

TRƯỜNG CAO ĐẲNG THỰC HÀNH FPT



## BÁO CÁO DỰ ÁN 2

# SMART DOOR

Giáo viên hướng dẫn: Nguyễn Thanh Điền

Nhóm: Bee Smart, Lớp: IO14301

Sinh viên thực hiện

Thành viên:

+ Nguyễn Đình Đại Nghĩa, PS10571

+ Trần Thế Pháp, PS08944

+ Trần Tấn Phát, PS10654

+ Nguyễn Quốc Đại, PS10593

*Hồ Chí Minh – 2020*

## MỤC LỤC

LỜI MỞ ĐẦU	TRANG
PHẦN 1 – GIỚI THIỆU ĐỀ TÀI	6
PHẦN 2: - KHẢO SÁT YÊU CẦU	7
2.1 – YÊU CẦU HỆ THỐNG	7
2.2 – CHỨC NĂNG	7
PHẦN 3 – PHÂN TÍCH HIỆN TRẠNG	8
3.1 – ĐIỂM MẠNH	8
3.2 – ĐIỂM YẾU	8
3.3 – CƠ HỘI	8
3.4 – THÁCH THỨC	8
PHẦN 4 – THIẾT KẾ HỆ THỐNG	9
4.1 – MÔ HÌNH HỆ THỐNG	9
4.2 – SƠ ĐỒ MẪU	10
4.3 – CÁCH THỨC THỰC HIỆN CHỨC NĂNG	10
4.4 – SƠ ĐỒ ERD	11
PHẦN 5 – THỰC HIỆN DỰ ÁN	12
5.1 – PHÂN CÔNG CÔNG VIỆC	12
5.2 – CÁC THIẾT BỊ, VẬT LIỆU CẦN CÓ	12
5.3 – DỰ TOÁN CHI PHÍ THIẾT BỊ, VẬT LIỆU	16
5.4 – SOURCE CODE	16
5.5 – THƯ VIỆN	28
5.6 – HÌNH ẢNH THỰC TẾ	29
PHẦN 6 – KIỂM THỬ	30
6.1 – KẾ HOẠCH	30
6.2 – KIỂM THỬ	30
PHẦN 7 – HƯỚNG DẪN TRIỂN KHAI VÀ SỬ DỤNG	31
TÀI LIỆU THAM KHẢO	32

## LỜI MỞ ĐẦU

IoT – The Internet of Things, là một trong những xu hướng công nghệ đứng đầu thế giới hiện nay. Dự tính cuối năm 2020 sẽ có khoảng 29 tỉ thiết bị kết nối IoT và đến năm 2025, con số này lên tới 70 tỉ thiết bị (Theo BI Intelligence). Vậy thì IoT là gì, tại sao nó lại có sức phát triển mạnh mẽ đến vậy?

IoT (The Internet of Things), hay Internet Vạn Vật là thuật ngữ mô tả mạng lưới các sự vật được kết nối Internet, được cung cấp định danh riêng (UIDs) và có khả năng tự động truyền tải dữ liệu qua một mạng mà không cần tương tác giữa con người với con người hoặc con người với máy tính. Một cách ngắn gọn, IoT đơn giản là đem tất cả mọi vật trên thế giới và kết nối chúng với Internet.

“Thing” – sự vật – trong Internet of Things, có thể là một con người với màn hình cấy ghép tim, một động vật trang trại với bộ tiếp sóng chip sinh học, một chiếc xe ô tô tích hợp các cảm biến để cảnh báo lái xe khi lốp quá non – hoặc bất kỳ đồ vật nào do tự nhiên sinh ra hoặc do con người sản xuất ra mà có thể được gán với một địa chỉ IP và được cung cấp khả năng truyền tải dữ liệu qua mạng lưới.

Vấn đề là, con người rất hạn chế về thời gian, sự chú ý và chính xác – nghĩa là con người không được tốt lắm trong việc lưu giữ dữ liệu về mọi thứ trong thế giới. Nếu chúng ta có những chiếc máy tính biết mọi thứ – sử dụng được dữ liệu chúng thu thập mà không cần sự giúp đỡ của con người – thì chúng ta sẽ có thể theo dõi và đếm mọi thứ, điều này sẽ giúp giảm rất lớn sự lãng phí, thất bại và chi phí. Chúng ta sẽ biết khi nào mọi thứ cần thay thế, sửa chữa hoặc phục hồi và liệu chúng còn có thể còn tiếp tục hoạt động hay hoạt động tốt nhất nữa không”.

Sự ra đời của IoT giúp chúng ta giải quyết bài toán đó và giúp cho cuộc sống con người được cải thiện rất nhiều. Bằng cách kết nối mọi vật với internet và cho chúng khả năng tự động thu thập dữ liệu và gửi/nhận dữ liệu thông qua mạng lưới, công việc của con người được cắt giảm đáng kể.

Trong môi trường kinh doanh, IoT giúp cho phần mềm quản lý doanh nghiệp thu thập thêm nhiều dữ liệu, tiếp cận được nhiều khách hàng hơn, tiết kiệm thời gian xử lý hơn. Nhờ các bộ cảm biến gắn ở máy móc và tại điểm bán hàng, IoT trực tiếp thu thập dữ liệu và tự động đưa lên ERP. Doanh nghiệp có thể kiểm soát quy trình hoạt động sản xuất cũng như nhu cầu thị trường, từ đó có thể xác định và giải quyết các vấn đề gây ảnh hưởng và đưa ra quyết định nhanh hơn.

## NHẬN XÉT CỦA GIẢNG VIÊN

This image shows a full page of a document template designed for handwritten notes or essays. It features approximately 28 evenly spaced, thin grey horizontal lines across the entire width of the page. The margins are consistent on all sides, providing ample space for writing. There are no pre-printed questions, headings, or other markings on the page.

Giáo viên hướng dẫn ký, ghi rõ họ tên

## NHẬN XÉT CỦA HỘI ĐỒNG PHẢN BIỆN

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

HĐ phản biện ký, ghi rõ họ tên

## PHẦN 1 – GIỚI THIỆU ĐỀ TÀI

Hệ sinh thái IoT luôn được quan tâm cho tới hiện nay chính là mô hình Smart Home – Ngôi nhà thông minh. Một số ứng dụng IoT đang được phổ biến rộng rãi để nâng cao chất lượng cuộc sống cũng như cải thiện chất lượng quản lý của mọi thành viên trong gia đình đối với ngôi nhà của mình:

- Camera an ninh: giám sát ngôi nhà và kiểm soát thông minh bằng AI.
- Bãi đỗ xe thông minh: giám sát không gian đỗ xe của ngôi nhà, kiểm soát và thông báo các vị trí đỗ còn trống.
- Hệ thống nước thông minh: giám sát nhu cầu dùng, kịp thời phát hiện rò rỉ, thất thoát nước trong nhà.
- Hệ thống năng lượng thông minh: giám sát và quản lý tình hình tiêu thụ năng lượng
- Trồng cây và chăm sóc thú cưng thông minh: tự động tưới nước, cho ăn, kiểm soát nhiệt độ,..., giám sát, quản lý từ xa
- Báo động thông minh: theo dõi các mức bức xạ, khí nguy hiểm, cháy nổ để kịp thời cập nhập và kiểm soát tình hình.

Từ những điều tuyệt vời của IOT – Smart Home có thể đem lại cho chúng ta như vậy, chúng em đã đề ra rất nhiều ý tưởng để triển khai một dự án liên quan đến nó như là: Smart key, Smart Garden, Smart Care,... dựa trên nền tảng IOT.

Một dự án Smart Home có rất nhiều chức năng và tiện ích khác nhau, nhưng con người chỉ ấn tượng thứ đầu tiên đập vào mắt họ, thứ đầu tiên cũng là thứ khá quan trọng mà trước khi chúng ta bước vào ngôi nhà để hưởng thụ những điều tuyệt vời ấy, đó chính là cánh cửa, cánh cửa mở ra thế giới thần tiên. Vì lẽ đó, chúng em đã thống nhất ý tưởng, lên kế hoạch và triển khai một dự án đó chính là mô hình ***“Smart Door”***.

## **PHẦN 2 – KHẢO SÁT YÊU CẦU**

### **2.1/ Yêu cầu của hệ thống:**

Vấn đề an ninh tại các khu vực chung cư hiện nay rất đáng báo động, nhiều tội phạm trộm cướp lựa chọn khu vực chung cư là đích đến. Nhiều vụ án mất trộm được phát hiện do kẻ gian phá khóa cửa đột nhập vào khu nhà. Trong các vụ án được khám phá gần đây, nhiều kẻ gian đã khai lý do chọn chung cư làm nơi đột nhập vì ít khi bị người lạ “ngáng đường”. Nếu trộm cắp tại xóm trọ, khu dân cư phải bố trí người canh giới để tránh bị hàng xóm phát hiện thì ở đây chúng lọt vào căn hộ là có thể thoải mái lục soát.

Nguyên nhân việc trộm cắp dễ dàng phá khóa cửa xuất phát từ việc nhà đầu tư muốn cắt giảm chi phí xây dựng nên chỉ lắp đặt các loại cửa, khóa rẻ tiền, dễ bị cạy phá.

Do đó để bảo vệ an toàn cho khu nhà, bảo vệ tài sản của mình thì chỉ trang bị cửa và khóa thật tốt là không đủ. Từ nhu cầu đó, cánh cửa thông minh “**Smart Door**” ra đời, đáp ứng hầu hết nhu cầu của khách hàng.

### **2.2/ Chức năng:**

#### **❖ Chức năng hệ thống:**

- \_ Nhận diện khuôn mặt
- \_ Quẹt thẻ
- \_ Nhập mã số
- \_ Quản lý qua Web app
- \_ Nút nhấn

#### **❖ Cửa sẽ mở khi:**

- \_ Nhận diện được khuôn mặt của gia chủ thông qua camera.
- \_ Gia chủ nhập đúng mã số.
- \_ Gia chủ quẹt đúng thẻ từ đã được cung cấp.
- \_ Sử dụng Web app.
- \_ Nút nhấn để có thể mở cửa từ bên trong

#### **❖ Hệ thống báo động sẽ hoạt động khi:**

- \_ Nhập sai mật mã quá 3 lần.

## PHẦN 3 – PHÂN TÍCH HIỆN TRẠNG

### 3.1/ Điểm mạnh:

- \_ Hệ thống triển khai dựa trên nền tảng IOT hiện đại.
- \_ Sử dụng hệ thống nhận diện khuôn mặt tiên tiến.
- \_ Nhiều chức năng bảo mật: Nhận diện khuôn mặt, mã số, quét thẻ.
- \_ Quản lý thông qua Web app nhanh chóng, tiện lợi.
- \_ Hệ thống báo động luôn trong trạng thái sẵn sàng.
- \_ Bắt kịp xu thế, phù hợp với nhu cầu của người dùng và giải quyết được các bất cập của xã hội về an toàn an ninh.

### 3.2/ Điểm yếu:

- \_ Nhận diện khuôn mặt chưa thật sự mượt mà.
- \_ Thiết bị mua với giá rẻ và kém chất lượng.
- \_ Còn chưa có chức năng tiết kiệm năng lượng.
- \_ Chưa có chức năng thông báo trực tiếp đến điện thoại của gia chủ.

### 3.3/ Cơ hội:

Vấn đề an ninh tại các khu vực chung cư hiện nay rất đáng báo động, nhiều tội phạm trộm cướp lựa chọn khu vực chung cư là đích đến. Nhiều vụ án mất trộm được phát hiện do kẻ gian phá khóa cửa đột nhập vào khu nhà. Do đó để bảo vệ an toàn cho khu nhà, bảo vệ tài sản của mình thì chỉ trang bị cửa và khóa thật tốt là không đủ. Từ nhu cầu đó, cánh cửa thông minh **“Smart Door”** ra đời, đáp ứng hầu hết nhu cầu của khách hàng.

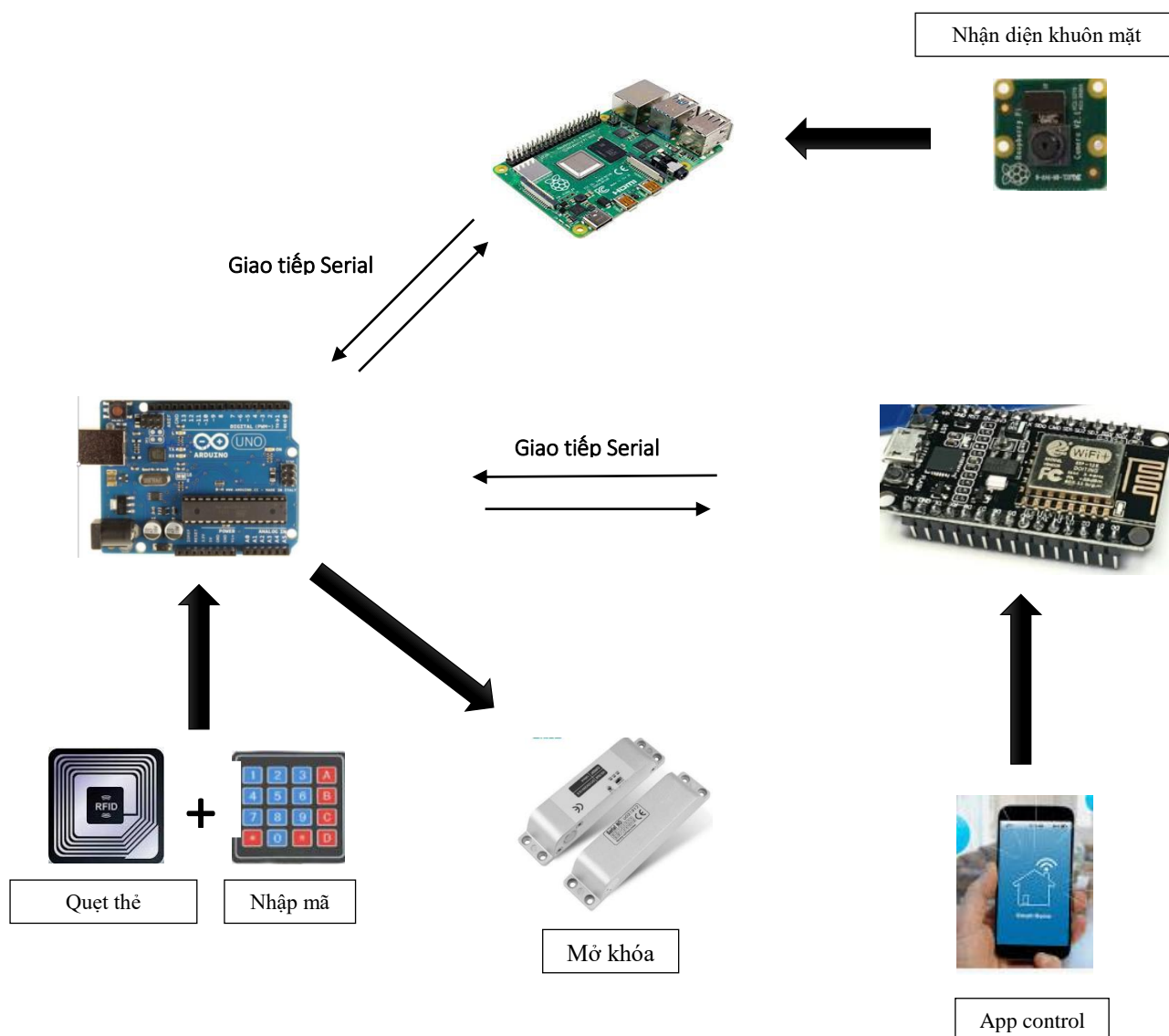
### 3.4/ Thách thức:

- \_ Hiện nay ở Việt Nam có rất nhiều công ty về IOT nên việc cạnh tranh rất gay gắt.
- \_ Khách hàng ngày càng khó tính hơn, yêu cầu ngày càng cao nên phải thường xuyên cập nhật và đổi mới công nghệ.
- \_ Cần tính toán kỹ càng về giá cả và nguồn nguyên vật liệu.

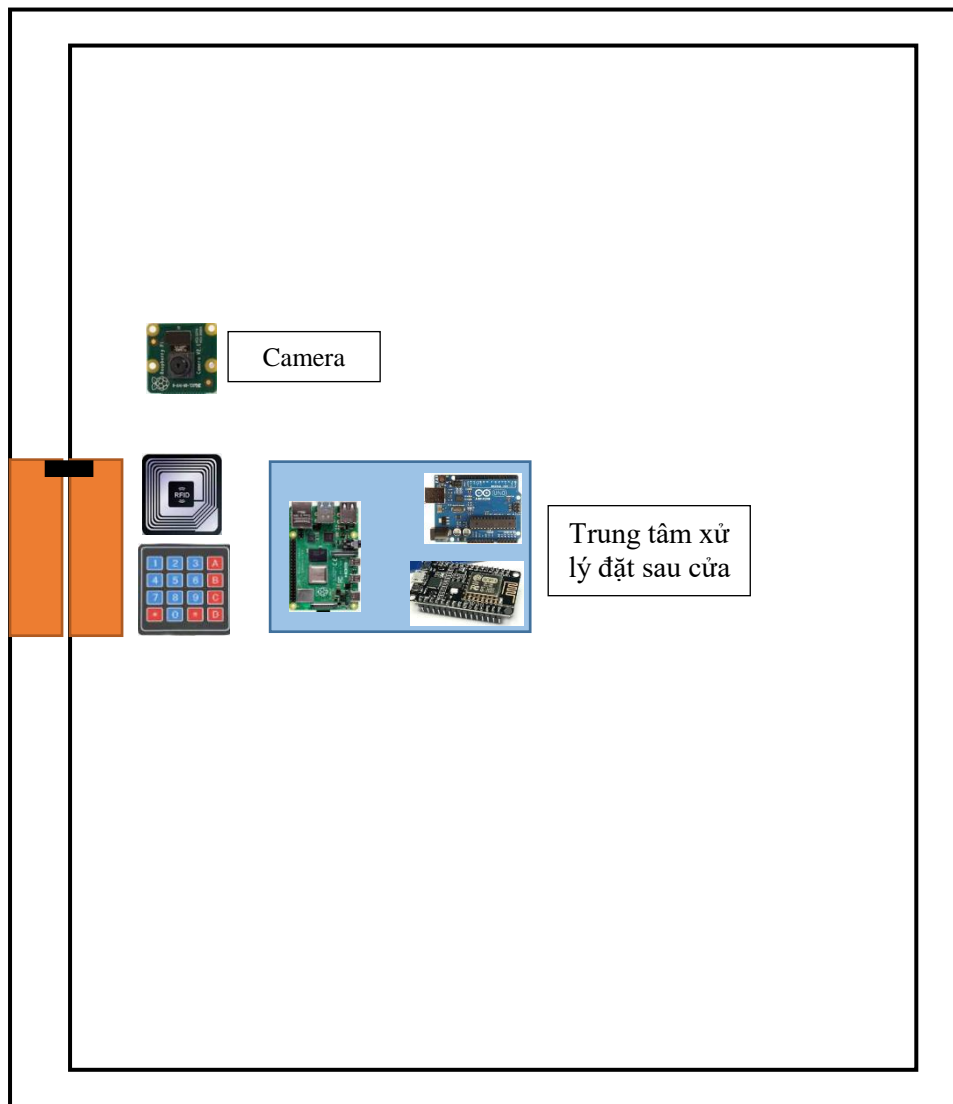


## PHẦN 4 – THIẾT KẾ HỆ THỐNG

### 4.1/ Mô hình hệ thống:



## 4.2/ Sơ đồ mẫu:



## 4.3/ Cách thức thực hiện chức năng:

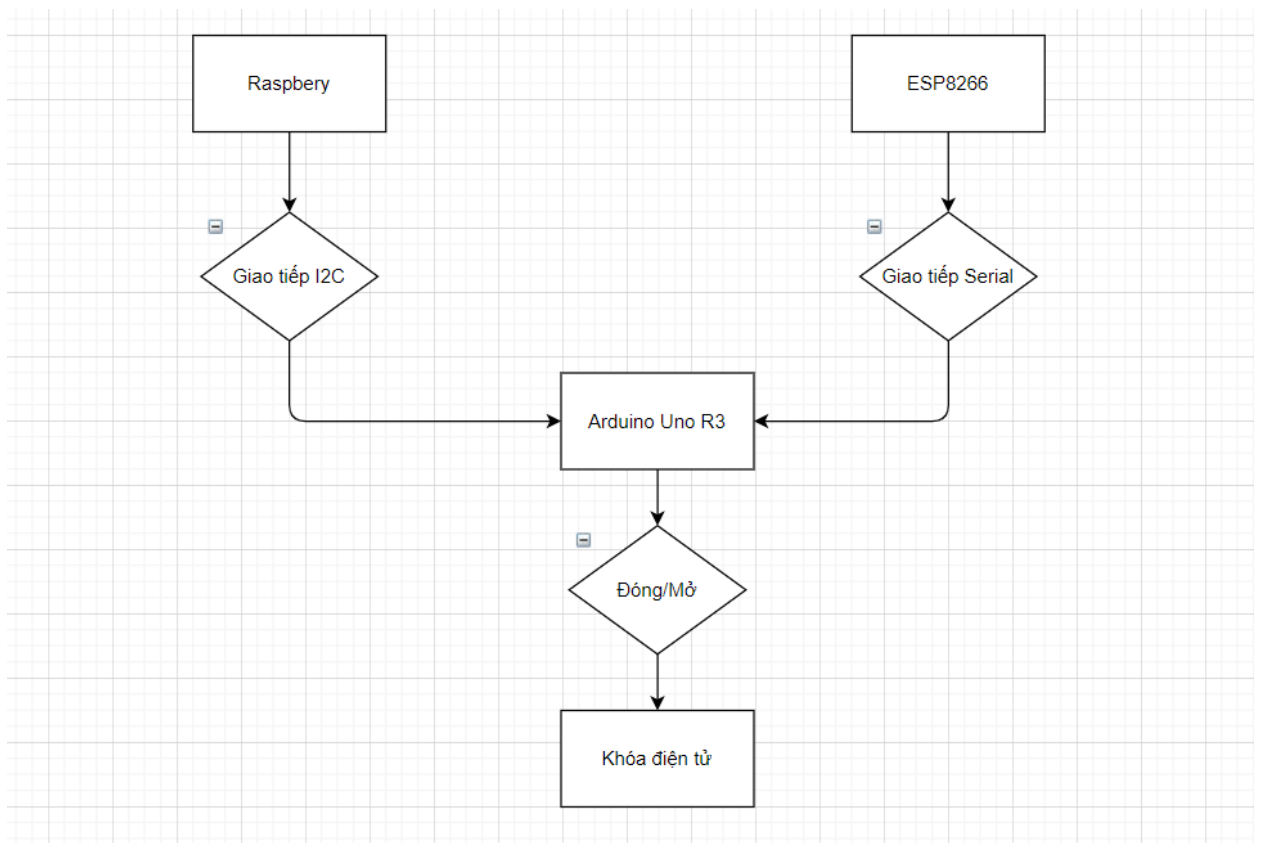
**Raspberry pi:** Kết nối với Camera pi và Arduino Uno, thực hiện nhận diện khuôn mặt và gửi tín hiệu đến Arduino thông qua giao tiếp Serial. Nếu đúng thì mở cửa.

**ESP8266:** Kết nối với Arduino Uno, lấy dữ liệu từ Firebase khi App Control đẩy dữ liệu lên Firebase. Arduino Uno bắt tín hiệu Serial từ ESP8266, mở khóa.

### Arduino UNO:

- ✓ Kết nối với Raspberry và ESP8266, nhận tín hiệu Serial.
  - ✓ Kết nối với đầu đọc RFID và Keypad 4x4
  - ✓ Nhận tín hiệu từ button
- ⇒ Mở khóa

#### 4.4/ Sơ đồ ERD:



- Tất cả dữ liệu của hình ảnh dùng để nhận diện khuôn mặt được lưu tại bộ nhớ của Raspberry.

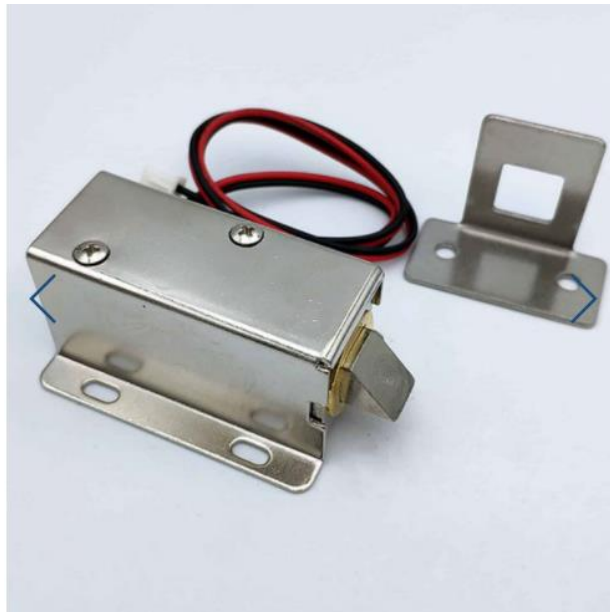
## PHẦN 5 – THỰC HIỆN DỰ ÁN

### 5.1/ Phân công công việc:

Tên	Công việc	Thời hạn	Tiến độ
Đại Nghĩa	<ul style="list-style-type: none"> <li>- Lập kế hoạch triển khai đồ án</li> <li>- Phân công công việc</li> <li>- Làm slide</li> <li>- Kiểm thử</li> </ul>	<ul style="list-style-type: none"> <li>- 1/10/2020</li> <li>- 1/10/2020</li> <li>- 18/12/2020</li> <li>- 20/12/2020</li> </ul>	Hoàn thành
Tấn Phát	<ul style="list-style-type: none"> <li>- Làm web app + giao tiếp I2C</li> <li>- Lắp ráp mô hình</li> <li>- Làm document</li> <li>- Kiểm thử</li> </ul>	<ul style="list-style-type: none"> <li>- 5/12/2020</li> <li>- 8/12/2020</li> <li>- 10/12/2020</li> <li>- 20/12/2020</li> </ul>	Hoàn thành
Quốc Đại	<ul style="list-style-type: none"> <li>- Chức năng quét thẻ RFID</li> <li>- Chức năng Keypad 4x4 + Giao tiếp Serial</li> <li>- Lắp ráp mô hình</li> <li>- Kiểm thử</li> </ul>	<ul style="list-style-type: none"> <li>- 5/12/2020</li> <li>- 5/12/2020</li> <li>- 10/12/2020</li> <li>- 20/12/2020</li> </ul>	Hoàn thành
Thế Pháp	<ul style="list-style-type: none"> <li>- Chức năng nhận diện khuôn mặt, nút nhấn, chuông báo động</li> <li>- Mua linh kiện vật liệu</li> <li>- Lắp ráp mô hình</li> <li>- Kiểm thử</li> </ul>	<ul style="list-style-type: none"> <li>- 5/12/2020</li> <li>- 20/11/2020</li> <li>- 10/12/2020</li> <li>- 20/12/2020</li> </ul>	Hoàn thành

### 5.2/ Các thiết bị, vật liệu cần có:

#### 1/ Khóa cửa điện từ:



## 2/ Raspberry pi 4:



## 3/ Camera pi:



**4/ Bàn phím ma trận:**



**5/ Đồ đọc thẻ RFID:**



## 6/ Arduino Uno:



## 7/ ESP8266:



## 9/ Mô hình: (Bao gồm khung và cửa)

Kích thước: 40cm x 30cm

Bề dày: 5mm



### 5.3/ Dự toán chi phí thiết bị, vật liệu:

STT	Tên thiết bị	Số lượng	Chi phí
1	Raspberry pi 4	1	1.725.000
2	Arduino UNO	2	250.000
3	ESP8266	1	85.000
4	Camera pi V2	1	175.000
5	Bàn phím ma trận 4x4	1	16.000
6	Đồ đọc thẻ từ RFID	1	46.000
7	Khóa cửa điện tử	1	108.000
8	Mô hình (bao gồm phụ kiện nhỏ)	1	100.000
<b>Tổng</b>			2.505.000

### 5.4/ Source code:

#### ❖ Raspberry Pi:

- Thu thập data

```

1  import cv2
2  import numpy as np
3
4  faceDetect=cv2.CascadeClassifier('haarcascade_frontalface_default.xml');
5  cam=cv2.VideoCapture(0);
6
7  id=input('enter user id')
8  sampleNum=0;
9  while(True):
10     ret,img=cam.read();
11     gray=cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
12     faces=faceDetect.detectMultiScale(gray,1.3,5);
13     for(x,y,w,h) in faces:
14         sampleNum=sampleNum+1;
15         cv2.imwrite("dataSet/User."+str(id)+"."+str(sampleNum)+".jpg",gray[y:y+h,x
16         cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255),2)
17         cv2.waitKey(100);
18     cv2.imshow("Face",img);
19     cv2.waitKey(1);
20     if(sampleNum>20):|
21         break
22 cam.release()
23 cv2.destroyAllWindows()
24

```



- **Trainning**

```

1 import os
2 import cv2
3 import numpy as np
4 from PIL import Image
5
6 recognizer = cv2.face.LBPHFaceRecognizer_create()
7
8 # recognizer=cv2.createLBPHFaceRecognizer();
9 path='dataSet'
10
11 def getImagesWithID(path):
12     imagePaths=[os.path.join(path,f) for f in os.listdir(path)]
13     faces=[]
14     IDs=[]
15     for imagePath in imagePaths:
16         faceImg=Image.open(imagePath).convert('L');
17         faceNp=np.array(faceImg,'uint8')
18         ID=int(os.path.split(imagePath)[-1].split('.')[1])
19         faces.append(faceNp)
20         print (ID)
21         IDs.append(ID)
22         cv2.imshow("training",faceNp)
23         cv2.waitKey(10)
24     return IDs,faces
25
26 Ids,faces=getImagesWithID(path)
27 recognizer.train(faces,np.array(IDs))
28 recognizer.save('recognizer/trainingData.yml')
29 cv2.destroyAllWindows()

```

- **Nhận diện**

```

1 import cv2
2 import numpy as np
3 import os
4 from PIL import Image
5 import serial
6 from time import sleep
7
8 ser = serial.Serial('/dev/ttyACM0',9600)
9 recognizer = cv2.face.LBPHFaceRecognizer_create();
10 recognizer.read('/home/pi/Face-Recognition-Train-YML-Python-master/recognizer/trai
11 cascadePath = "/home/pi/Face-Recognition-Train-YML-Python-master/haarcascade_front
12 faceCascade = cv2.CascadeClassifier(cascadePath);
13
14 font = cv2.FONT_HERSHEY_SIMPLEX
15
16 #iniciate id counter
17 id = 0
18
19 # names related to ids: example ==> KUNAL: id=1, etc
20 names = ['None', 'Thoai', 'Hieu']
21
22 face = 5
23 # Initialize and start realtime video capture
24 cam = cv2.VideoCapture(0)

```

```

25 cam.set(3, 320) # set video width
26 cam.set(4, 240) # set video height
27
28 # Define min window size to be recognized as a face
29 minW = 0.1*cam.get(3)
30 minH = 0.1*cam.get(4)
31
32 while True:
33     ret, img = cam.read()
34     #img = cv2.flip(img, -1) # Flip vertically
35     gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
36
37     faces = faceCascade.detectMultiScale(
38         gray,
39         scaleFactor = 1.2,
40         minNeighbors = 5,
41         minSize = (int(minW), int(minH)),
42     )
43
44     for(x,y,w,h) in faces:
45         cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 2)
46         id, confidence = recognizer.predict(gray[y:y+h,x:x+w])
47
48         # Check if confidence is less than 100 ==> "0" is perfect match
49         if (confidence < 100 ):
50             conf = round(100 - confidence)
51             print(type(conf))
52             if(conf > 30):
53                 id = names[id]
54                 confidence = " {0}%".format(round(100 - confidence))
55                 cv2.putText(img, str(id), (x+5,y-5), font, 1, (255,255,255), 2)
56                 cv2.putText(img, str(confidence), (x+5,y+h-5), font, 1, (255,255,0), 1)
57                 #print(type(confidence))
58                 face=6
59
60             else:
61                 id = "unknown"
62                 confidence = " {0}%".format(round(100 - confidence))
63                 cv2.putText(img, str(id), (x+5,y-5), font, 1, (0,0,255), 2)
64                 cv2.putText(img, str(confidence), (x+5,y+h-5), font, 1, (255,255,0), 1)
65
66         if(face==6):
67             try:
68                 while(True):
69
70                     string = 'Open'
71                     ser.write(string.encode())
72                     print('open')
73                     # conf=0
74                     sleep(2)
75
76                     string = ''
77                     ser.write(string.encode())
78
79                     ser.close()
80
81                     ser.resume()
82
83             except:
84                 print('close')
85
86
87         cv2.imshow('camera',img)
88
89         k = cv2.waitKey(10) & 0xff # Press 'ESC' for exiting video
90         if k == 27:
91             break
92
93     # Do a bit of cleanup
94     print("\n [INFO] Exiting Program and cleanup stuff")
95     cam.release()
96     cv2.destroyAllWindows()
97
98

```

## ❖ ESP8266:

```
// #include "printf.h"
#include "user_defined.h"
#include "Globalvariables.h"

#include <DNSServer.h>
#ifdef ESP32
#include <WiFi.h>
#include <AsyncTCP.h>
#elif defined(ESP8266)

#include <ESP8266WiFi.h>
#include <ESPAsyncTCP.h>
#endif
#include "ESPAsyncWebServer.h"
#include <ESPAsyncWebServer.h>
#include <FS.h>
#include <Wire.h>
#include <ArduinoJson.h>
#include <SoftwareSerial.h>
SoftwareSerial s(3, 1);
#define test_master_
// Replace with your network credentials
const char* ssid = "REPLACE_WITH_YOUR_SSID";
const char* password = "REPLACE_WITH_YOUR_PASSWORD";
const char* ssid_ = "Smart_Door";
const char* password_ = "12345678";
const char* _apName = "no-net";
const char* _apPassword = NULL;

const char* PARAM_MESSAGE_ftxtmode4 = "running_model";
```

## FPT POLYTECHNIC

```
const char* PARAM_MESSAGE_ftxt = "ftxt";
const char FILESETTING[] PROGMEM = "/config.json";

// Create AsyncWebServer object on port 80
DNSServer dnsServer;
AsyncWebServer server(80);

//#include <Servo.h>
//Servo myservo;
//#include <Wire.h>

class CaptiveRequestHandler : public AsyncWebHandler {
public:
    CaptiveRequestHandler() {}
    virtual ~CaptiveRequestHandler() {}

    bool canHandle(AsyncWebServerRequest *request) {
        //request->addInterestingHeader("ANY");
        return true;
    }

    void handleRequest(AsyncWebServerRequest *request) {
        AsyncResponseStream *response = request->beginResponseStream("text/html");
        response->print("<!DOCTYPE html><html lang=\"en\"><head><meta charset=\"UTF
        response->print("<div id='box'><h1 style='color: red;'>Smart Home App</h1><
        response->print("</body></html>");
        request->send(response);
    }
};

void setup() {

    Serial.begin(9600);
    s.begin(9600);
    init_wifi();
    Wire.begin();

    // Serial.println(".....");
    // Serial.println("AirConditioner Controller Center");
    // Serial.println("VerAug182020 13h40");

}

void loop() {
    // put your main code here, to run repeatedly:
    dnsServer.processNextRequest();
}

int init_wifi() {
    // Initialize SPIFFS
    if (!SPIFFS.begin()) {
        Serial.println("An Error has occurred while mounting SPIFFS");
        return 1;
    }
    BTS_wifi(ssid_, password_);
    // Route for root / web page
    server.on("/", HTTP_GET, [] (AsyncWebServerRequest * request) {
        request->send(SPIFFS, "/index.html", String(), false, processor);
    });
};
```

```
// Route to load style.css file
server.on("/style.css", HTTP_GET, [] (AsyncWebServerRequest * request) {
    request->send(SPIFFS, "/style.css", "text/css");
});
// Route to load logo.png file
server.on("/logo", HTTP_GET, [] (AsyncWebServerRequest * request) {
    request->send(SPIFFS, "/logo.png", "image/png");
});
server.on("/get1", HTTP_GET, [] (AsyncWebServerRequest * request) {
    s.print("ON");
    request->send(200, "text/html", "<!DOCTYPE html><html lang=\"en\"><head><

    if (s.available()) {
        Serial.println(char(s.read()));
    }
});

server.onNotFound(notFound);
// Start server
server.addHandler(new CaptiveRequestHandler()).setFilter(ON_AP_FILTER); //on
server.begin();
return 0;
}

void BTS_wifi(char const *apName, char const *apPassword) {
    _apName = apName;
    _apPassword = apPassword;

    WiFi.mode(WIFI_AP);
    WiFi.softAP(_apName, _apPassword); //password option
    // delay(500); // Without delay I've seen the IP address blank
    Serial.println(F("AP IP address: "));
    dnsServer.start(53, "*", WiFi.softAPIP());
    Serial.println(WiFi.softAPIP());
}

String processor(const String& var) {
    String s_;
    Serial.println(var);
}

void notFound(AsyncWebServerRequest *request) {
    request->send(404, "text/plain", "Not found");
}
```

❖ **Arduino 1:**• **Thẻ từ + Button**

```
#include <SoftwareSerial.h>
#include <SPI.h>
#include <MFRC522.h>
#include <Keypad.h>
#include <Wire.h>

#define SS_PIN 10
#define RST_PIN 9

MFRC522 mfrc522(SS_PIN, RST_PIN);
int button = 6;
SoftwareSerial mySerial(2, 3);

void setup() {
    Serial.begin(9600);
    SPI.begin();          // Initiate SPI bus
    mfrc522.PCD_Init();   // Initiate MFRC522

    mySerial.begin(9600);
    pinMode(button, INPUT);
    digitalWrite(button, LOW);
}
```

```

void loop() {

    //---GIAO TIẾP SERIAL---//

    if (Serial.available()) {
        String nhan = "";
        while (Serial.available()) {
            nhan = Serial.readString();
            Serial.print(nhan);
        }
        if (nhan == "ON") {

            char ch = Serial.read();
            Serial.write(ch);
            mySerial.write(ch);
            delay(500);
        }
    }

    // //-----ĐỌC GIÁ TRỊ BUTTON-----//
    int buttonStatus = digitalRead(button);
    if (buttonStatus == HIGH) {
        char ch = Serial.read();
        Serial.write(ch);
        mySerial.write(ch);
        delay(500);
        buttonStatus = LOW;
    }
}

//-----THẺ TỪ-----//

// Look for new cards
if ( ! mfrc522.PICC_IsNewCardPresent())
{
    return;
}
// Select one of the cards
if ( ! mfrc522.PICC_ReadCardSerial())
{
    return;
}
//Show UID on serial monitor

String content = "";
byte letter;
for (byte i = 0; i < mfrc522.uid.size; i++)
{
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], HEX);
    content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
    content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
content.toUpperCase();
if (content.substring(1) == "B6 1C AE F8")
{
    Serial.println("ID đúng, mở khóa");
    char ch = Serial.read();
    Serial.write(ch);
    mySerial.write(ch);
    delay(1000);
}
}

```

## ❖ Arduino 2:

- Bàn phím số

```
#include <Wire.h>
#include <Keypad.h>
#include <SoftwareSerial.h>

//----I2C-----//
#define SLAVE_ADDRESS 0x04
int number = 0;
//-----//

#define Password_Length 8
const int relayLock = 2;
const int relayBuzzer = 3;

//-----KEYPAD-----//
char STR[4] = {'2', '3', '1', '2'}; // Cài đặt mật khẩu tùy ý
char str[4] = {' ', ' ', ' ', ' '};
int i, j, count, error = 0;
char key;
char ch;
const byte ROWS = 4;
const byte COLS = 4;
String buff;

char hexaKeys[ROWS][COLS] = {

    {'1', '2', '3', 'A'},
    {'4', '5', '6', 'B'},
    {'7', '8', '9', 'C'},
    {'*', '0', '#', 'D'},
};

byte rowPins[ROWS] = {13, 12, 11, 10};
byte colPins[COLS] = {9, 8, 7, 6};
```



```
Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

//-----//

unsigned long time = 0;
unsigned long delayTime = 3000;

void setup() {
    Wire.begin(0x8);
    Wire.onReceive(receiveEvent);
    Serial.begin(9600); //bật serial, baudrate 9600
    pinMode(relayLock, OUTPUT);
    pinMode(relayBuzzer, OUTPUT);
    digitalWrite(relayLock, LOW);
    digitalWrite(relayBuzzer, LOW);

    //-----//
}

void receiveEvent(int howMany) {

    while (Wire.available()) {
        ch = Wire.read();
        if (ch == 1) {
            digitalWrite(relayLock, HIGH);
            delay(500000);
            Serial.print("ON");
        }
    }
}
```

```

void loop() {

    //-----PHÁT HIỆN TÍN HIỆU-----//
    if (Serial.available()) {
        time = millis();
        digitalWrite(relayLock, HIGH);
        while (Serial.available()) {
            Serial.read();
            delay(3);
        }
    }

    if (millis() - time > delayTime) {
        digitalWrite(relayLock, LOW);
    }

    //-----ĐỌC GIÁ TRỊ TỪ KEYPAD VÀ SO SÁNH PASSWORD-----//

    key = customKeypad.getKey();
    //char key = keypad.getKey(); // Ký tự nhập vào sẽ gán cho biến Key

    if (key) // Nhập mật khẩu
    {
        if (i == 0) {
            str[0] = key;

            Serial.print(str[0]);
            delay(100); // Ký tự hiển thị trên màn hình LCD trong 1s
            // lcd.setCursor(6, 1);
            // Serial.print("*"); // Ký tự được che bởi dấu *
        }
        if (i == 1) {
            str[1] = key;

            Serial.print(str[1]);
            delay(100);

            //Serial.print("*");
        }
        if (i == 2) {
            str[2] = key;

            Serial.print(str[2]);
            delay(100);
            ;
            //Serial.print("*");
        }
        if (i == 3) {
            str[3] = key;

            Serial.print(str[3]);
            delay(100);
        }
    }
}

```

```

        //Serial.print("");
        count = 1;
    }
    i = i + 1;
}

if (count == 1) {
    if (str[0] == STR[0] && str[1] == STR[1] && str[2] == STR[2] &&
        str[3] == STR[3]) {

        Serial.print("    Correct!");
        digitalWrite(relayLock, HIGH);
        delay(3000);
        digitalWrite(relayLock, LOW);
        i = 0;
        count = 0;
        error = 0;
        delay(1000);
    } else {
        Serial.print("    Try Again!");
        digitalWrite(relayBuzzer, HIGH);
        delay(100);
        digitalWrite(relayBuzzer, LOW);
        error++;
        i = 0;
        count = 0;
        Serial.print("    ");
        Serial.println(error);

        if (error == 3) {
            Serial.println("Sai 3 lan");
            digitalWrite(relayBuzzer, HIGH);
            delay(2000);
            digitalWrite(relayBuzzer, LOW);
            error = 0;

        }
        Serial.println("Enter Password");
    }
}
//-----//
}

```

## 5.5/ Thư viện:

### ❖ ESP8266:

<DNSServer.h>

<WiFi.h>

<AsyncTCP.h>

<ESP8266WiFi.h>

<ESPAsyncTCP.h>

<FS.h>

<Wire.h>

<ArduinoJson.h>

❖ **Bàn phím ma trận:**

<Keypad.h>

❖ **Giao tiếp Serial:**

<SoftwareSerial.h>

<Wire.h>

❖ **Thẻ từ RFID:**

<SPI.h>

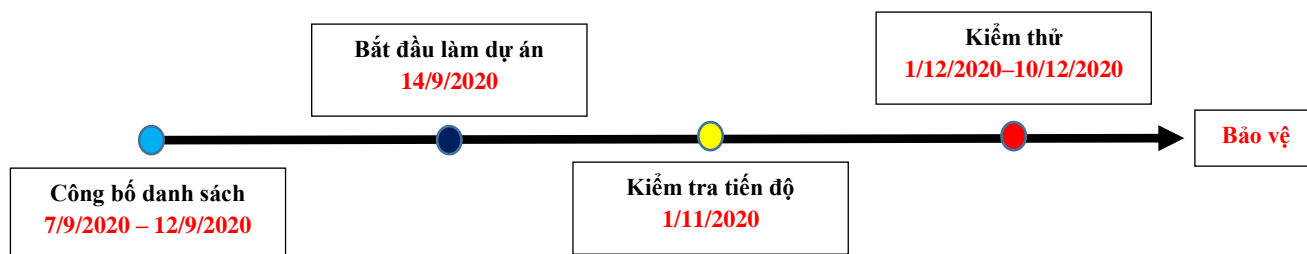
<MFRC522.h>

## 5.6/ Hình ảnh thực tế



## PHẦN 6 – KIỂM THỬ

### 6.1/ Kế hoạch:



### 6.2/ Kiểm thử:

- ❖ Bàn phím ma trận keypad 4x4
- ❖ Quẹt thẻ RFID
- ❖ Keypad 4x4 kết hợp LCD 16x2, lock, led và buzzer
- ❖ RFID kết hợp LCD 16x2, lock, led và buzzer
- ❖ Giao tiếp Serial giữa ESP8266 và Arduino
- ❖ Giao tiếp Serial giữa Raspberry Pi và Arduino

## PHẦN 7 – TRIỂN KHAI VÀ SỬ DỤNG

### **Bên ngoài cửa:**

- Người dùng sử dụng thẻ từ do bên hệ thống cung cấp quét vào vùng quét thẻ, cửa sẽ mở.
- Bên ngoài cửa có bàn phím số để nhập mật khẩu, người dùng sẽ bấm vào 4 số, nếu đúng thì cửa sẽ mở, nếu nhập sai hơn 5 lần sẽ có chuông báo động.
- Camera đặt trên cửa dùng để nhận diện khuôn mặt của người dùng, phần thiết lập khuôn mặt sẽ do người cung cấp tạo dữ liệu khuôn mặt của người dùng.

### **Bên trong cửa:**

- Người dùng chỉ cần xài nút bấm là cửa sẽ mở.

## TÀI LIỆU THAM KHẢO

<http://arduino.vn/bai-viet/275-giao-tiep-giua-hai-mach-arduino-bat-ky>

<https://pypi.org/project/opencv-python/>

<http://arduino.vn/bai-viet/80-su-dung-digitalread-va-giao-tiep-serial-digital-read-serial>

<https://randomnerdtutorials.com/display-images-esp32-esp8266-web-server/>

<http://arduino.vn/bai-viet/80-su-dung-digitalread-va-giao-tiep-serial-digital-read-serial>

<https://github.com/kroimon/Arduino-SerialCommand>

<https://github.com/esp8266/arduino-esp8266fs-plugin>