

**VISVESVARAYA TECHNOLOGICAL UNIVERSITY,  
BELGAUM 59014**



Internet of Things Project Report on

**“Digital lock using mask detection”**

By

**Md Ibaduddin Saffan(1BM19CS085) Naman Singh(1BM19CS093)**  
**Preetham H (1BM19C120) Prithviraj T Chavan(1BM19CS123)**

Under the Guidance of

**Mrs. Antara Roy**

Assistant Professor, Department of CSE  
BMS College of Engineering

IoT Application Development carried out at



Department of Computer Science and Engineering  
BMS College of Engineering  
(Autonomous college under VTU)  
P.O. Box No.: 1908, Bull Temple Road, Bangalore-560 019  
2021-2022

**BMS COLLEGE OF ENGINEERING**  
**DEPARTMENT OF COMPUTER SCIENCE AND**  
**ENGINEERING**



***CERTIFICATE***

This is to certify that the Internet of Things project titled “**Digital lock using mask detection**” has been carried out by **Md Ibaduddin Saffan(1BM19CS085), Naman Singh(1BM19CS093), Preetham H (1BM19C120) and Prithviraj T Chavan(1BM19CS123)** during the academic year 2021-2022.

Signature of the guide

**Mrs. Antara Roy**

Assistant Professor

Department of Computer Science and Engineering

BMS College of Engineering, Bangalore

**Examiners**

**Name**

**Signature**

**1.**

**2.**

**BMS COLLEGE OF ENGINEERING**  
**DEPARTMENT OF COMPUTER SCIENCE AND**  
**ENGINEERING**



***DECLARATION***

We, **Md Ibaduddin Saffan(1BM19CS085), Naman Singh(1BM19CS093), Preetham H (1BM19C120) and Prithviraj T Chavan(1BM19CS123)** students of 5<sup>th</sup> Semester, B.E, Department of Computer Science and Engineering, BMS College of Engineering, Bangalore, hereby declare that, this IoT Application development work entitled "**Digital lock using mask detection**" has been carried out

by us under the guidance of **Mrs. Antara Roy**, Assistant Professor, Department of CSE, BMS College of Engineering, Bangalore during the academic semester October-January 2022

We also declare that to the best of our knowledge and belief, the development reported here is not from part of any other report by any other students.

Signature

**Md Ibaduddin Saffan(1BM19CS085)**

**Naman Singh(1BM19CS093)**

**Preetham H (1BM19C120)**

**Prithviraj T Chavan(1BM19CS123)**

# **Introduction**

## **Objective of the project**

- **COVID-19 pandemic is having a devastating effects on society and economies around the world. It is essential to wear masks all the time at public places or any place with mass gatherings to curb the spread of this virus.**
- **We are trying to enforce masks on everyone at any place of risk such as shopping malls, offices, schools etc.**

## **Abstract description of the project**

**Our project consists of a camera, digital lock, ESP32 Board and a processing device. The camera is placed in front of the door and the door is equipped with a digital lock which is connected to ESP32 board. The camera captures the image of the person in front of the door and sends it to the processing device. If the mask is detected on the face of the person correctly, a signal is sent to the digital lock via Bluetooth. If the signal is positive, the door is unlocked or else the door remains locked.**

## Literature Survey

Sl.No	Name of the Project or Product (Existing)	Commercial or Non-Commercial	Features
1.	Detection of Face Mask using Convolutional Neural Network	Non-Commercial	Detects the presence of mask on the face.
2.	A Deep Learning Model for Face Mask Detection	Non-Commercial	Simple and effective model for real-time monitoring using the convolution neural network to detect

Explanation about each existing similar commercial and non-commercial product {product feature, commercial or not, drawbacks etc.}

1. They propose a simple and effective model for real-time monitoring using the convolution neural network to detect whether an individual wears a face mask or not. The model is trained, validated, tested upon two datasets. Corresponding to dataset 1, the accuracy of the model was 95.77% and, it was 94.58% for dataset 2.
2. They made a successful attempt to preserve inter and intra class variations of face mask detection using symbolic approach. They studied the different classifiers like Support Vector Machine and a Symbolic Classifier. The project is developed as a prototype to monitor temperature measurement and to detect mask for the people. The first method is performed using temperature sensor used to detect the present temperature of the body and automatically spray the sanitizer.

**Proposed Project : Digital lock using mask detection**

**Feature and its advantage**

Detects if the person is wearing mask and unlocks the door only if the mask is detected on the face.

It helps prevent the spread of COVID-19 in public places such as malls, offices, colleges etc. It enforces the use of masks onto the people.

# **Hardware and Software Requirements**

## **Hardware requirement**

- **ESP 32 Board**
- **Camera**
- **Digital Lock**

## **List the Component used**

- **ESP 32 Board**
- **Camera**
- **Digital Lock**
- **Mobile phones**
- **12V adaptor**
- **Micro USB cable**

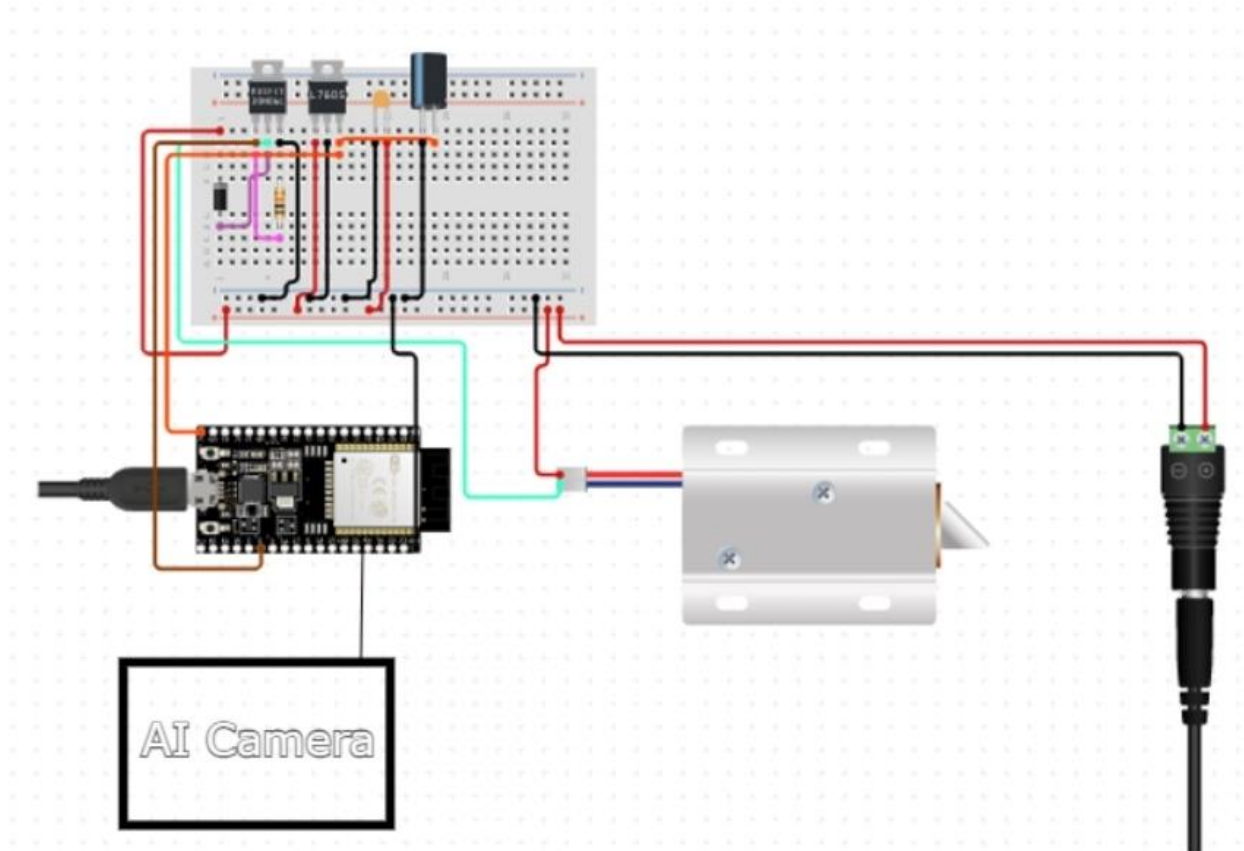
## **Software Requirements**

**OS: Android**

**IDE: Arduino IDE**

# Design and cost analysis

## Architectural diagram or Circuit diagram



### Explanation :

A camera is placed in front of the doors and the doors are equipped with digital locks connected to ESP32 Board. When a person is detected, camera captures the image and sends the image for processing to the mobile application. If the mask is detected on the person, the mobile application sends the signal to ESP32 and the digital lock unlocks the door. If the mask is not detected, the door remains locked. All the devices are connected via Bluetooth.

### Cost Analysis:

Component	Quantity	Price(in Rs)	Total Price
ESP32 Board	1	400	400
Digital Lock	1	2300	2300
AI Camera	1	999	999
Total			3699

## Source code

```
#include "BluetoothSerial.h"
```

```
#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif
```

```
int a=0;
```

```
BluetoothSerial SerialBT;
```

```
void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32test"); //Bluetooth device name
  Serial.write("Device Ready!");
}
```

```
void loop() {

  if (Serial.available()) {
    SerialBT.write(Serial.read());
  }
  if (SerialBT.available()) {
    a=SerialBT.read();
    if(a=='0')
      Serial.write("\nMask not detected --> Door locked\n");
    else
      Serial.write("Mask detected --> Door unlocked\n");

  }
  delay(1000);
}
```

### **For Mask Detection:**

```
_pickImage() async {
  final ImagePicker _picker = ImagePicker();
  final XFile image = await _picker.pickImage(source: ImageSource.camera);
```



```

if (image == null) {
    return;
}
final tmp = File(image.path);
var recognitions = await Tflite.runModelOnImage(
    path: image.path, numResults: 16, threshold: 0.5);
print(recognitions);
pred = recognitions[0]['label'];
setState(() {
    img = tmp;
});
}

_loadModel() async {
    var res = Tflite.loadModel(
        labels: "assets/labels.txt", model: "assets/model_unquant.tflite");
}

void _sendMessage() async {
    String text = "0";
    await _pickImage();
    if (pred == "Mask") {
        text = "1";
    } else {
        text = "0";
    }
    if (text.length > 0) {
        try {
            connection.output.add(utf8.encode(text + "\r\n"));
            await connection.output.allSent;

            setState(() {
                messages.add(_Message(clientID, text));
            });

            Future.delayed(Duration(milliseconds: 333)).then((_) {
                listScrollController.animateTo(
                    listScrollController.position.maxScrollExtent,
                    duration: Duration(milliseconds: 333),
                    curve: Curves.easeOut);
            });
        } catch (e) {
            // Ignore error, but notify state
            setState(() {});
        }
    }
}

```

```
}  
}
```

### Flutter App:

```
Widget build(BuildContext context) {  
  final List<Row> list = messages.map((_message) {  
    return Row(  
      children: <Widget>[  
        Container(  
          child: Text(  
            (text) {  
              return text == '/shrug' ? '\_(ツ)_/-' : text;  
            }(_message.text.trim()),  
            style: TextStyle(color: Colors.white)),  
          padding: EdgeInsets.all(12.0),  
          margin: EdgeInsets.only(bottom: 8.0, left: 8.0, right: 8.0),  
          width: 222.0,  
          decoration: BoxDecoration(  
            color:  
              _message.whom == clientID ? Colors.blueAccent : Colors.grey,  
            borderRadius: BorderRadius.circular(7.0)),  
        ),  
      ],  
      mainAxisAlignment: _message.whom == clientID  
        ? MainAxisAlignment.end  
        : MainAxisAlignment.start,  
    );  
  }).toList();  
  
  return Scaffold(  
    appBar: AppBar(  
      title: (isConnecting  
        ? Text('Connecting chat to ' + widget.server.name + '...')  
        : isConnected  
        ? Text('Mask Verification with ' + widget.server.name)  
        : Text('Chat log with ' + widget.server.name))),  
    body: SafeArea(  
      child: Column(  
        children: <Widget>[  
          SizedBox(height: 20),  
          SizedBox(  
            child: img != null ? Image.file(img) : Text("Null"),  
            height: 500,
```

```

    ),
    Text("Pred : " + pred),
    SizedBox(
      height: 100,
    ),
    Row(
      children: <Widget>[
        Flexible(
          child: Container(
            margin: const EdgeInsets.only(left: 78.0),
            child: Text(
              isConnecting
                ? 'Wait until connected...'
                : isConnected
                  ? 'Take Photo'
                  : 'Chat got disconnected',
            ),
          ),
        ),
      ],
    ),
    SizedBox(width: 100),
    Container(
      margin: const EdgeInsets.only(right: 8.0),
      child: IconButton(
        icon: const Icon(Icons.camera_alt),
        onPressed: isConnected ? () => _sendMessage() : null,
      ),
    ),
  ],
),
),
);
}

```