu này tập trung vào việc thiết kế và phát triển hệ thống tưới thông minh cho Smart Garden. Nghiên cứu sử dụng các cảm biến để giám sát độ ẩm đất và ánh sáng, và kết hợp với hệ thống tưới tự động để tưới cây trồng đúng lượng nước cần thiết.

Các công trình nghiên cứu liên quan:

➤ Nghiên cứu [5] trình bày một hệ thống tưới thông minh dựa trên IoT (Internet of Things) và kỹ thuật máy học để giám sát, phân tích và điều khiển nước tưới cho cây trồng.

➤ Nghiên cứu [6] là một nghiên cứu về việc tích hợp công nghệ IoT và giao thức MQTT vào hệ thống tưới thông minh trong vườn. Đề tài này tập trung vào việc thiết kế và triển khai một hệ thống tưới thông minh với khả năng theo dõi thời gian thực và giao tiếp thông minh bằng cách sử dụng giao thức MQTT. 2. Điều khiển và giám sát môi trường phát triển cây trồng trong Smart Garden: Nghiên cứu này tập trung vào việc điều khiển và giám sát môi trường phát triển của cây trồng trong Smart Garden bằng cách sử dụng các cảm biến để giám sát độ ẩm đất, nhiệt độ, độ pH và độ sáng. Hệ thống sử dụng các thuật toán để điều khiển các thông số này để đảm bảo cây trồng phát triển tốt nhất. Các công trình nghiên cứu liên quan:

➤ Bài báo [7] này giới thiệu một hệ thống giám sát và điều khiển nhà kính thông minh sử dụng các cảm biến để thu thập dữ liệu về nhiệt độ, độ ẩm và ánh sáng, và các thiết bị điều khiển để kiểm soát các thông số này.

➤ Nghiên cứu [8] đã thiết kế và triển khai một hệ thống tự động giám sát các thông số như độ ẩm đất, nhiệt độ, độ ẩm không khí, ánh sáng, độ pH của đất bằng cách sử dụng các cảm biến và thiết bị điện tử. 3. Ứng dụng công nghệ IoT (Internet of Things) vào Smart Garden: Nghiên cứu này tập trung vào việc sử dụng các thiết bị IoT để giám sát và điều khiển các thông số môi trường trong Smart Garden. Các thiết bị IoT được kết nối với mạng internet để giám sát từ xa, và sử dụng các thuật toán để tự động điều chỉnh các thông số môi trường để cây trồng phát triển tốt nhất. Các công trình nghiên cứu và giải pháp:

➤ Nghiên cứu [9] mô tả việc phát triển một hệ thống khu vườn thông minh sử dụng công nghệ Internet of Things (IoT) để giám sát quá trình phát triển của cây trồng.

➤ Nghiên cứu [10] thảo luận về thiết kế một vườn thông minh sử dụng các cảm biến, vi điều khiển và IoT để kiểm soát và giám sát điều kiện môi trường trong vườn. 4. Ứng dụng trí tuệ nhân tạo (AI) vào Smart Garden: Nghiên cứu này tập trung vào việc sử dụng các thuật toán trí tuệ nhân tạo để phân tích dữ liệu từ các cảm biến và giúp đưa ra các quyết định điều khiển môi trường phát triển cây trồng trong Smart Garden. Hệ thống sử dụng các thuật toán học máy để học từ dữ liệu và cải thiện hiệu suất của hệ thống. Các công trình nghiên cứu và giải pháp:

➤ Nghiên cứu [11] mô tả việc ứng dụng Machine Learning vào việc nhận diện lá hư hại trong việc giám sát cây trồng. Tổng quan tình hình nghiên cứu liên quan đến đề tài Smart Garden

cho thấy rằng đây là một lĩnh vực đầy tiềm năng với rất nhiều khả năng ứng dụng thực tế. Các nghiên cứu này cũng đã đưa ra nhiều giải pháp và nghiên cứu hữu ích để phát triển các hệ thống Smart Garden thông minh hơn, giúp tăng năng suất và giảm thiểu tác động đến môi trường.

Ngoài ra, các nghiên cứu này cũng đã đưa ra một số thách thức cần được giải quyết để phát triển các hệ thống Smart Garden hiệu quả hơn. Các thách thức này bao gồm việc tích hợp các hệ thống cảm biến và điều khiển môi trường phức tạp, đảm bảo tính ổn định và độ tin cậy của hệ thống, và đảm bảo tính bảo mật và an toàn của dữ liệu.

Tóm lại, các nghiên cứu liên quan đến đề tài Smart Garden đã đưa ra rất nhiều kết quả quan trọng và hữu ích để phát triển các hệ thống Smart Garden thông minh hơn. Tuy nhiên, còn rất nhiều thách thức cần được giải quyết để đưa các hệ thống này vào ứng dụng thực tế một cách hiệu quả.

### **B1.3 Lý do thực hiện đề tài**

Trong thời đại hiện nay, việc phát triển giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao có ý nghĩa rất quan trọng. Một trong những lý do quan trọng nhất là xu hướng phát triển phổ biến của mô hình trồng trọt trong nhà kín. Theo báo cáo của AgFunder News, thị trường trồng trọt trong nhà kín dự kiến sẽ đạt 12,77 tỷ USD vào năm 2026, với tỷ lệ tăng trưởng hàng năm đạt 24,8%. Bên cạnh đó, theo bài báo của Nikkei Asia, mô hình trồng trọt trong nhà kín đang ngày càng được áp dụng rộng rãi tại nhiều nước nhằm giải quyết các vấn đề liên quan đến đại dịch COVID-19.

Lý do thứ hai là các chính sách của nhà nước về phát triển nông nghiệp tự động, nông nghiệp thông minh, ứng dụng công nghệ vào nông nghiệp. Theo Luật Nông nghiệp 2018 và Nghị định số 57/2020/NĐ-CP của Chính phủ, việc ứng dụng khoa học công nghệ trong sản xuất nông nghiệp sẽ giúp nâng cao năng suất, chất lượng và giá trị sản phẩm nông nghiệp. Đồng thời, việc phát triển giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao cũng giúp giảm thiểu tác động tiêu cực đến môi trường.

Lý do cuối cùng là việc tưới tiêu hợp lý và xác định chiều cao cây trồng trong quá trình chăm sóc cây là một yếu tố rất quan trọng để đảm bảo cho cây trồng phát triển tốt nhất có thể. Theo bài báo của AgFunder News, việc sử dụng phương pháp tưới tiêu chính xác và phân bón giúp tăng cường sức khỏe và sự phát triển của cây trồng. Đồng thời, theo bài báo của ScienceDirect, việc đo chiều cao cây trồng là một phương pháp thường được sử dụng để đánh giá sinh trưởng của cây trồng.

Vì vậy, việc phát triển giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao không chỉ đáp ứng các yêu cầu của xu hướng trồng trọt trong nhà kín và các chính sách hỗ trợ phát triển nông nghiệp thông minh, mà còn giúp giải quyết một số vấn đề đang tồn tại trong sản xuất nông nghiệp.

Trước đây, việc đo đạc chiều cao cây trồng và tưới tiêu thường được thực hiện thủ công, dẫn đến sai sót và mất thời gian. Tuy nhiên, với giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao, các dữ liệu liên quan đến chiều cao cây và sinh trưởng của chúng được thu thập và phân tích một cách chính xác và nhanh chóng.

Ngoài ra, việc tưới tiêu hợp lý và xác định chiều cao cây trồng cũng đóng vai trò quan trọng trong việc giảm thiểu lượng nước và phân bón được sử dụng, từ đó giảm thiểu tác động tiêu cực đến môi trường. Điều này rất quan trọng đặc biệt khi nguồn nước và đất đai ngày càng khan hiếm và quý giá hơn.

Trên cơ sở đó, giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao không chỉ giúp tăng hiệu quả sản xuất và nâng cao năng suất, mà còn đóng góp tích cực vào bảo vệ môi trường và tài nguyên thiên nhiên.

Tóm lại, việc phát triển giải pháp theo dõi chiều cao cây trồng tự động kết hợp theo dõi sinh trưởng theo chiều cao là một bước tiến quan trọng trong sản xuất nông nghiệp hiện đại, giúp giảm thiểu sai sót, tiết kiệm thời gian và tài nguyên, đồng thời giảm thiểu tác động tiêu cực đến môi trường.

#### **B1.4 Các thách thức**

Trong quá trình triển khai đề tài Smart Garden, có một số thách thức cần phải được giải quyết để đảm bảo hiệu quả và tính ứng dụng của hệ thống. Dưới đây là một số thách thức phổ biến:

1. Đảm bảo tính ổn định và độ tin cậy của hệ thống: Hệ thống Smart Garden phải hoạt động liên tục trong một khoảng thời gian dài, do đó đảm bảo tính ổn định và độ tin cậy của hệ thống là một thách thức quan trọng. Mất điện hoặc sự cố kỹ thuật có thể gây ra nguy hiểm cho cây trồng.

2. Tích hợp các hệ thống cảm biến và điều khiển môi trường: Hệ thống Smart Garden sử dụng nhiều cảm biến để giám sát các thông số môi trường như nhiệt độ, độ ẩm, ánh sáng và độ pH. Tuy nhiên, tích hợp các cảm biến này và điều khiển môi trường sao cho phù hợp với các yêu cầu của các loại cây trồng khác nhau là một thách thức khó khăn.

3. Giải quyết vấn đề văn hóa và thói quen truyền thống: Trong một số nền văn hóa, trồng cây và chăm sóc vườn hoa được coi là một hoạt động thú vị và tạo ra niềm đam mê cho các nhà vườn. Tuy nhiên, việc sử dụng hệ thống Smart Garden có thể đòi hỏi thay đổi thói quen và phong cách truyền thống này, và đó là một thách thức khó khăn cần phải được giải quyết.

4. Phát triển các phần mềm và các thuật toán phù hợp: Hệ thống Smart Garden cần phải sử dụng các phần mềm và thuật toán để điều khiển và giám sát môi trường phát triển cây trồng. Phát triển các phần mềm và thuật toán phù hợp để đảm bảo hiệu quả của hệ thống là một thách thức khó khăn.

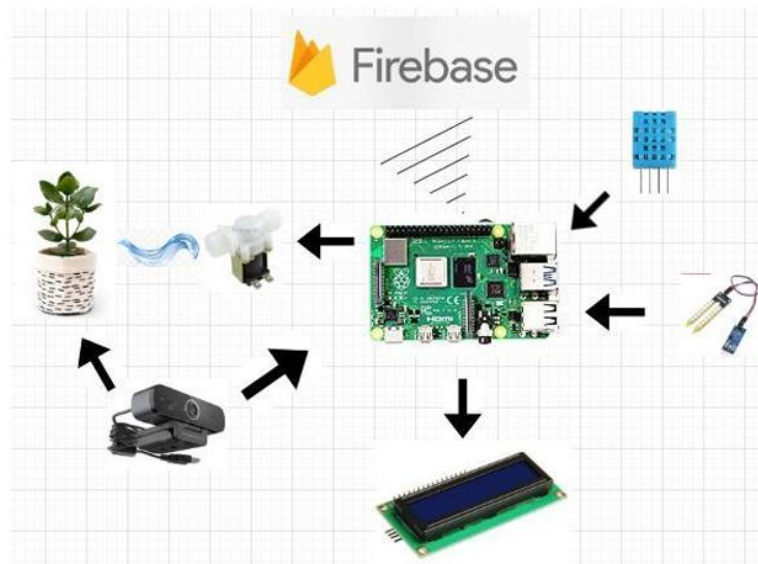
## B2. Mục tiêu, nội dung, kế hoạch nghiên cứu

### B2.1 Mục tiêu

Xây dựng hệ thống kiểm soát tưới nước và theo dõi phát triển chiều cao cây trồng tự động theo một kịch bản chăm sóc cụ thể. Hệ thống có thể được tích hợp vào các mô hình tổ chức vườn ươm thông minh, trồng cây trong nhà kính.

### B2.2 Nội dung và phương pháp nghiên cứu

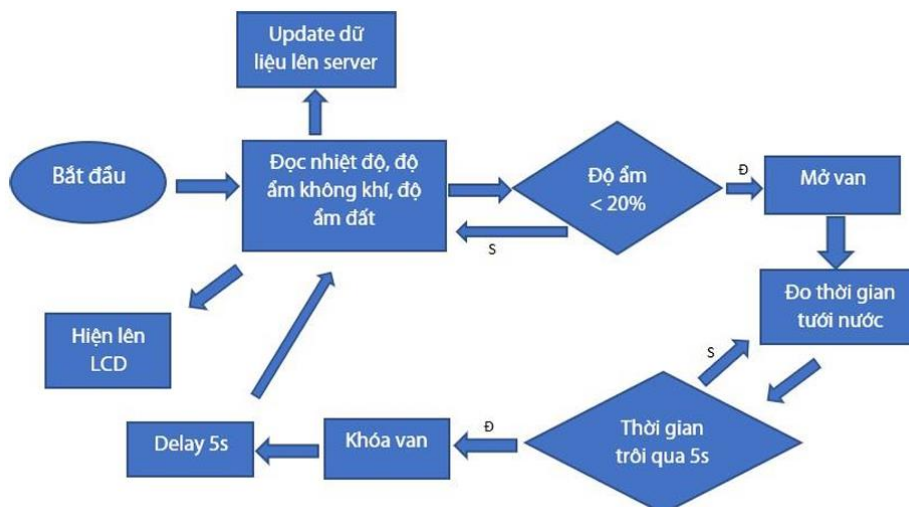
**Nội dung 1:** Xây dựng hệ thống thu thập dữ liệu môi trường và ghi ảnh hiện trạng cây trồng



Hình 1 Sơ đồ khối hệ thống tưới nước tự động

- Các cảm biến gồm: DHT11, Cảm biến độ ẩm đất
- Các thông tin cảm biến sẽ được cập nhật lên Realtime database và màn LCD 2004a

**Nội dung 2:** Xây dựng hệ thống điều khiển tưới nước tự động



Hình 2 Lưu đồ giải thuật hệ thống tưới nước

- Khi độ ẩm đất dưới 20%, Relay bật và máy bơm bắt đầu tưới nước trong 5s rồi relay tắt. Delay thêm 5s để đợi nước ngấm vào đất. Sau đó tiếp tục đo độ ẩm đất để check xem có cần phải tiếp tục tưới thêm nước nữa hay không.

### Nội dung 3: Xây dựng giải pháp xác định chiều cao cây trồng



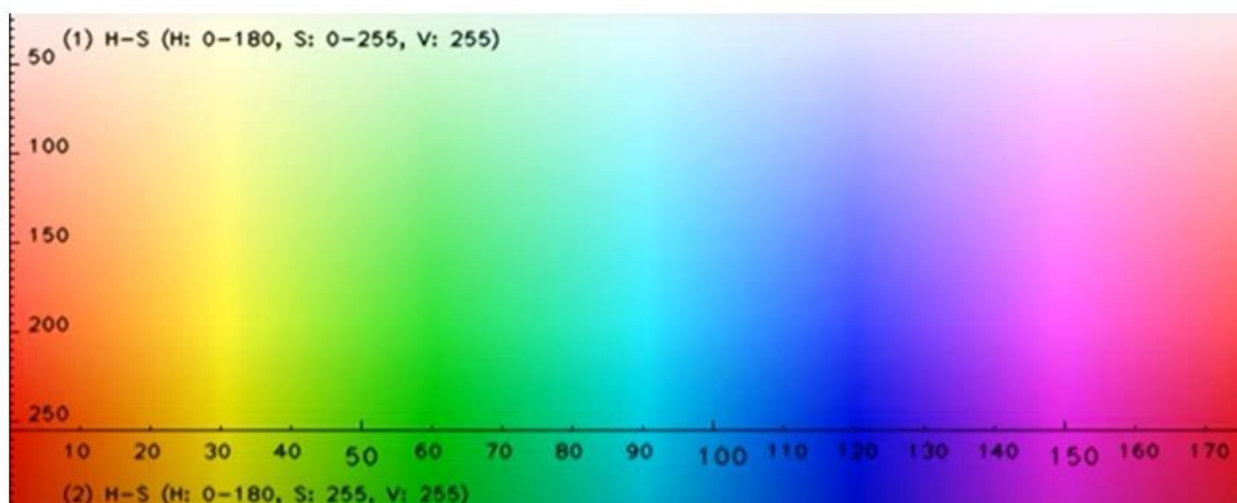
Hình 3 Quy trình nhận diện chiều cao cây

Để xác định chiều cao của cây, hệ thống thực hiện 4 bước như trong hình 3, trong đó:

- Đầu tiên ở bước 1, nhóm sẽ set up 1 góc cam cố định, sau đó đánh dấu từng mốc chiều cao lên tường và để cho camera chụp lại từng bức hình của từng mốc chiều cao đó. Mỗi mốc sẽ có màu đỏ để thuận lợi cho việc detect và sẽ cách nhau khoảng 2cm.

- Sau đó, nhóm sẽ lọc màu từng bức ảnh chứa từng mốc chiều cao để detect từng mốc, sau đó dùng thư viện matplotlib để lấy được thông số pixel theo trục y của từng mốc (thông số y), từng mốc sẽ có 1 thông số y riêng, thu thập được thông số y của từng mốc chiều cao, ta có được được bộ dữ liệu chiều cao.

- Tiếp theo ở bước 2, nhóm sẽ chụp hình cây và cũng tiến hành lọc màu. Vì cây đa phần sẽ có màu xanh ở lá, nên nhóm sẽ gom lại tất cả màu xanh dựa trên bảng thông số màu sắc ở phía dưới, dựa vào những màu xanh đã được gom lại trước đó, nhóm sẽ detect được phần lớn cái cây, qua đó detect ra được đỉnh cây. Sau đó, nhóm cũng sẽ dùng thư viện matplotlib để lấy được thông số y của đỉnh cây.



Hình 4 Bảng màu HSV

- Ở bước 3, khi lấy được thông số y của đỉnh cây, nhóm tiến hành đối chiếu, so sánh với bộ dữ liệu chiều cao đã được thu thập ở bước 1.

- Cuối cùng ở bước 4, khi so sánh xong, nhóm sẽ đưa ra được chiều cao tương ứng của cây.

#### Nội dung 4: Xây dựng hệ thống lưu trữ và quản lý trên điện toán đám mây

- Xây dựng CSDL sử dụng Firebase: Nhóm tiến hành sử dụng tiện ích realtime database của firebase làm cơ sở dữ liệu để lưu trữ thông tin. Vì mỗi lần đẩy thông tin lên realtime database thì nó sẽ lưu trữ tới nhưng ID token ngẫu nhiên. Nên để có thể dễ dàng truy xuất thông tin trên realtime Database nhóm sẽ tiến hành tạo thêm 1 Authentication Firebase để thông tin có thể lưu trữ vào 1 ID token cố định.

- Xây dựng ứng dụng tương tác trên nền tảng web: Trang Web sẽ được viết dựa trên html/css/js và được deploy bởi Firebase App tương tác nhóm lựa chọn Flutter một Framework được phát triển bởi google có thể chạy trên đa nền tảng và hỗ trợ khá mạnh với Firebase.

#### B2.3 Kế hoạch nghiên cứu.

Nội dung	Tháng					
	1	2	3	4	5	6
Thu thập dữ liệu môi trường xung quanh cây						
Hoàn thiện hệ thống điều khiển tưới nước tự động						
Thu thập dữ liệu các mốc chiều cao						
Đo được và trích xuất chiều cao của cây						
Hoàn thiện đề tài						

#### B3. Kết quả dự kiến

Xây dựng một hệ thống thực nghiệm (demo) cho nghiên cứu với các đặc điểm sau:

+ Sai số đo chiều cao:  $\pm 1\text{cm}$ .

+ Sai số đo nhiệt độ:  $\pm 2^{\circ}\text{C}$ .

+ Sai số đo độ ẩm:  $\pm 5\%$ .

Hoàn chỉnh một báo cáo tổng kết.

#### B4. Tài liệu tham khảo

[1] Thamaraimanalan, T., Vivekk, S. P., Satheeshkumar, G., & Saravanan, P. (2018). Smart garden monitoring system using IoT. Asian Journal of Applied Science and Technology (AJAST), 2(2), 186-192.

[2] Sambath, M., Prasant, M., Raghava, N. B., & Jagadeesh, S. (2019, November). IoT based garden monitoring system. In Journal of Physics: Conference Series (Vol. 1362, No. 1, p. 012069). IOP Publishing.



- [3] Karunakanth, M., Venkatesan, R., & Kathrine, G. J. W. (2018). IOT based smart irrigation system for home based organic garden. International Journal of Pure and Applied Mathematics, 119(12), 16193-16199.
- [6] Kurniawan, D., Putra, R. J., Bella, A., Ashar, M., & Dedes, K. (2021, October). Smart Garden with IoT Based Real Time Communication using MQTT Protocol. In 2021 7th International Conference on Electrical, Electronics and Information Engineering (ICEEIE) (pp. 1-5). IEEE.
- [7] Nath, S. D., Hossain, M. S., Chowdhury, I. A., Tasneem, S., Hasan, M., & Chakma, R. (2021). Design and implementation of an IoT based greenhouse monitoring and controlling system. Journal of Computer Science and Technology Studies, 3(1), 01- 06.
- [8] Sowmiya, M., & Prabavathi, S. (2019). Smart agriculture using IoT and cloud computing. Int. J. Recent Technol. Eng, 7(6), 251-255.
- [9] Singh, S., Iqbal, A., Singh, J., Kumar, R., Yadav, A. K., & Pandey, Y. (2020). Smart garden with iot based plant monitoring system. Solid State Technology, 63(4), 2780- 2787.
- [10] Wiyono, A. T. (2021). Design of Smart Garden Based On The Internet of Things (IoT). BEST: Journal of Applied Electrical, Science, & Technology, 3(2), 36-40.
- [11] Khan, R. U., Khan, K., Albattah, W., & Qamar, A. M. (2021). Image-based detection of plant diseases: from classical machine learning to deep learning journey. Wireless Communications and Mobile Computing, 2021, 1-13.
- [12] Criminisi, A., Zisserman, A., Van Gool, L. J., Bramble, S. K., & Compton, D. (1999, February). New approach to obtain height measurements from video. In Investigation and Forensic Science Technologies (Vol. 3576, pp. 227-238). SPIE.

*Ngày 15 tháng 4 năm 2023*  
**Giảng viên hướng dẫn**  
(Ký và ghi rõ họ tên)



Đoàn Duy

*Ngày 15 tháng 4 năm 2023*  
**Chủ nhiệm đề tài**  
(Ký và ghi rõ họ tên)



Nguyễn Minh Trí



