EVM 2.0

Abstract

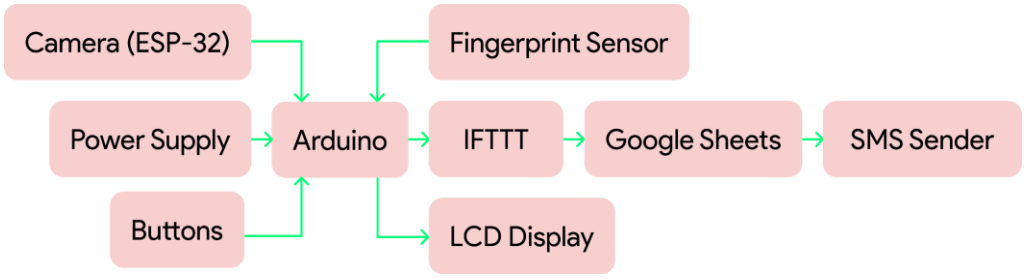
Election plays a major role in the political process of democracy. However, if the election is corrupted in the process itself, it is clear that the democracy based on that election may be fraudulent. Therefore, the cheating strategies in the election will destroy the democratization that the citizens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

Introduction

Previously, in several constituencies under the paper ballot system, polling booths would be captured, and ballot boxes would be stuffed. 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 own limitations but also introduces a lot of friction in the process, causing delay and a whole host of other problems such as EVM tampering.

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.

Block Diagram



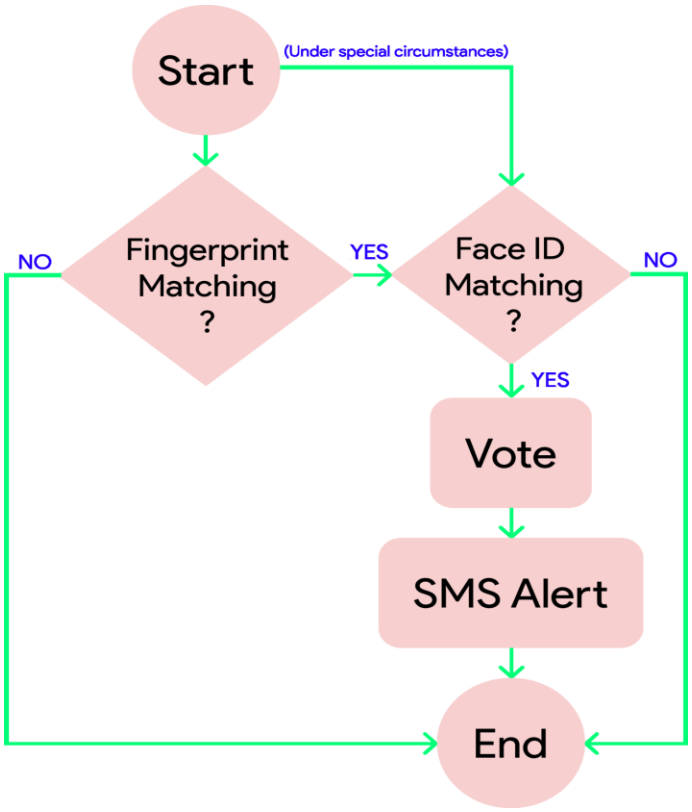
Working

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 from 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 Sheets (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 ESP32-CAM Module. When a user comes in front of the camera, it detects the face and will match with the enrolled faces. We can enroll the faces using the localhost IP address provided by the ESP32-CAM Module after successful setup. 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 flowchart as well as in the block diagram.

Flow Chart



Components' List

The total cost is estimated to be ₹3660.

Components' name	Specifications	Quantity	Price ₹
Arduino UNO R3	• ATmega328P Microcontroller • Input voltage 7-12V • 5V Electric current: 500MA • 3.3V Electric current: 50MA • 14 Digital I/O Pins (6 PWM outputs) • 8 Analog Inputs • 32KB Flash Memory • 16MHz Clock Speed	1	550
LCD Display	• Display: 2-lines x 16-characters • LCD Controller: HD44780 • Pin Definition: VCC, GND, SDL and SCA • Default Address: 0x27 for PCF8574T chip; 0x3F for PCF8574AT chip • Working Voltage: 5V • Backlight Adjust: Jumper	1	200
Wires	-	-	100
Push-buttons	-	4	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
ESP32-CAM Module	• Ultra-small 802.11b/g/n Wi-Fi + BT/BLE SoC module • Low-power dual-core 32-bit CPU for application processors • Up to 240MHz, up to 600 DMIPS • Built-in 520 KB SRAM, external 4M PSRAM • Supports interfaces such as UART/SPI/I2C/PWM/ADC/DAC • Supports OV2640 and OV7670 cameras with built-in flash • Support for images Wi-Fi upload • Embedded lwip and FreeRTOS • Supports STA/AP/STA+AP working mode	1	840
Google Sheets	-	-	-
Third-party SMS sender	-	-	-
Web-based Service (IFTTT)	-	-	-

Application & Benefits

For the aforementioned setup, the entire process and the tech needed to pull it off is widely available and also economically viable on a large scale. If implemented properly, this can have widespread application across all elections in the country. Benefits will range from making the election process more secure and reducing the chances of forgery. This not only helps in restoring public trust in elections but improves democracy as a whole. While ensuring election integrity, this setup can be easily designed and tweaked to make the voting process easy and more accessible to all across various backgrounds.

Conclusion

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 implementing technologies in an innovative way, for the betterment of all.

References

- The Impact of Electronic Voting Machines on Electoral Frauds, Democracy, and Development - By Debnath, Sisir & Kapoor, Mudit
- Case Studies on E-Voting - University of Fribourg
- 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 Onginjo
- Face Recognition as a Genuine Technique in Electronic Voting - By Abhishek Jain, Ayush Srivastava & Amritesh Patidar
- Smart Voting System Support through Face Recognition - By Vetrivendan Lakshmanan & Viswanathan Ramasamy
- Smart voting system using Fingerprint Scanner - By S. Charan, K. Hari Prashant & D. Anand Joseph Daniel