Project Report On

EVM 2.0

Submitted by

Sayan Sengupta - 16905518012
Sarthak Mukherjee- 16905518013
Priyanshu Bandyopadhyay - 16905518014
Nirmalya Maity - 16905518021
Anusree Mukherjee - 16905518050

"A dissertation submitted in partial fulfillment of the requirements of Bachelor of Technology Degree in Applied Electronics and Instrumentation Engineering of the Maulana Abul Kalam Azad University of Technology"

Under the guidance of

Shri Jayjeet Sarkar
Assistant Professor,
Dept. of Applied Electronics and Instrumentation Engineering

Department of Applied Electronics & Instrumentation Engineering
Academy of Technology
(Affiliated to Maulana Abul Kalam Azad University of Technology, West Bengal)
Aedconagar - 712121, Hooghly, West Bengal

Forward

This is to certify that the project report entitled “EVM 2.0”, submitted to the Department of Applied Electronics and Instrumentation Engineering, Academy of Technology, in partial fulfilment for the award of the degree of Bachelor of Technology in Applied Electronics and Instrumentation Engineering is a record of bonafide work by Sayan Sengupta (12), Sarthak Mukherjee (13), Priyanshu Bandyopadhyay (14), Nirmalya Maity (21) & Anusree Mukherjee (50) under my supervision and guidance.

Place:
Date:

(Jayjeet Sarkar)

Assistant Professor, Dept. of AEIE,
Academy of Technology.

Acknowledgement

We are grateful to our respectable guide, Prof. Jayjeet Sarkar, whose insightful leadership and knowledge benefited us to complete this project successfully. Thank you so much for your continuous support and presence whenever needed. The completion of this project could not have been possible without the participation and assistance of a lot of individuals contributing to this project. However, we would like to express our deep appreciation and indebtedness to our teachers and supervisors for their endless support, kindness, and understanding during the project duration. Last but not the least, We would like to thank everyone who is involved in the project directly or indirectly.

Contents

Chapter 1: Introduction

The voting system is a set of rules which define how the desire of people may be expressed and how results may be achieved from it. For this purpose, an electronic voting machine EVM is introduced in this paper which replaced conventional methods of voting i-e manual voting.

The proposed machine in this paper is faster, efficient, and reliable, and error-free as compared to the manual voting system which is slower, poses full-day fatigue on people, and chances of error are greater. Its main feature is its ease of operation. Voters vote very easily and final results are displayed in no time by just pressing a result button after the elections have been conducted.

Elections play a major role in the political process of democracy. However, if the election is corrupted in the process itself, it is clear that the democracy based on that election may be fraudulent. Therefore, the cheating strategies in the election will destroy the democratization that the citizens would like to have.

[1]Voting procedures play a significant role in the conduct of free and fair elections in a democracy. It converts voters' preferences into a political mandate which in turn forms the basis for policy making. In India, the largest democracy, with more than 800 million registered voters and a complex multi-party system, electoral fraud has been one of the leading causes of concern.[1]

Nowadays in India, two types of methods are used for voting. The 1st method is secret ballot paper, in which lots of paper is used and the second method is EVM (Electronic Voting Machine) which has been used since 2003.

• Paper-based electronic voting system

Paper-based electronic voting system Also called "document ballot voting system". [2]This system was used to count the votes electronically which were marked by a hands-on ballot paper. Punch card voting, mark sense, and digital pen voting systems are all examples of this system. In several constituencies under the paper ballot system, polling booths would be captured, and ballot boxes would be stuffed.[2]

• Direct-recording electronic (DRE) voting system

In this system, a mechanical or electro-optical component like a touch screen was used to provide a ballot display. The machine was programmed to record voting data and ballot images, count and tabulate the voting data and finally, the results may be stored on removable memory or provided as a printed copy. There may be a means of transmitting the individual ballots or vote totals to a central location for the formation of final results.

To address these frauds and simplify the electoral procedure, the Election Commission of India (ECI) introduced Electronic Voting Machines (EVMs) in the late ’90s. But, since their introduction in around 1998 EVMs haven’t witnessed any major change and still use old technology, which not only has its limitations but also introduces a lot of friction in the process, causing delay and a whole host of other problems such as EVM tampering.

Now biometrics is the science and technology of measuring and analyzing biological data. Biometrics refers to technologies that measure and analyze human body characteristics, such as DNA, fingerprints, eye retinas and irises, voice patterns, face recognition, and hand measurements, for authentication purposes. Among the several human fingerprints and face recognition remains a very common identifier and the biometric method of choice among law enforcement. These concepts of human identification have to leadmakesStore to the development of fingerprint scanners and webcams that serve to quickly identify individuals and assign access privileges.

Through our project, EVM 2.0, we want to upgrade the existing EVMs with newer technologies and by doing so, we want to make the voting process more transparent. We are trying to make a more modern and secure version of the EVM, which will be easy to use and will protect the democratic rights of the citizens of a country. This will not only solve the problem of election fraud but also make the tallying and verification process more frictionless. We aim to achieve this by integrating multiple technologies like fingerprint authentication, face-detection & SMS alerts directly into the EVMs which will act as multiple layers of authentication towards verifying the voter. This will ensure that only the authentic voters can cast their votes and that too only once in a specific election. In this way, the problem of booth rigging can be minimized and the overall voting processes would be a lot easier too.

Chapter 2: Review of previous work

In April 2018, According to the paper [7] published “Authentication of Electronic Voting Machine” and its related research and works, several voting systems, voter Identification, and authentication techniques were introduced to secure voting platforms and Overcome fake voting. First, the voter gets its RFID tag scanned by an RFID scanner.

The ID number Obtained by the RFID reader is checked with the database. If a match is found then facial recognition makes it done, else the voter is declared not eligible to vote. If the ID is just cleared the voter Proceeds to the GUI voting interface, wherein the voter can cast the vote for desired from the Given choices. This work is done on the face recognition of the voter and makes voting secure.

In August 2016, the paper named [8] “Introduction Design of an Electronic Voting Machine to Ensure Proper Voting Process, Minimizing Rigging, Using Radio Frequency Identification Technology” If we talk about EVM’s literature then according to another paper publish Work is done with NIT Durgapur and this was for Store proper voting process, minimizing Rigging, using radio frequency identification technology. This project was also on the secure Voting by voting machine. This project was done by using an RFID reader to read the Id of The person at the gate to ensure the voter is legal or not.

In March 2012[9] a paper was published with “Electronics Voting Machine a Review”. The work of that paper was to calculate the possibilities of its use in the future. In April 2010, another paper[10] was published with the Collaboration of the “Net India, (P) Ltd. Hyderabad” and the “The University of Michigan” with the name of the paper “Security Analysis of India’s Electronic Voting Machines”. According to this paper, EVM stores just concluded that dishonest insiders and criminals with physical access to machines can insert malicious hardware that can steal votes for the lifetime of the machines. And our project is inspired by this work. This paper says that we should carefully reconsider how to achieve a secure and transparent voting system that is suitable to Its national values and requirements. One option is given in this paper as advice to use “Voter-Verifiable paper audit trail” (VVPAT), which combines an electronic record in a DRE with a paper vote record that can be audited by hand. And then another option was given in this Precinct-Count Optical Scan (PCOS) voting, where voters fill out ballot papers that are scanned by the voting machine at the polling station before being placed in a ballot box but this is too costly and also results will take time to declare. Another and last paper [5] “A Preview on a Microcontroller Based Electronic Voting Machine” was published in Mar 2013. This was a microcontroller-based testing methodology, developed new tools that are specifically tailored to the security analysis of these systems, and learned several lessons, all of which should be of use to another user. In both systems that we analyzed, we found major security vulnerabilities that could compromise the confidentiality and availability of the voting process.

Chapter-3: Theory

Basic Idea

With voting being a major pillar in any democracy, securing and making the voting process easy and accessible to all, re-establishes trust among voters which not only benefits individual societies but also the entire country as a whole. Our solution puts forward a unique blend of modern technologies along with long-trusted voting methodologies to make the voting process easier and more secure. It also ensures viability and avoids any hefty additional economic toll, as we are innovatively implementing technologies, for the betterment of all.

General working

When a voter comes, they'll be asked to place their finger on the fingerprint scanner. If the fingerprint matches with our voter database, then only they'll be able to go to the next step otherwise not. After that, the face ID will be matched using a face recognition system. For the physically disabled, we'll omit the first step of verification and go for face recognition directly. After the verification process is over, the voter can cast their valuable vote. Then, as soon as the vote is cast, an SMS alert will be sent to the registered mobile number of the voter.

First, we are going to create the database of the voters by taking the fingerprint samples, face ID, mobile numbers. These details will be stored in various components in our system like fingerprints in the fingerprint sensor itself, Face-ID in the localhost, mobile numbers in the excel sheet. But for implementation in the real world, we can use the Aadhaar database UIDAI.

We are using IoT to implement this project. Now how we are going to implement this project. We will be storing the fingerprints in the fingerprint sensor itself as it comes with a memory to store up to 128 unique fingerprints. We are going to use Google Sheets to store the log details of the users. For this, we are using IFTTT. In the "If This Then That" block setup we are using Webhooks (as 'This') and Google Sheet (as 'That'). After creating a few columns in the Google Sheet, we can store the log details. If the fingerprint is matched with any of the samples stored in the database, then the user will be asked for face authentication. For this purpose, we are using the Laptop Webcam. When a user comes in front of the camera, it detects the face and matches the enrolled faces. If fingerprint and face authentication both are successful then the corresponding user details will be selected from our database and the mobile number will be shared with Google Sheets. A column will be there in the Google Sheets to store the mobile number of the user. An automatic SMS sender will be used to send the SMS to the respective users. The SMS sender will be programmed in such a way that as soon as the mobile number comes from the

Arduino, it will send the personalized SMS to the user. The whole process is shown in the block diagram below.

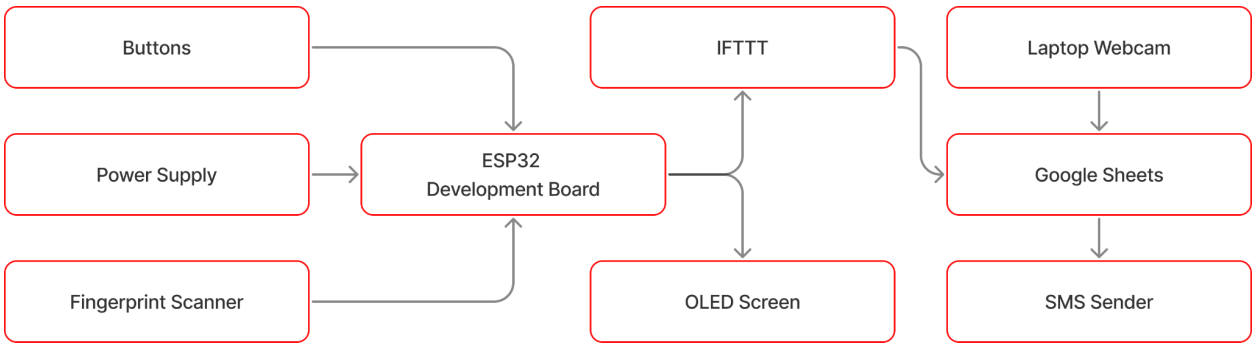

Fig. 3.1: Block Diagram

Component wise Functions

ESP-32 Development Board

This can be understood as the brain of the system. We have chosen the ESP32 development board instead of Arduino because of the ports and features present in this board. If we would have used Arduino then we would have required a Wi-Fi shield for the Arduino. The ESP32 is the ESP8266 successor. It adds an extra CPU core, faster Wi-Fi, more GPIOs, and supports Bluetooth 4.2 and Bluetooth Low Energy (BLE). For the Wi-Fi shield and other connection protocols, the ESP32 development board seemed to be the best choice for this project.

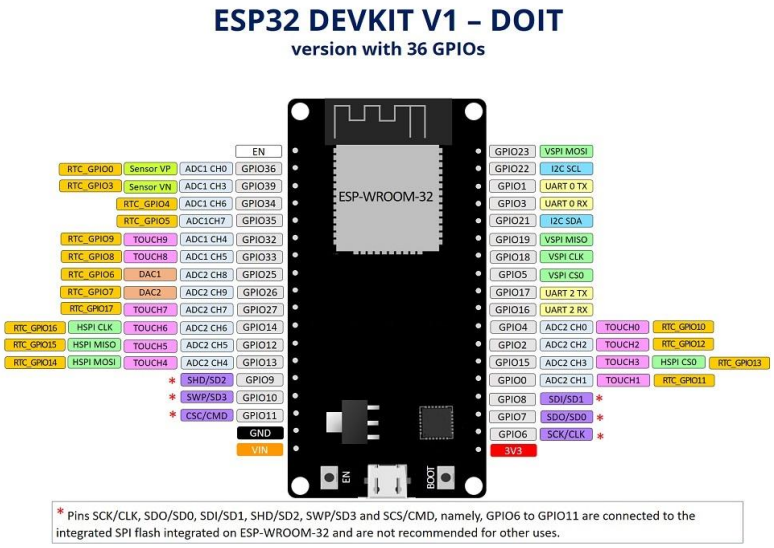

Fig. 3.2: ESP-32 Development Board

OLED

The OLED is used to display all the relevant information to the voter. That means we can indicate when the device is ready, when to keep the face in front of the camera, when to place the finger in the fingerprint scanner and if the vote has been registered. The corresponding message will be displayed and then the next steps will also be indicated through this OLED screen.

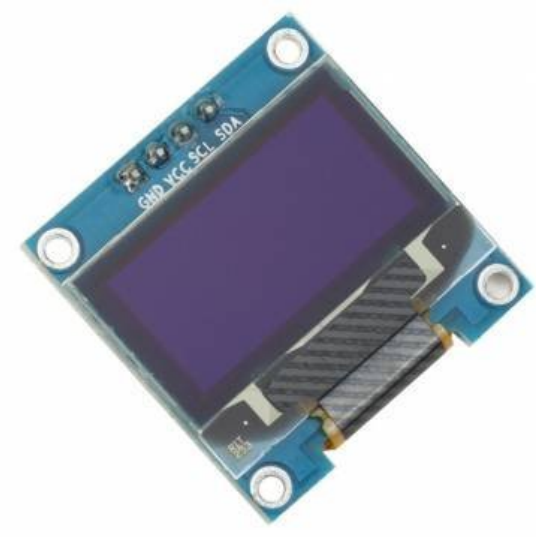

Fig. 3.3: OLED Screen

Fingerprint Scanner

As for the fingerprint scanner we are using R307 Optical Fingerprint Scanner. It is a widely available scanner that we can easily find in the market. It can be integrated easily with the device and has a fair success rate.

Fig. 3.4: Fingerprint Scanner

IFTTT

If This Then That (IFTTT) is a private commercial company that runs services that allow a user to program a response to events in the world. We are using IFTTT to keep a log in the Google Sheets about the successful authentication of fingerprints and sending SMS to the registered mobile numbers after the successful casting of the vote.

Fig. 3.5: IFTTT

Google Sheets

Google sheets are used as it is a free tool and in it, we are storing the database of the voters. Also, we can see whose fingerprints are successfully authenticated and at what time. The phone numbers stored in that database will be the official and registered mobile number of the voter where an SMS will be sent.

Buttons

2-pin push buttons will be used as the physical buttons present in an EVM. Here the user presses the button of their choice to cast the vote in favor of their favorite party.

Fig. 3.6: Buttons

SMS Sender

A third-party SMS sender will be used to send an SMS to the user after he/she has cast the vote successfully. This is done to assure the user that his vote has been successfully registered and also acts as a security measure so that someone else can't cast the vote.

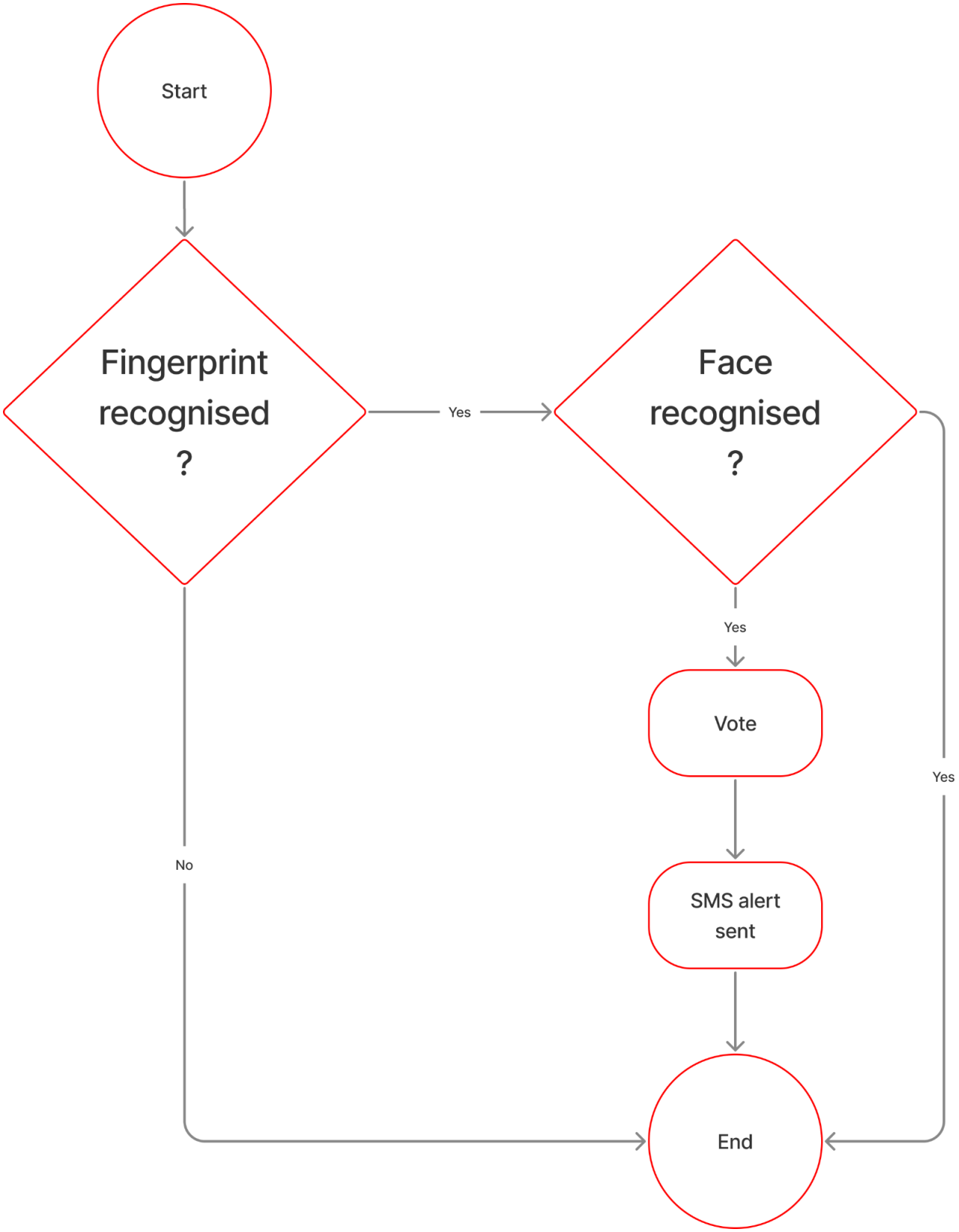

Fig. 3.7: Flowchart

Chapter-4: Design/Simulation

In this chapter, we will discuss the interfacing of the ESP-32 development board with the R307 fingerprint sensor module, OLED screen and other components.

Interfacing of the ESP-32 development board with the R307 fingerprint sensor

Let's first discuss the interfacing of the ESP-32 board with the fingerprint sensor module R307. We have connected these two components as per the image shown below.

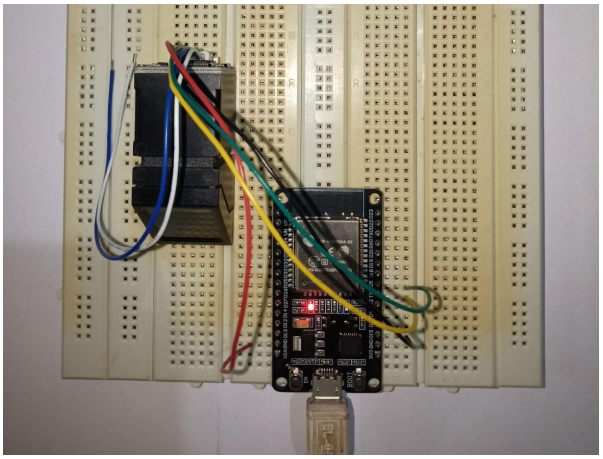

Fig. 4.1: Fingerprint enrollment setup

The 5V pin of R307 is connected to the Vin pin of the ESP-32 development board; the ground pin of R307 is connected to the ground pin of the ESP-32 development board; the RX2 and TX2 pins of the ESP-32 board is connected to the Tx and Rx pins of the R307 fingerprint sensor module respectively.
This setup has been used to enroll sample fingerprints of the voters by us. R307 has an inbuilt memory that can store up to 1000 unique fingerprints. Here we are using the inbuilt memory to store the sample fingerprints of our team members.

To enroll fingerprint we have to install the Adafruit fingerprint sensor library in our Arduino IDE software.
The R307 fingerprint sensor is communicating with the hardware serial ports of the ESP-32 board. For this reason, we need to include the “Hardwareserial” library. Here we are using serial2 of the ESP-32 development board.
After selecting the right port and board in the Arduino IDE we have uploaded the enrollment code in the ESP-32.

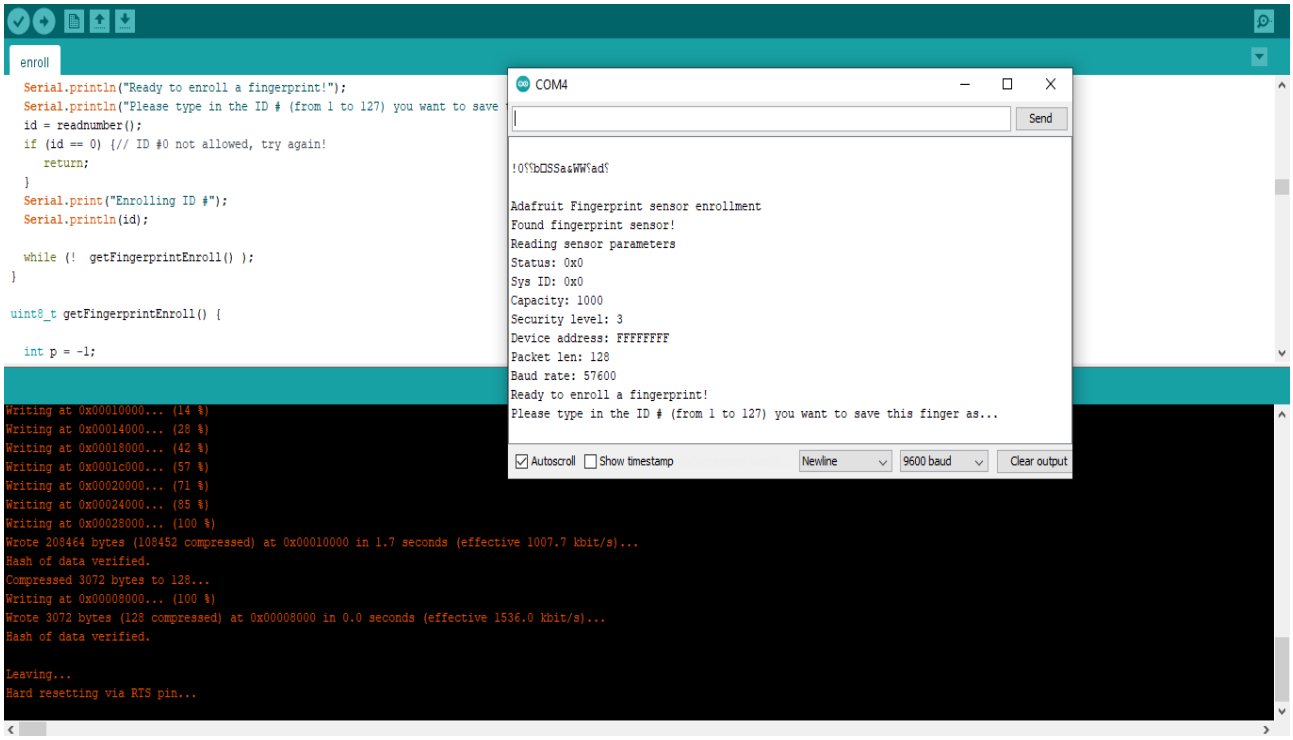

Fig. 4.2: Starting fingerprint enrollment (Arduino IDE interface)

As per the code, we can upload up to 128 unique fingerprints corresponding to ID-0 to ID-127. Let's upload Priyanshu's fingerprint sample in ID-1.

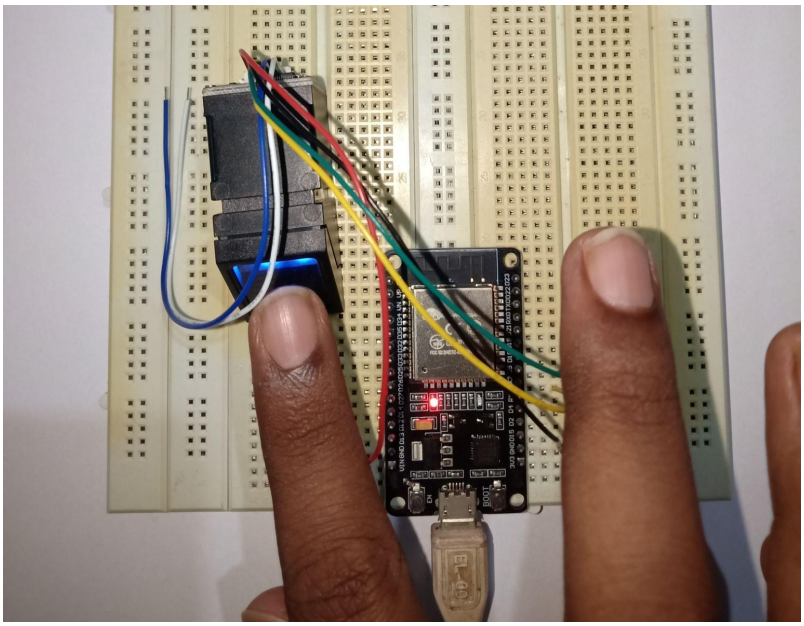

Fig. 4.3: Enrolling Priyanshu's fingerprint as ID-1

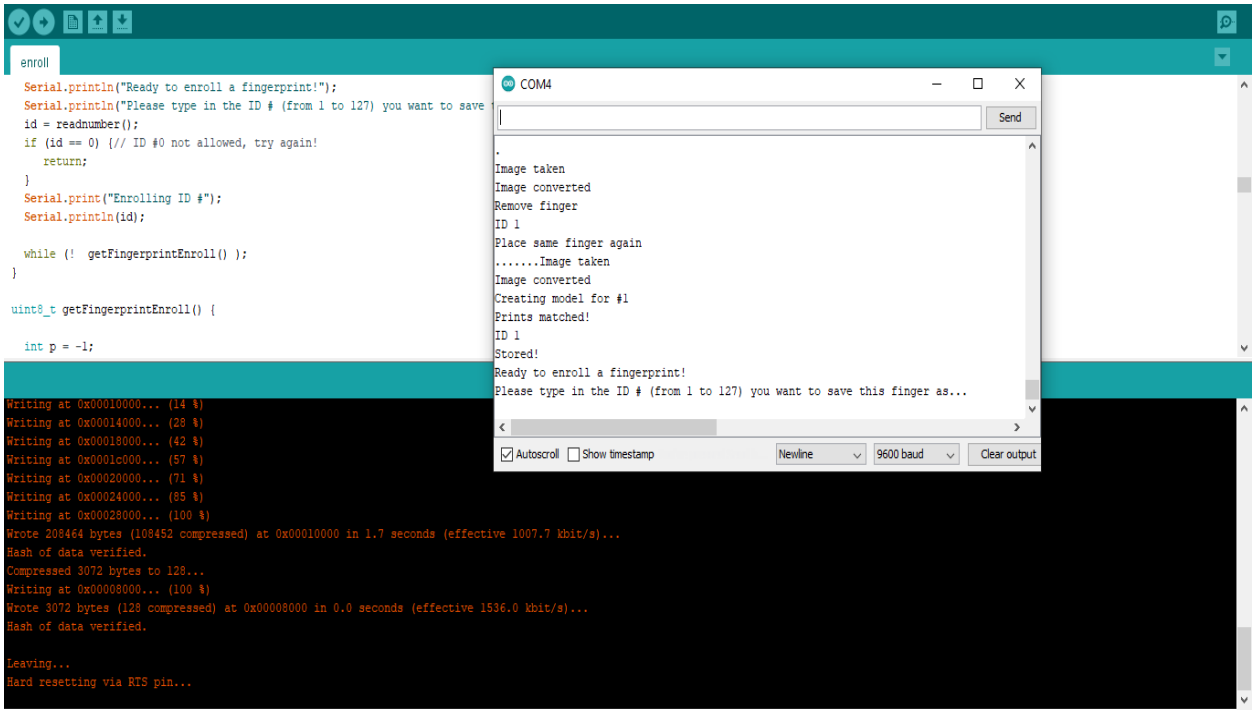

Fig. 4.4: Enrolling Priyanshu's fingerprint as ID-1 (Arduino IDE interface)

We can see that the fingerprint has been successfully stored in ID-1. Now let's store the fingerprint of our other group member Sayan in ID-2. After entering '2' as input in the serial monitor interface, Sayan placed his finger on the fingerprint scanner and the image was taken successfully. We can see this in the below images.

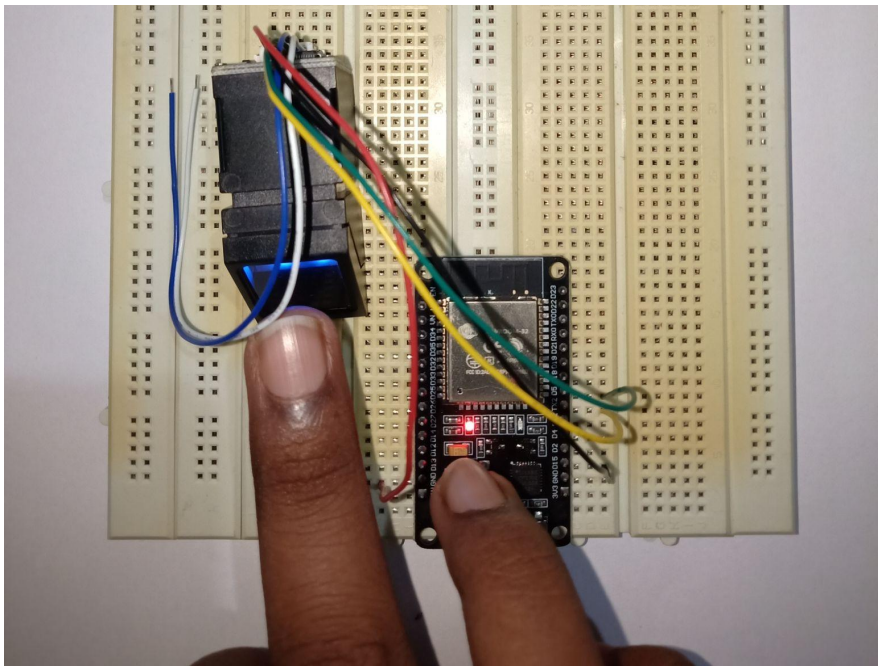

Fig. 4.5: Enrolling Sayan's fingerprint as ID-2

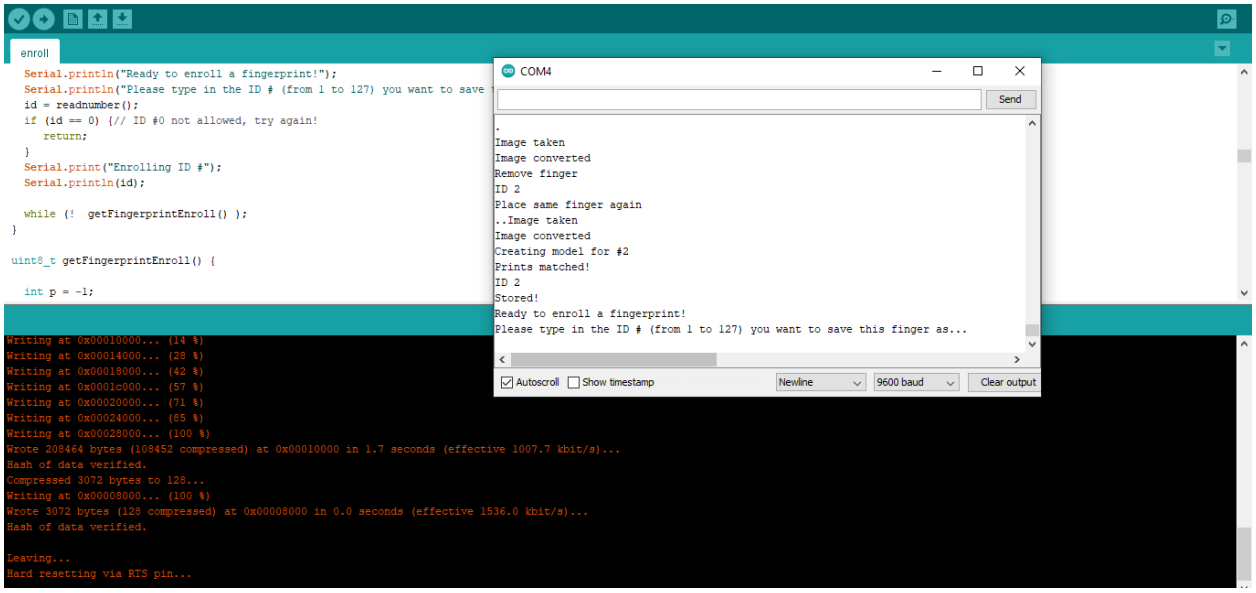

Fig. 4.6: Enrolling Sayan's fingerprint as ID-2 (Arduino IDE interface)

We can also check if the uploaded fingerprints are working or not with the fingerprint checking code provided by Adafruit. Now coming to the uploading of data to Google sheets i.e cloud. For this, we are going to take the help of IFTTT web services. To create a new event, we have to go to the ifttt.com website and search for the "If This Then That" block.

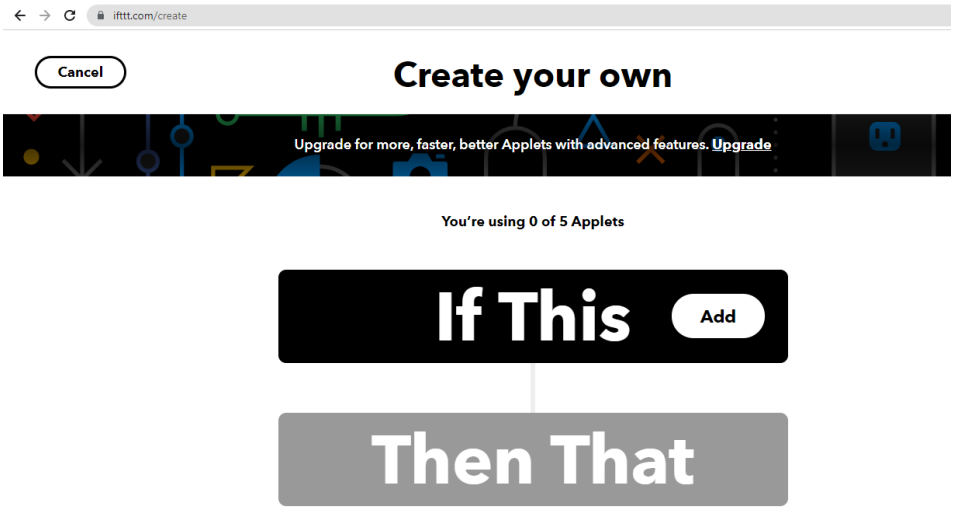

Fig. 4.7: Creating "If This Then That" in IFTTT

For "This" block, we are using "Webhooks". Let's search for "webhooks" in the search bar.

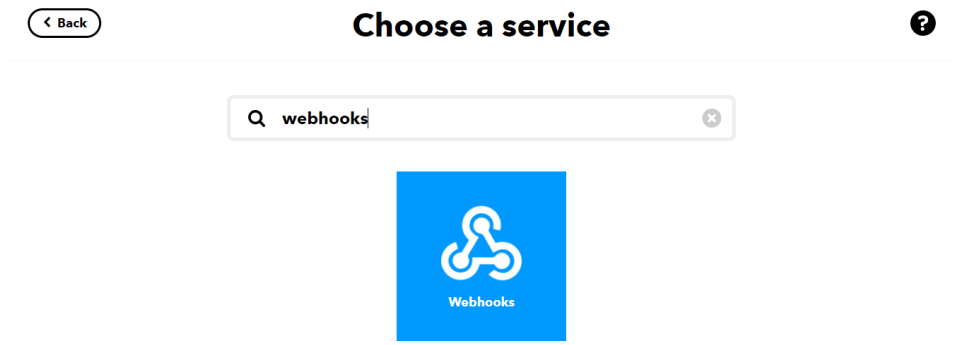

Fig. 4.8: Searching for Webhooks for "If" block

Now we have to create a web trigger. For this, we have to create an event. Here our event name is - "Final Year Project". The picture is attached below.

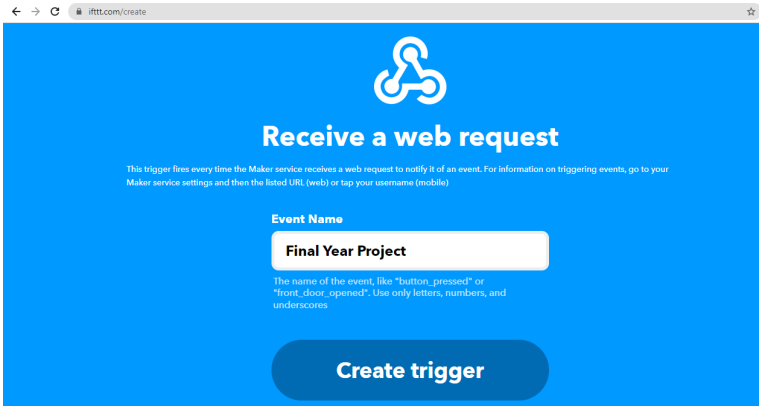

Fig. 4.9: Creating event name in IFTTT

After creating a trigger for the "If" block, we need to set up the event for the "then block. As we are going to store all the logs in a Google sheet, we are using "Google sheet" in the "Then" block.

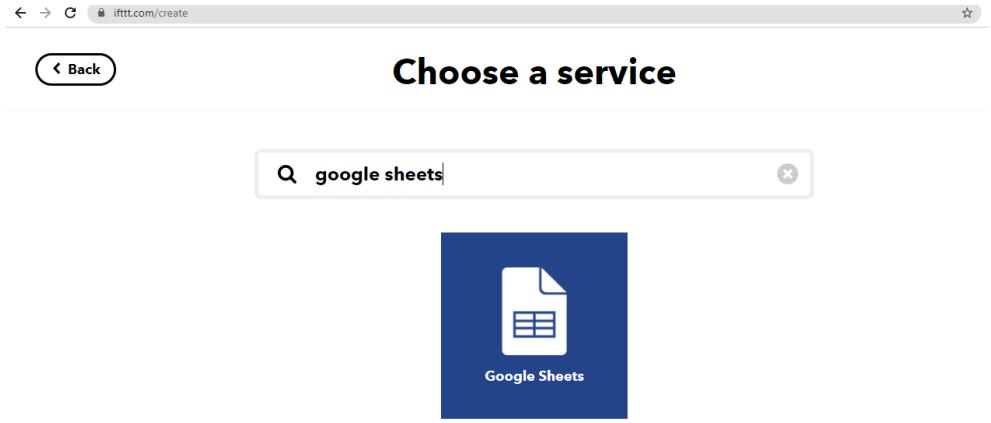

Fig. 4.10: Searching for Google sheets for "Then" block

For every event triggered, we are going to add a new row in the sheet. Now we need to connect our Google account by entering the email ID and password and give permission to IFTTT so that this data can be turned into Google drive. Now we need to enter the spreadsheet name. We have entered "Fingerprint Logs" here. The drive folder path has also been entered.

The integration with the Google sheet step is shown in the image below.

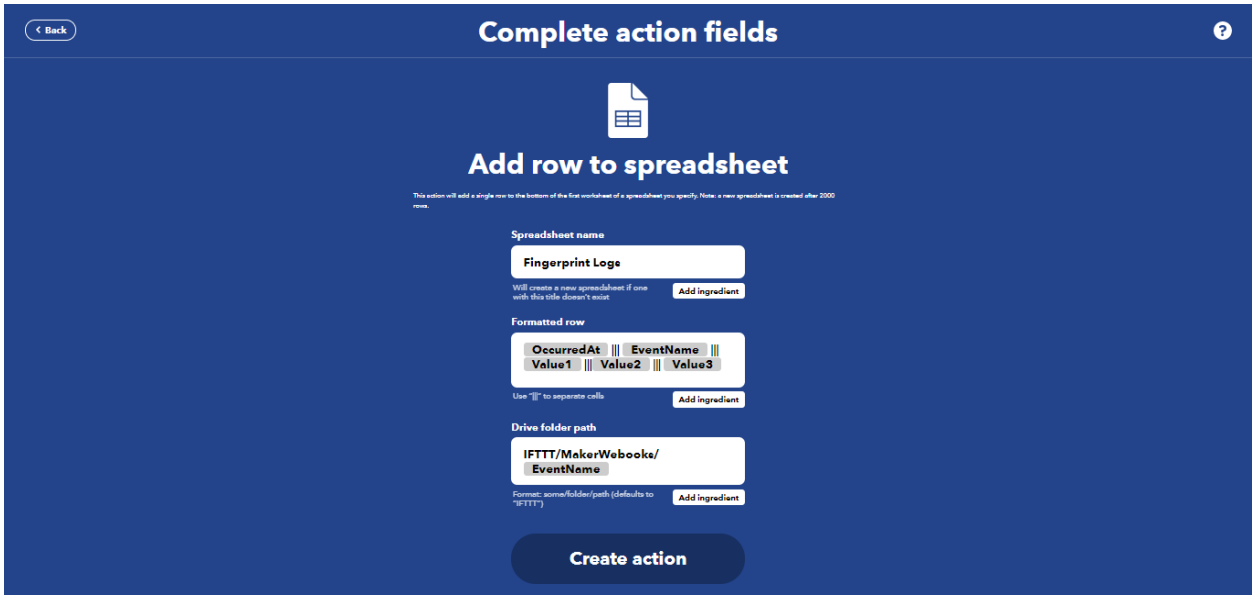

Fig. 4.11: Integrating Google sheet to store fingerprint sensor

Now let's code for fingerprint detection and store the logs into Google sheet. We need to connect the webhooks with the key provided by them. We can get this from the documentation. This key is unique for every webhooks account. We have also provided the wifi username and password so that the ESP-32 development board can connect to the internet. We have coded the name of the user and phone number (for sending automated SMS) corresponding to each fingerprint ID. This information will be sent to the Google sheet when an event occurs successfully.

Now let's upload the codes.

Interfacing of the ESP-32 development board with OLED Screen

Now let's connect the OLED screen module with the ESP-32 development board.

The wiring connection has been done as per the below table.

ESP-32 Development Board	OLED Panel
3V3	VCC
GND	GND
D22	SCL
D21	SDA

Table 4.1: Connection between ESP-32 development board and OLED screen module

The image of the connection is also attached.

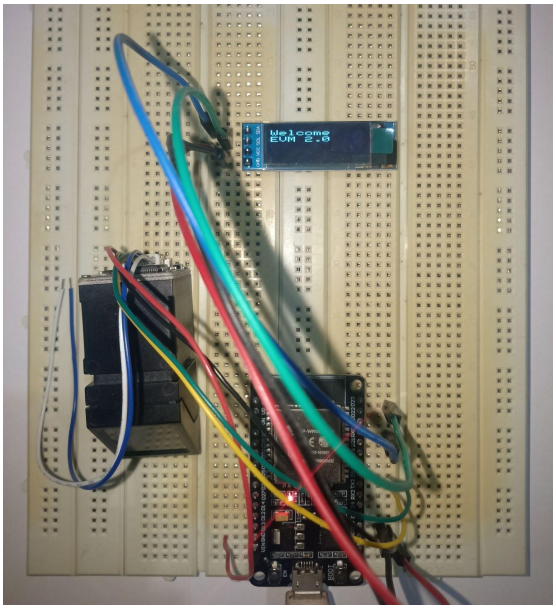

Fig. 4.12: Interfacing with OLED screen

The appropriate library and code have been installed in our ESP-32 board to make our system more user-friendly.

Face Detection and SMS Sending

For face detection we are using the Webcam of our laptop. We are running a simple Python program built with Opencv. When a voter comes to cast his/her vote, the face is compared with the stored samples using the Python program.

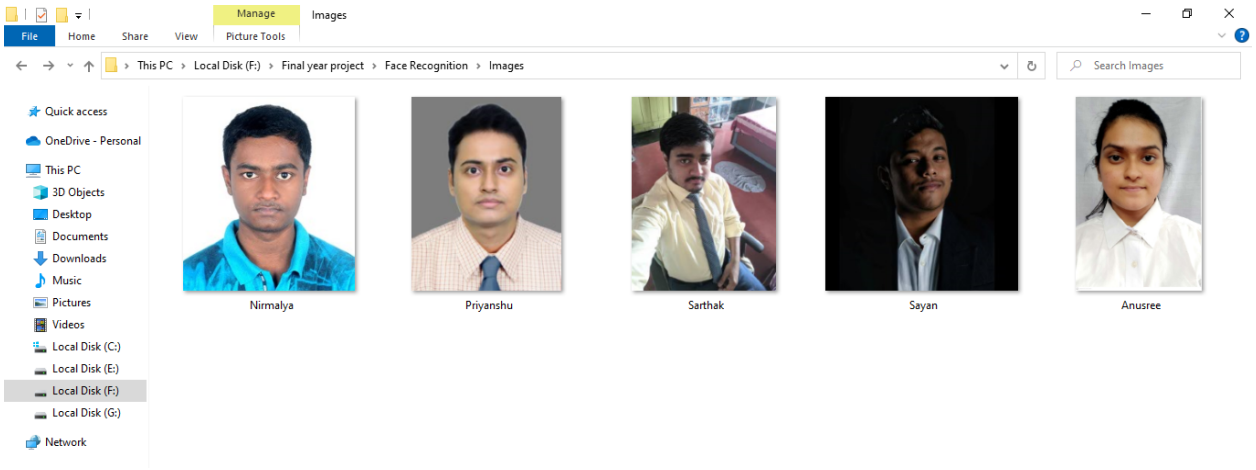

Fig: 4.13: Face Database

We are using Twilio API to send SMS. The code is written in the Python program itself. We are using this API because it is free.

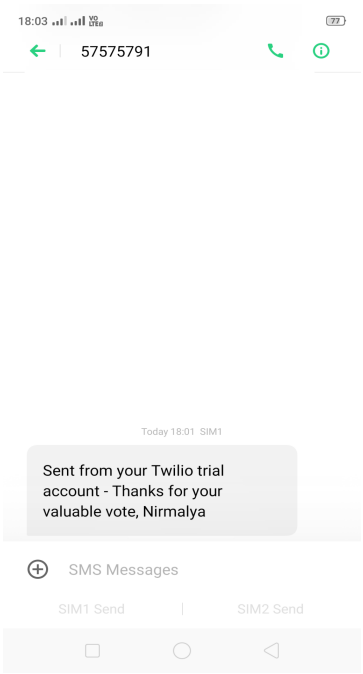

Fig: 4.14: Confirmation SMS sent using Twilio API

Full circuit diagram

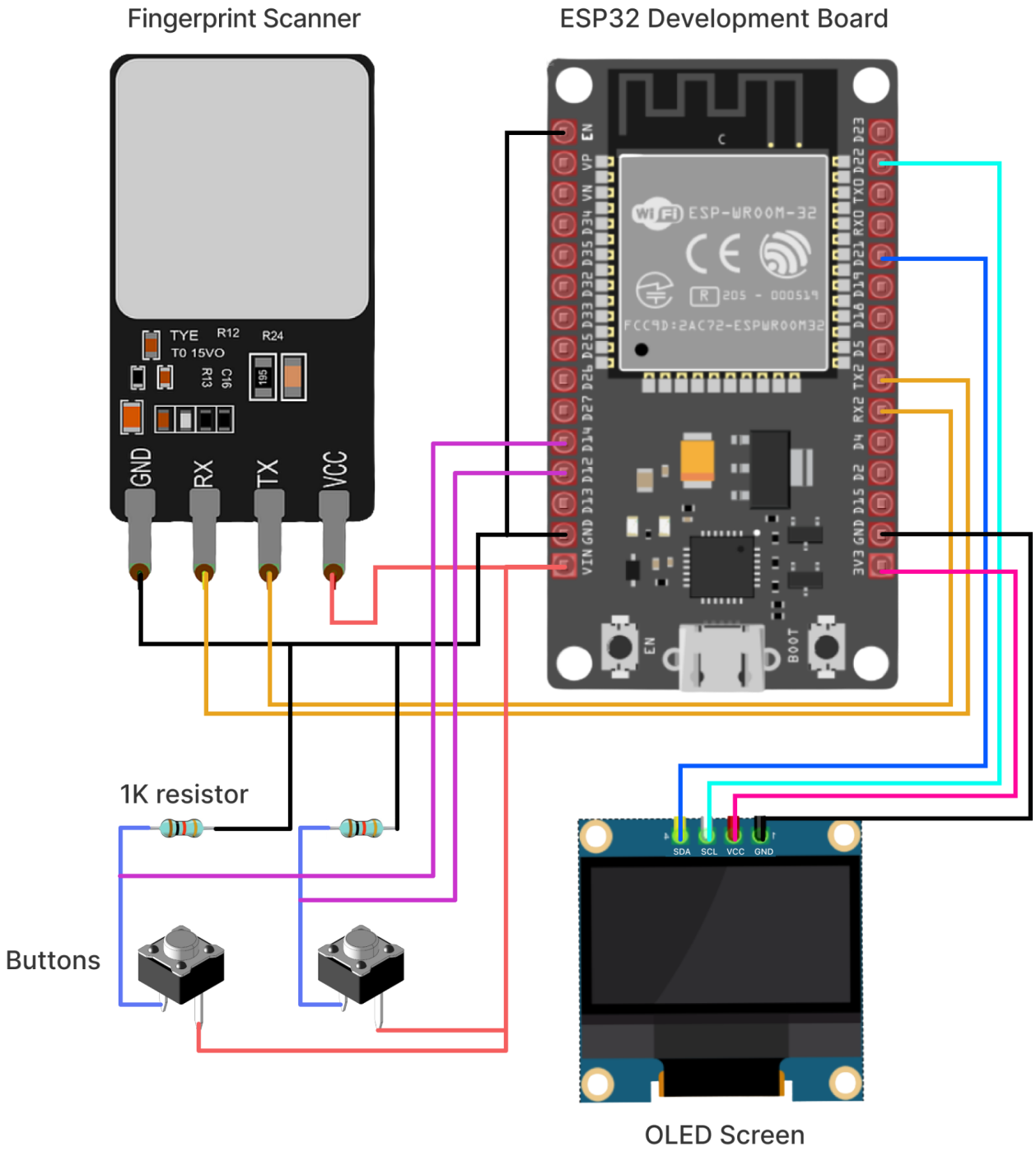

Fig. 4.15: Circuit Diagram

Now, let's look into the details of the components used.

Component Name	Specifications	Quantity	Price ₹
ESP32-Dev Board	• CPU: Xtensa dual-core (or single-core) 32-bit LX6 microprocessor, operating at 160 or 240 MH • Memory: 320 KB RAM, 448 KB ROM • 34 programmable GPIOs • 12-bit SAR ADC up to 18 channels • 2 × 8-bit DACs • Power: 3.3 V DC	1	550
OLED Screen	• Display: 2-lines x 16-characters • LCD Controller: HD44780 • Pin Definition: VCC, GND, SDL, and SCA • Working Voltage: 5V	1	200
Jumper wires	-	-	100
Push-buttons	-	2	70
Fingerprint Module	• Supply Voltage: DC 3.3V • Operating current: <60mA • Fingerprint image time: <1.0 seconds • Window area: 15.3 x 18.2 mm • Resolution: 500dpi	1	1900
Laptop Webcam	-	-	-
Google Sheets	-	-	-
SMS sender	-	-	-
IFTTT	-	-	-

Table 4.2: Component List

Chapter-5: Result & Performance Analysis

In this chapter, we are going to discuss the testing results that we have got while performing the performance analysis of our system.

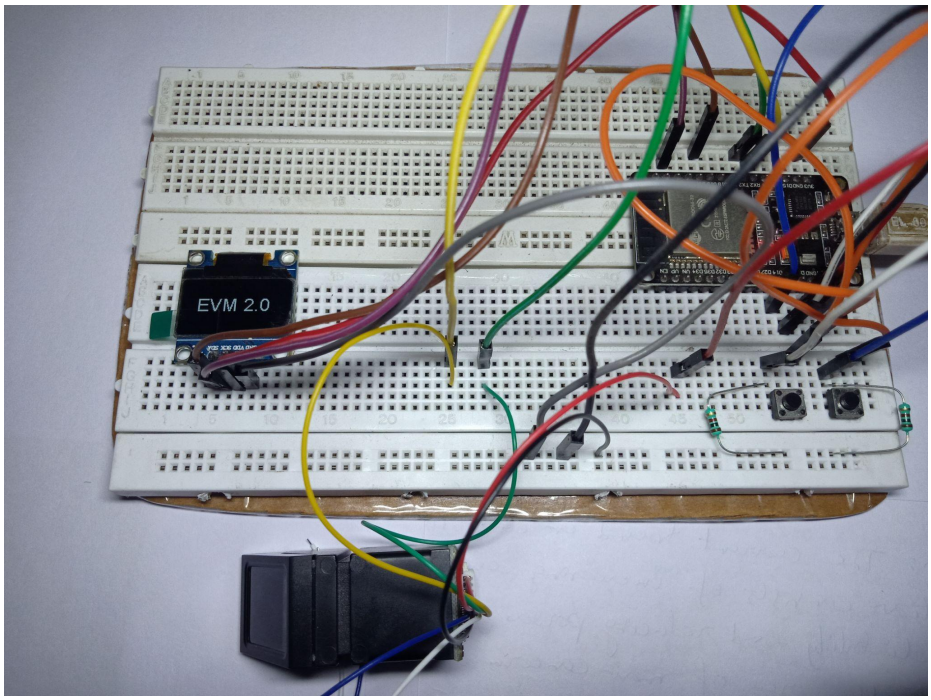

Fig. 5.1: Full set up

First we asked Nirmalya to cast his vote. In the below image we can see “Place Finger” is displaying on the OLED screen.

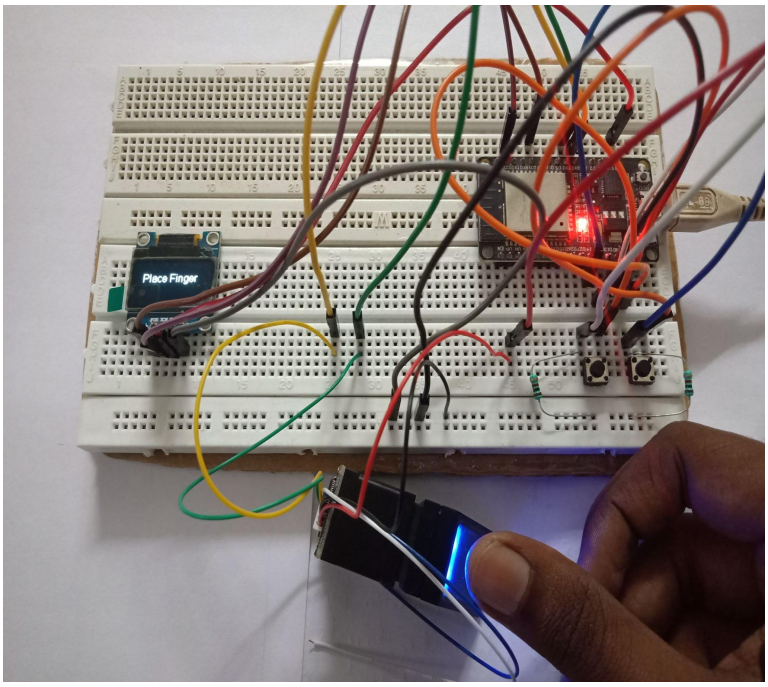

Fig. 5.2: Nirmalya is placing his finger on the R307 sensor

After Nirmalya placed his finger, his fingerprint was successfully verified by the R307 scanner and an appropriate message was also shown. We can find it in the below image.

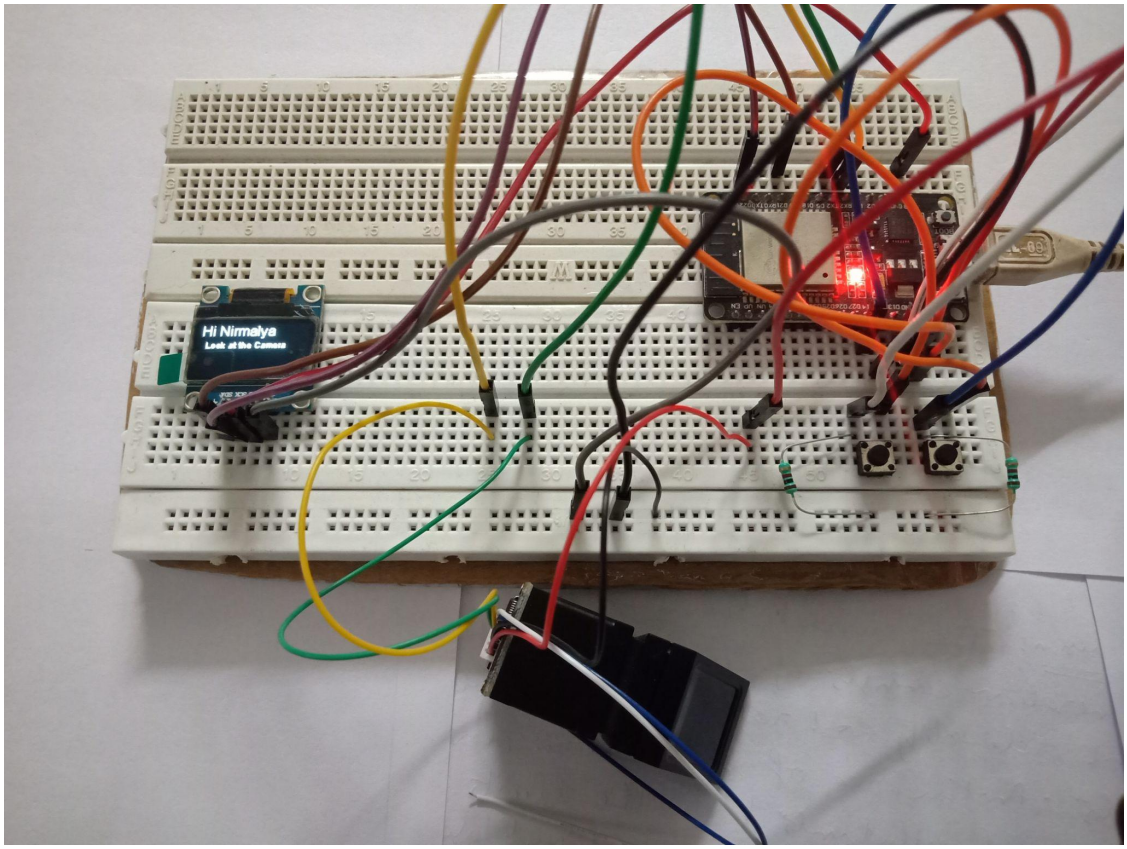

Fig. 5.3: Nirmalya's fingerprint was successfully verified

Now EVM 2.O is asking Nirmalya to look at the Webcam so that his face can be verified.

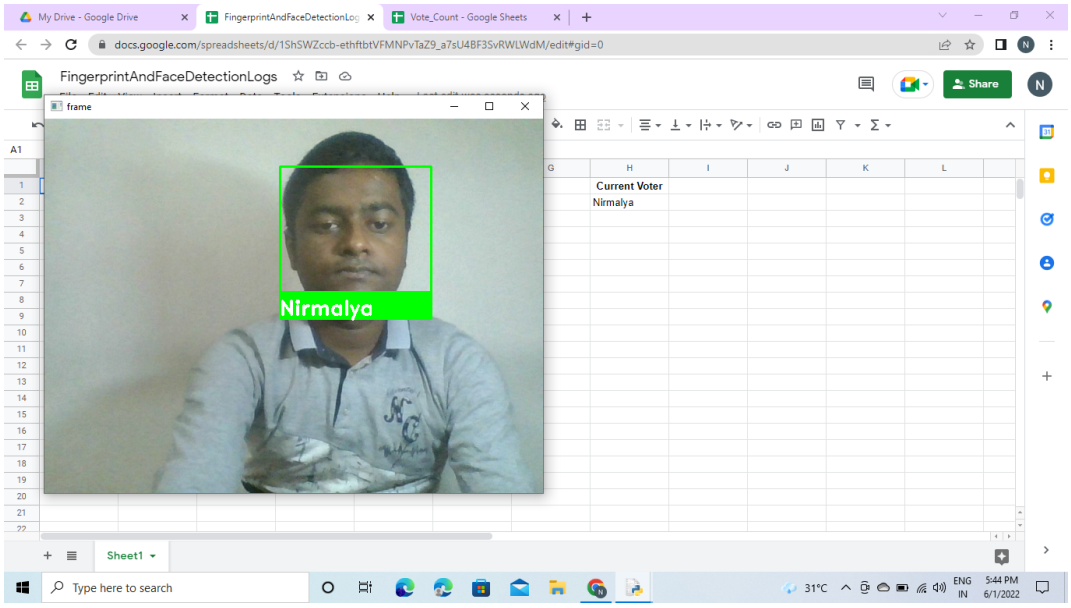

Fig. 5.4: Nirmalya's face detected

Nirmalya's face was successfully verified. The current voter name is also shown in the Google sheet.

Now Nirmalya is casting his vote by pressing a button.

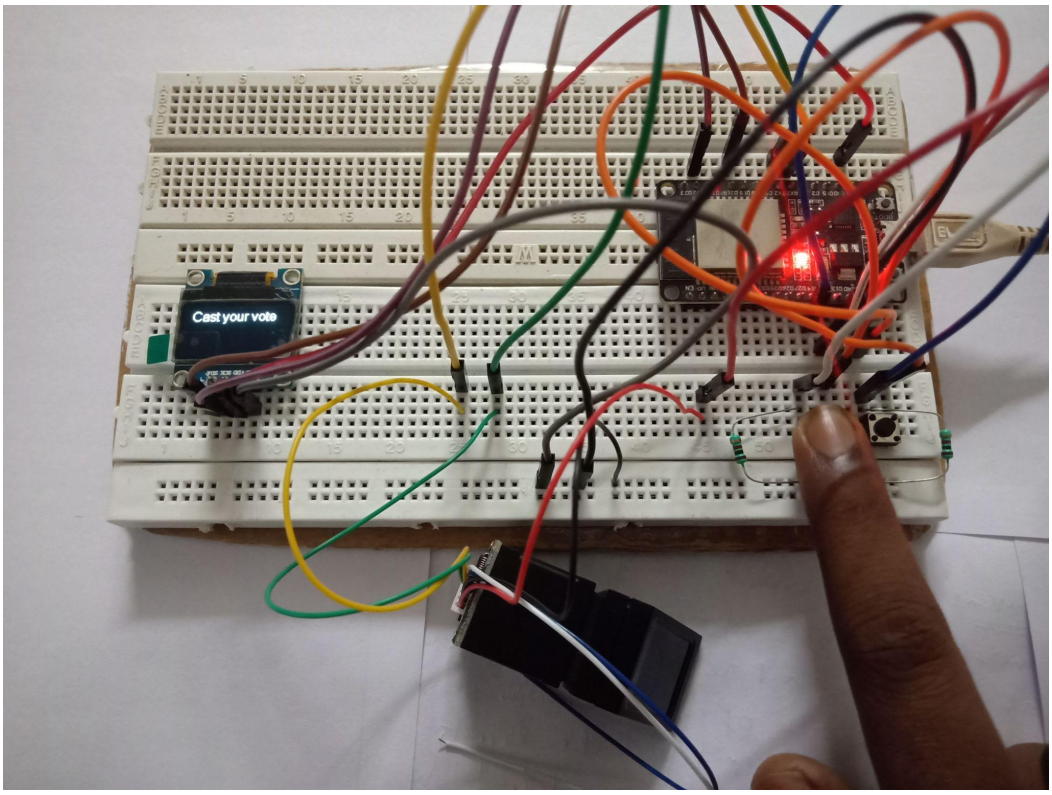

Fig. 5.5: Nirmalya casting his vote

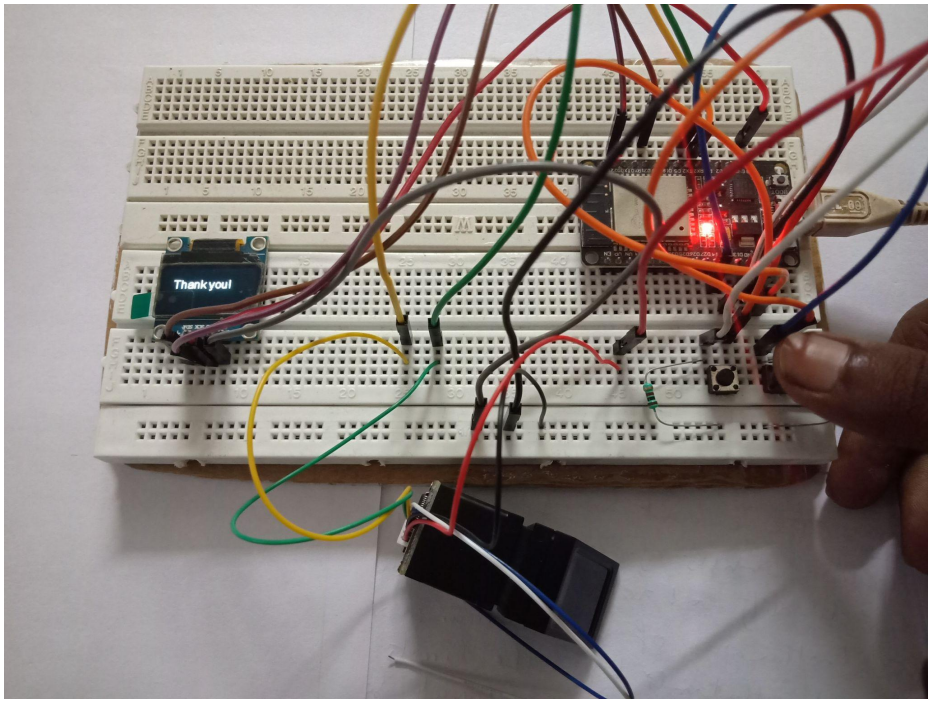

Fig. 5.6: Vote casted successfully

After Nirmalya pressed the button, his vote was recorded successfully in the Google sheet.

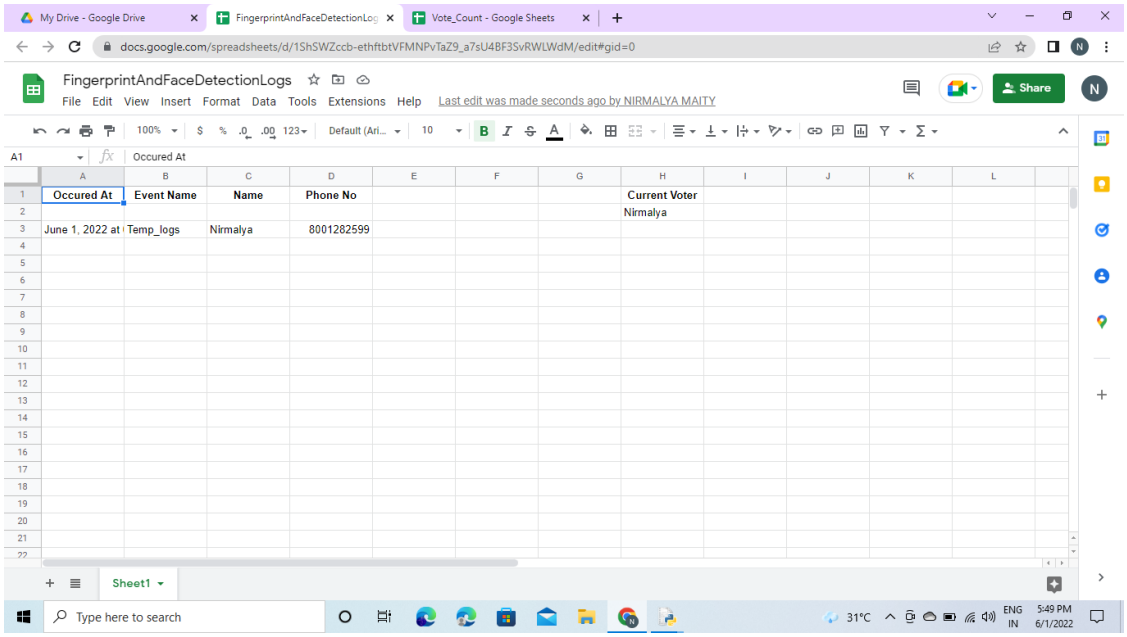

Fig. 5.7: Log recorded in Google sheet

We can see the voter's name and phone number along with date and time have been recorded in the log sheet. The vote count is recorded in another Google sheet that is shown in the below image.

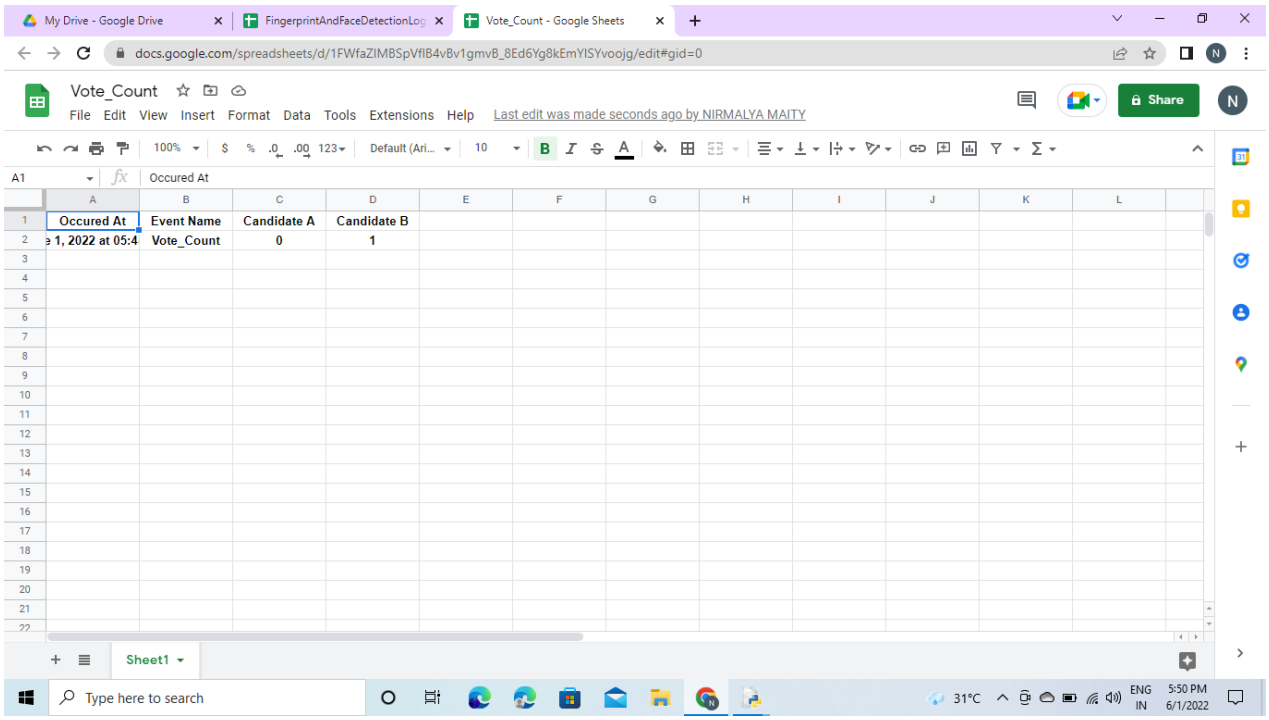

Fig. 5.8: Vote count log

After all these things an SMS was sent by the EVM 2.O to Nirmalya's mobile number.

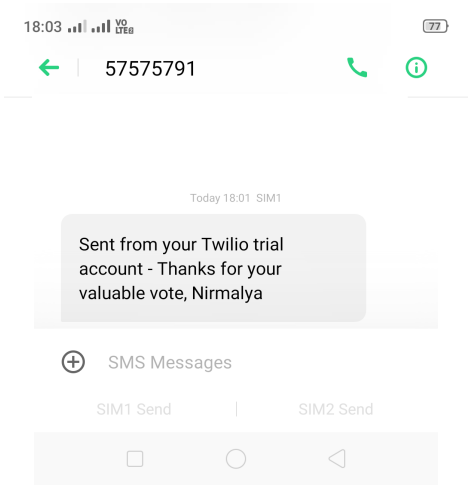

Fig. 5.9: Confirmation SMS

In this way our other group member Sayan casted his vote and the data was recorded in the google sheet.

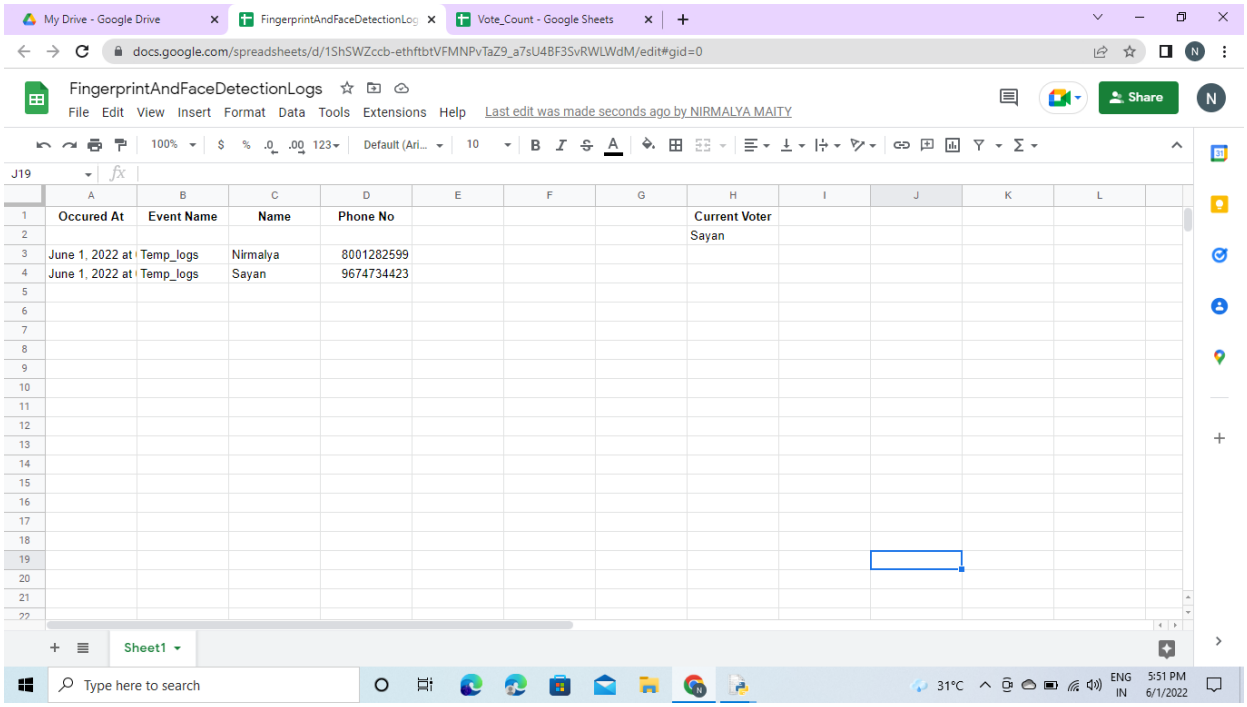

Fig. 5.10: Log after Sayan's vote

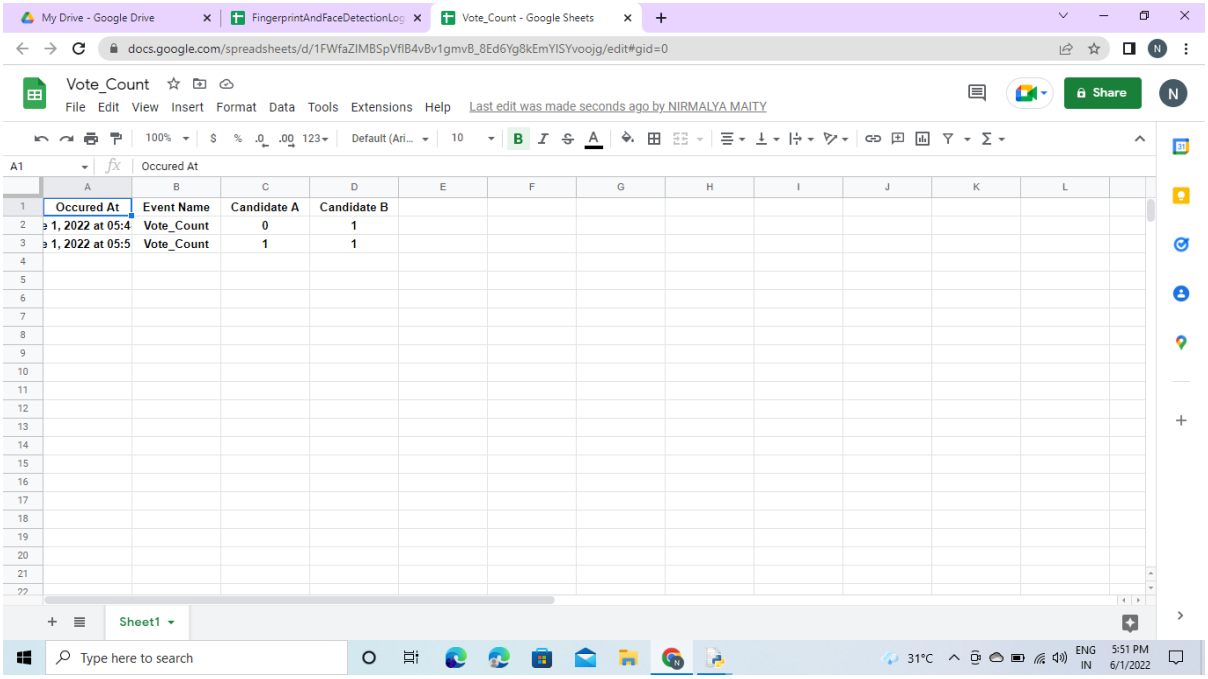

Fig. 5.11: Vote count log

Now we asked Priyanshu to cast his vote. His fingerprint was matched somehow, but his face didn't match. So he was not able to cast his vote.

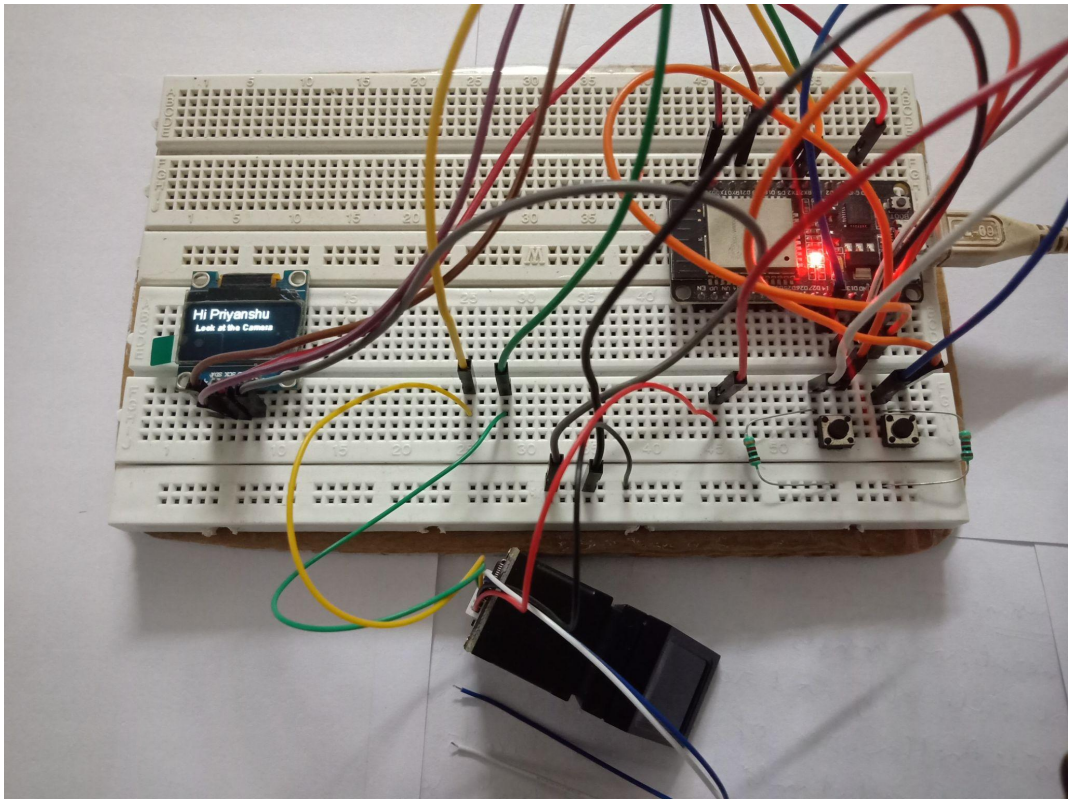

Fig. 5.12: Priyanshu's fingerprint matched

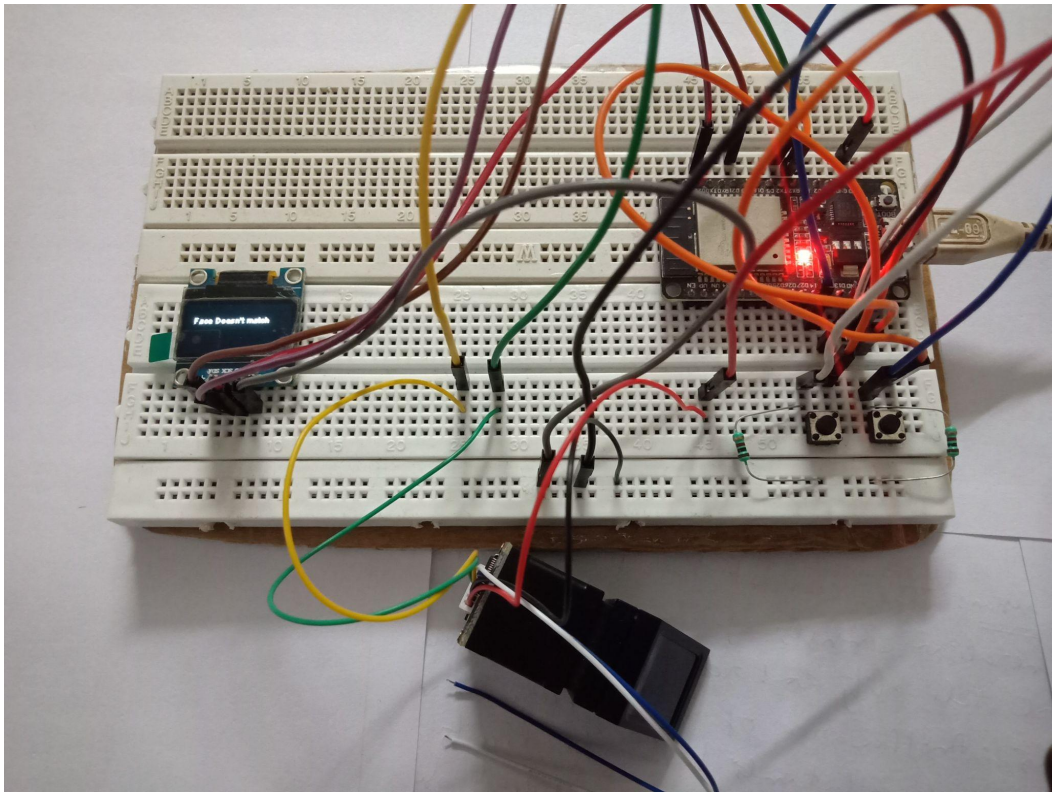

Fig. 5.13: Priyanshu's face didn't match

So we can say that EVM 2.O is working perfectly fine as per the goal of our project.

Chapter-6: Future work and Conclusion

In this paper, we have described the specification and architecture of the ELECTRONIC VOTING MACHINE (EVM). Various fault-tolerance and security issues are delegated to the platform itself, therefore relieving the application designer from accommodating these features in the application design itself. This approach allows for the easy development and deployment of applications. For quite some time, voting equipment vendors have maintained that their systems are secure and that the closed-source nature makes them even more secure. Our glimpse into the code of such a system reveals that there is little difference in the way code is developed for voting machines relative to other commercial endeavors.

We believe that an open process would result in more careful development, as more scientists, software engineers, political activists, and others who value their democracy would be paying attention to the quality of the software that is used for their elections. (Of course, open-source would not solve all of the problems with electronic elections. It is still important to verify somehow that the binary program images running in the machine correspond to the source code and that the compilers used on the source code are non-malicious. However, open-source is a good start.) Such open design processes have proven successful in projects ranging from focused efforts, such as specifying the Advanced Encryption Standard (AES) to very large and complex systems such as maintaining the Linux operating system. Australia is currently using an open-source votingsystem10 Alternatively, security models such as the voter-verified audit trail allow for electronic voting systems that produce a paper trail that can be seen and verified by a voter. In such a system, the correctness burden on the voting terminal's code is significantly less as voters can see and verify a physical object that describes their vote. Even if, for whatever reason, statements that they would support such features if their customers required it.

The EVM projection ambitious attempt to create an open-source voting system with a voter-verifiable audit trail — a laudable goal the model where individual vendors write proprietary code to run our elections appears to be unreliable, and if we do not change the process of designing our voting systems, we will have no confidence that our election results will reflect the will of the electorate. Our system and our future to have robust, well-designed election systems to preserve the bedrock of our democracy.

References

1. The Impact of Electronic Voting Machines on Electoral Frauds, Democracy, and Development - By Debnath, Sisir & Kapoor, Mudit
2. Case Studies on E-Voting - University of Fribourg
3. Secure large-scale E-voting system based on blockchain contract using a hybrid consensus model combined with sharing - By Yousif Abuidris, Rajesh Kumar, Ting Yang & Joseph Ongoing
4. Face Recognition as a Genuine Technique in Electronic Voting - By Abhishek Jain, Ayush Srivastava & Amritesh Patidar
5. Smart Voting System Support through Face Recognition - By Vetrivendan Lakshmanan & Viswanathan Ramasamy
6. A smart voting system using Fingerprint Scanner - By S. Charan, K. Hari Prashant & D. Anand Joseph Danie
7. D. Maio, D. Maltoni, A. K. Jain, and S. Prabhakar (2003). Handbook of Fingerprint Recognition. Springer Verlag.
8. Jain, K, Ross, A, Prabhakar, S (2004). An Introduction to Biometric Recognition. New York, USA.
9. Farah Azirar, (2011). Fingerprint Recognition. Bachelor Thesis. School of Electronics and Physical Sciences, Department of Electronic Engineering,
   University of Surrey.
10. Jawad Nagi, (2009). Design of an Efficient High-speed fingerprint Recognition System.
11. A. Ross, S. Dass, and A. K. Jain (2010). A deformable model for fingerprint matching, Pattern Recognition. The University of Michigan.

Appendix

1. AES: Advanced Encryption Standard
2. BLE: Bluetooth Low Energy
3. EN: Enable
4. IFTTT: If This Then That
5. GND: Ground
6. I2C: Inter-Integrated Circuit
7. IDE: Integrated Development Environment
8. OLED: Organic Light Emitting Diode
9. RFID: Radio Frequency Identification
10. Rx: Receiver
11. SCL: Serial Clock
12. SDA: Serial Data Pin
13. SPI: Serial Peripheral Interface
14. PWM: Pulse Width Modulation
15. TTL: Transistor-Transistor Logic
16. Tx: Transmitter
17. UART: Universal Asynchronous Receiver-Transmitter
18. USB: Universal Serial Bus
19. VCC: Voltage Common Collector
20. VVPAT: Voter Verifiable Paper Audit Trail