TWO LEVEL BIOMETRIC SECURITY SYSTEM FOR VOTING

Supervised by - Ms. Bindu E.

Presented by –

Avinash Kumar (00110407317)

Ritika Garg (02210402816)

Snigdha Srivastva (02810402816)

Yashika Kalra (40710402816)

CONTENTS

OBJECTIVE

- Improve the security performance in the voting process.
- Assure the aim of ‘One voter – One vote’ using database.
- Include multiple layers of verifications to ensure the reliability of the voting process.

REQUIREMENTS

Development Environment - Arduino IDE

Languages – C++, SQL, HTML, CSS, Bootstrap

Database – SQL Server

Libraries – Fingerprint Reader SDK, Business Entities SDK

Integration Platform – Wampserver 2.4E, Visual Studio Code

Hardware – Fingerprint Sensor R307, Buzzer, Arduino Uno, NPN Transistor, LM Transistor, Tip 122 Transistor

Online Arduino Simulators – Simuino, Virtronics

Designing Platforms – Figma, Adobe Photoshop

WHAT'S NEW

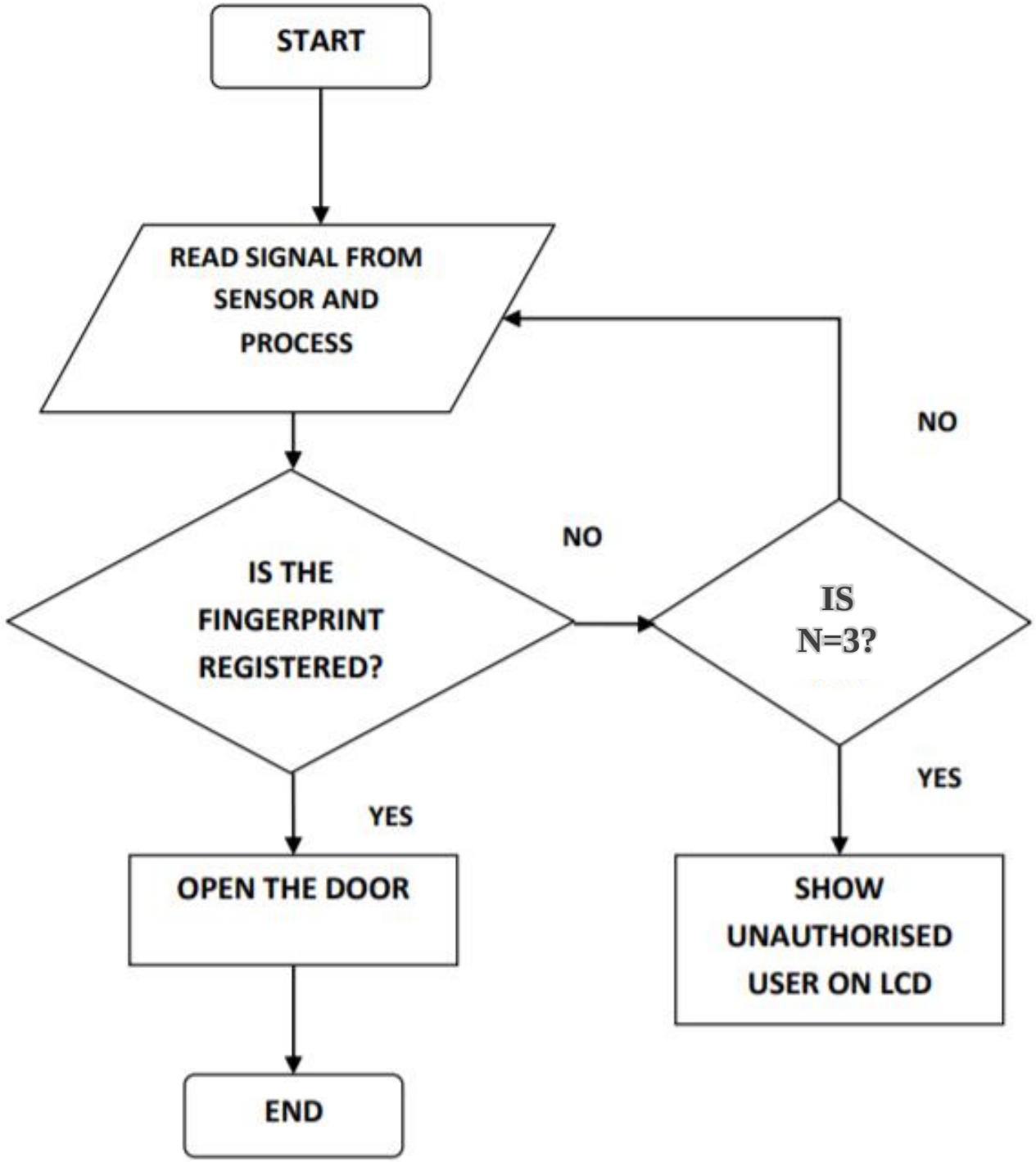
- Buzzer alarm on re-entry for vote.
- 2-step authentication using face recognition at ballot side for secured experience in voting.
- Activity Tracker – use of database to keep track of votes.

ALGORITHM

A) Entry side:

- The voter scans his finger at the entrance gate.
- The fingerprint of the voter is matched with the stored finger ID in the database.
- If the fingerprint is matched, voter is authorized voter, the door will open.
- The voter enters inside the door and the door will close.
- If the same finger is scanned more than once, door doesn’t open and shows error.
- If once fingerprint doesn’t match with database, user gets 2 more chances, if it still doesn’t match LCD shows Unauthorized Access.

FLOW CHART (ENTRY SIDE)

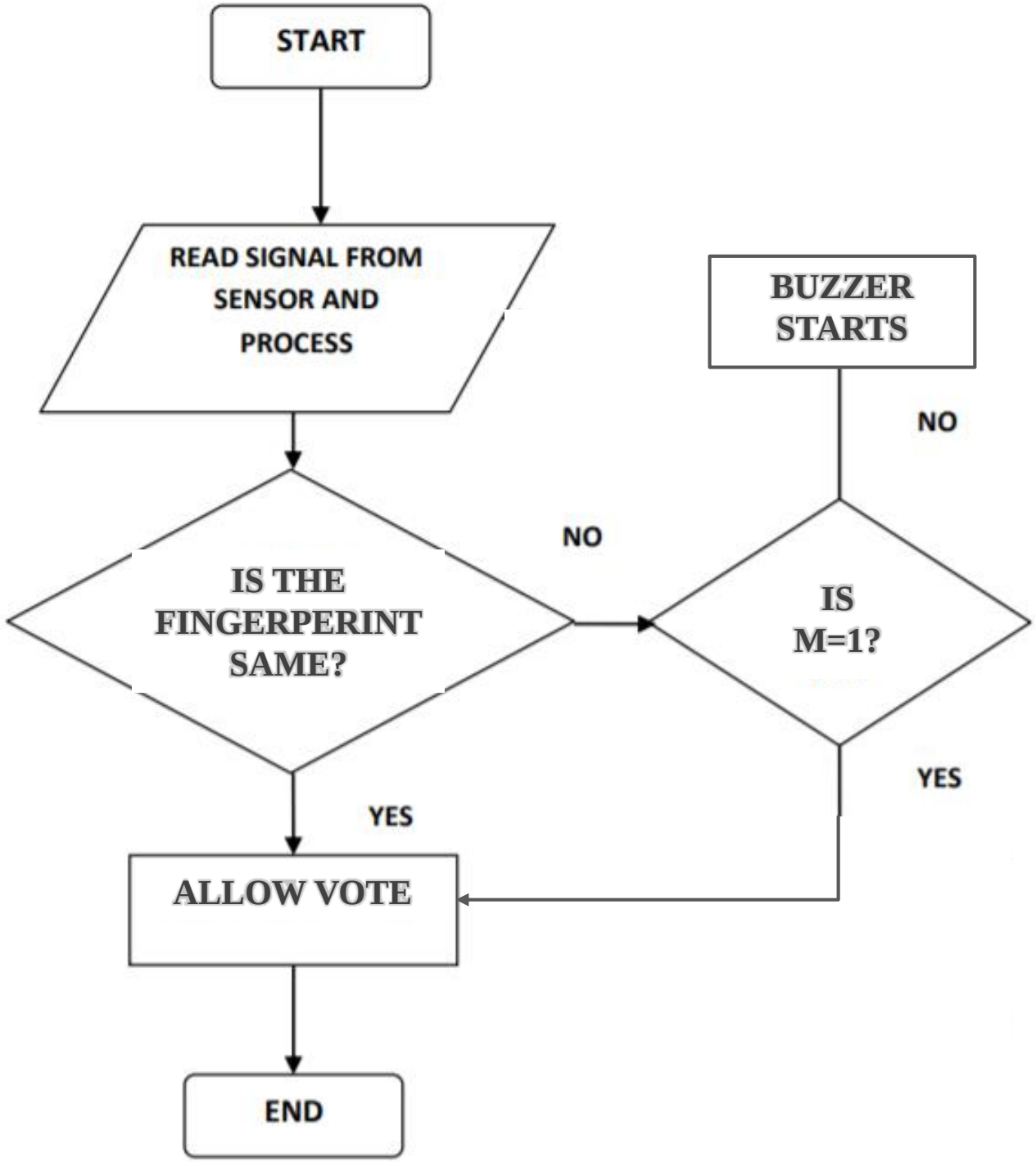


ALGORITHM

B) Ballot side:

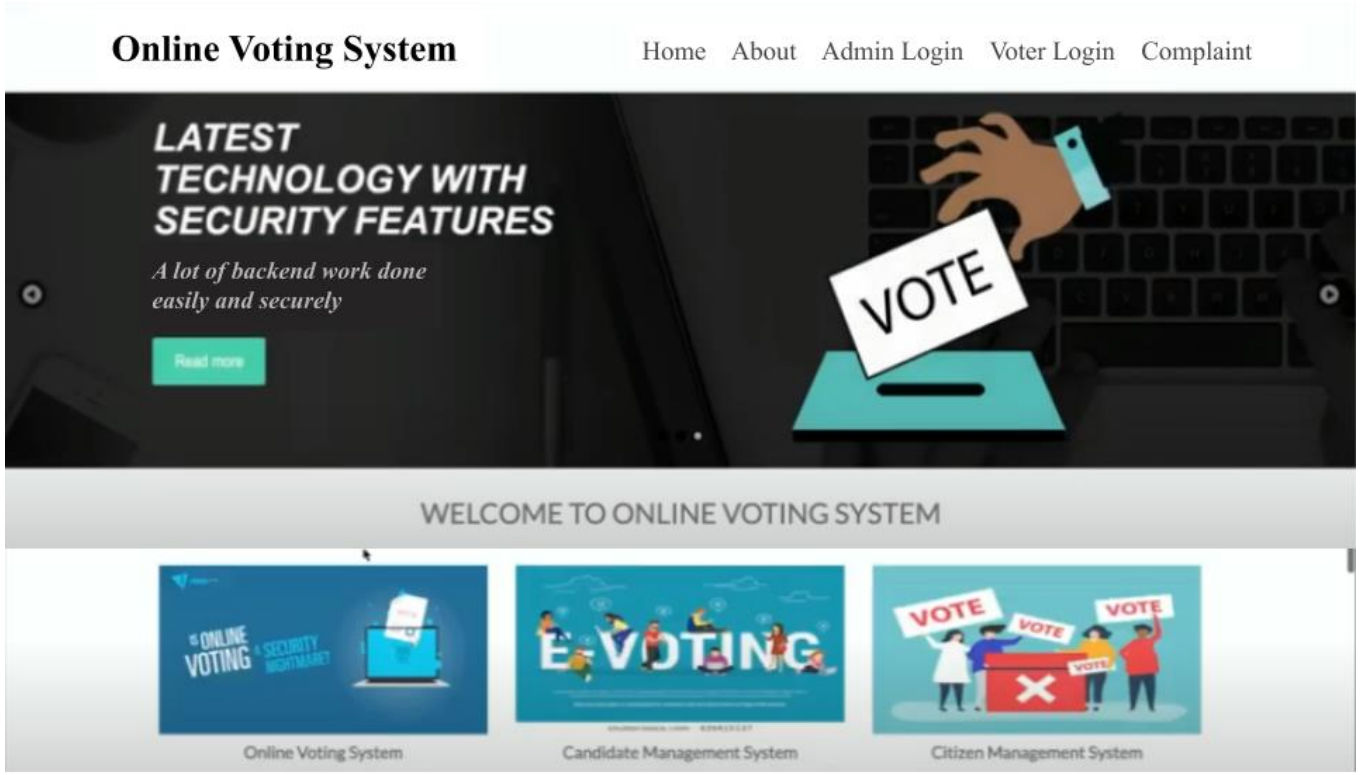
- Voter will be asked to scan finger again on sensor and checks if fingerprint matches with the database.
- After manually scanning the photo with the data record in our system, it will enable the voter to cast his vote.
- If the same finger is scanned more than once, buzzer will start and the person will be caught.

FLOW CHART (BALLOT SIDE)



PROGRESS

- Literature Survey
  - Matching Algorithms
  - Database Integration
  - Online simulators
- Ballot Side Designing

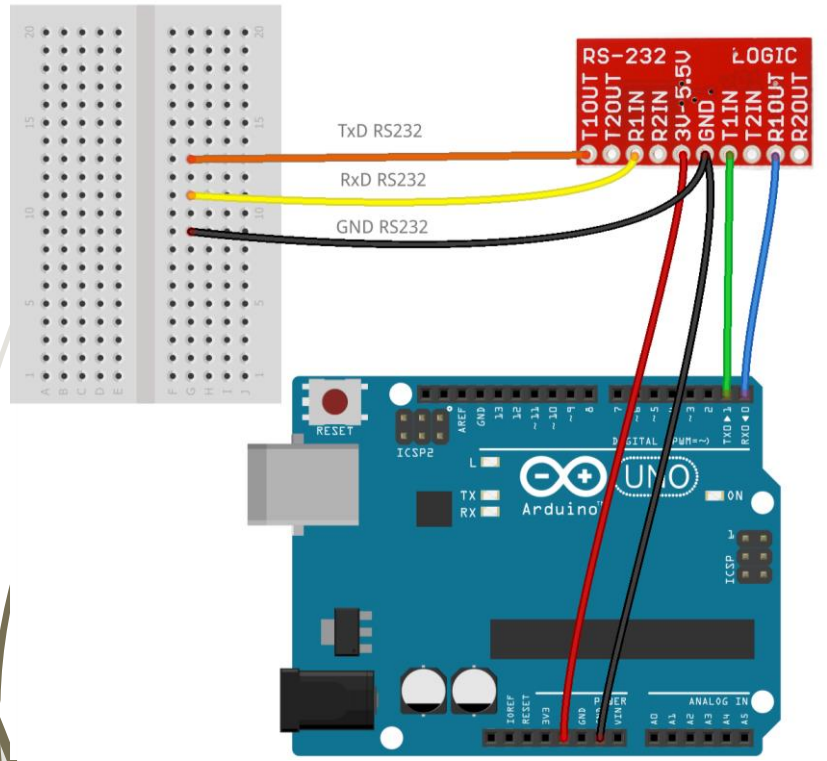


TYPES OF FINGERPRINT ALGORITHMS

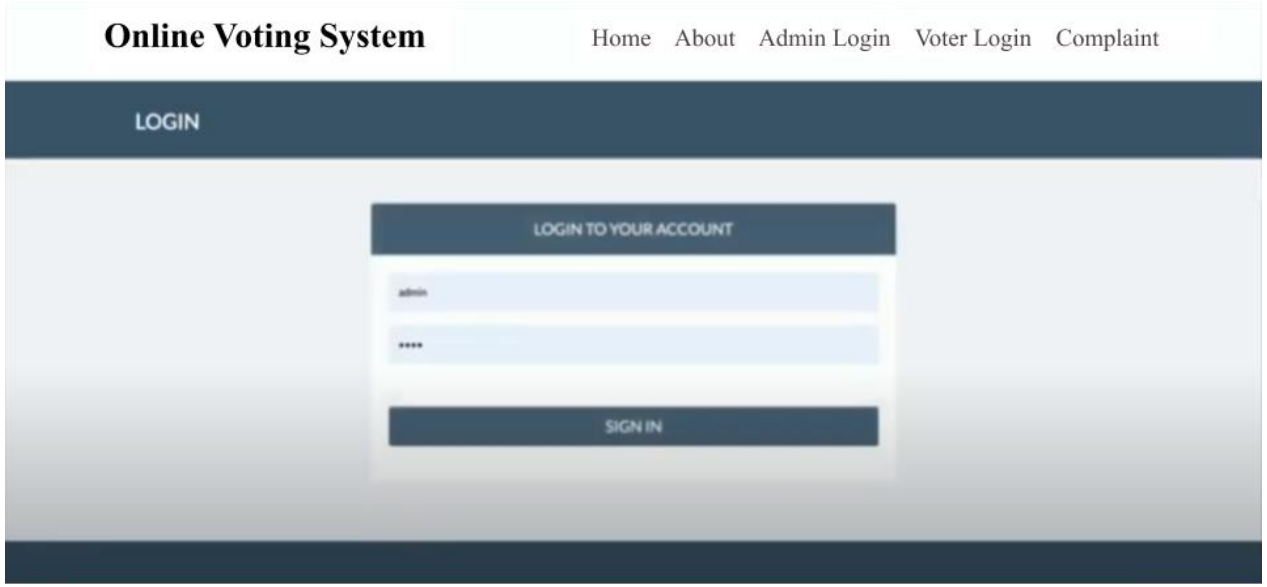
- **Minutiae Based Algorithm**
  - Ridge ending, bifurcation and short ridge or dot.
  - Only small part of finger image is required.
  - Two algorithms used:
    - minutiae-extraction algorithm (fingerprint detection).
    - minutiae-matching algorithm.
- **Image Based Algorithm**
  - Contains size, type and orientation of patterns.

RESULTS

HARDWARE PART



SOFTWARE PART



9

Send

```
Adafruit Fingerprint sensor enrollment
Found fingerprint sensor!
Ready to enroll a fingerprint! Please Type in the ID # you want to save this finger as...
```

1

```
Image taken
Image converted
Remove finger
ID 9
Place same finger again
.............Image taken
Image converted
Creating model for #9
Prints matched!
ID 9
Stored!
Ready to enroll a fingerprint! Please Type in the ID # you want to save this finger as...
```

2

COM7 (Arduino/Genuino Uno)

Send

```
Adafruit finger detect test
Found fingerprint sensor!
Waiting for valid finger...
```

3

COM7 (Arduino/Genuino Uno)

Send

```
Adafruit finger detect test
Found fingerprint sensor!
Waiting for valid finger...
UNLOCKED!Found ID #1 with confidence of 60
```

Autoscroll | Carriage return | 9600 baud | Clear output

File Edit Sketch Tools Help

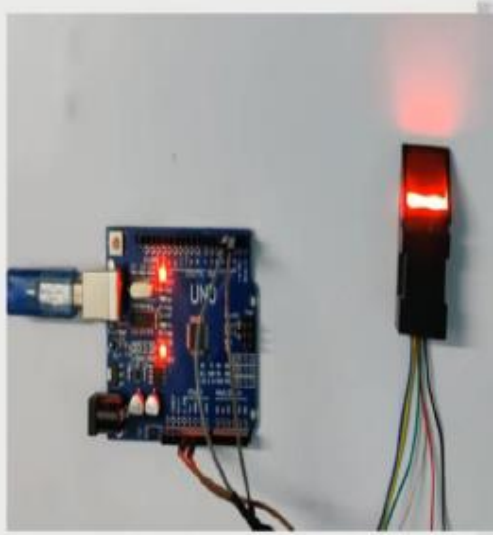
fingerprint

```
/*************************************************
  This is an example sketch for our optical Fingerprint sensor

  Designed specifically to work with the Adafruit BMP085 Breakout
  ----> http://www.adafruit.com/products/751

  These displays use TTL Serial to communicate, 2 pins are required to
  interface
  Adafruit invests time and resources providing this open source code,
  please support Adafruit and open-source hardware by purchasing
  products from Adafruit!

  Written by Limor Fried/Ladyada for Adafruit Industries.
  BSD license, all text above must be included in any redistribution
 *************************************************/
```



```
#include <Adafruit_Fingerprint.h>
```

Done uploading.

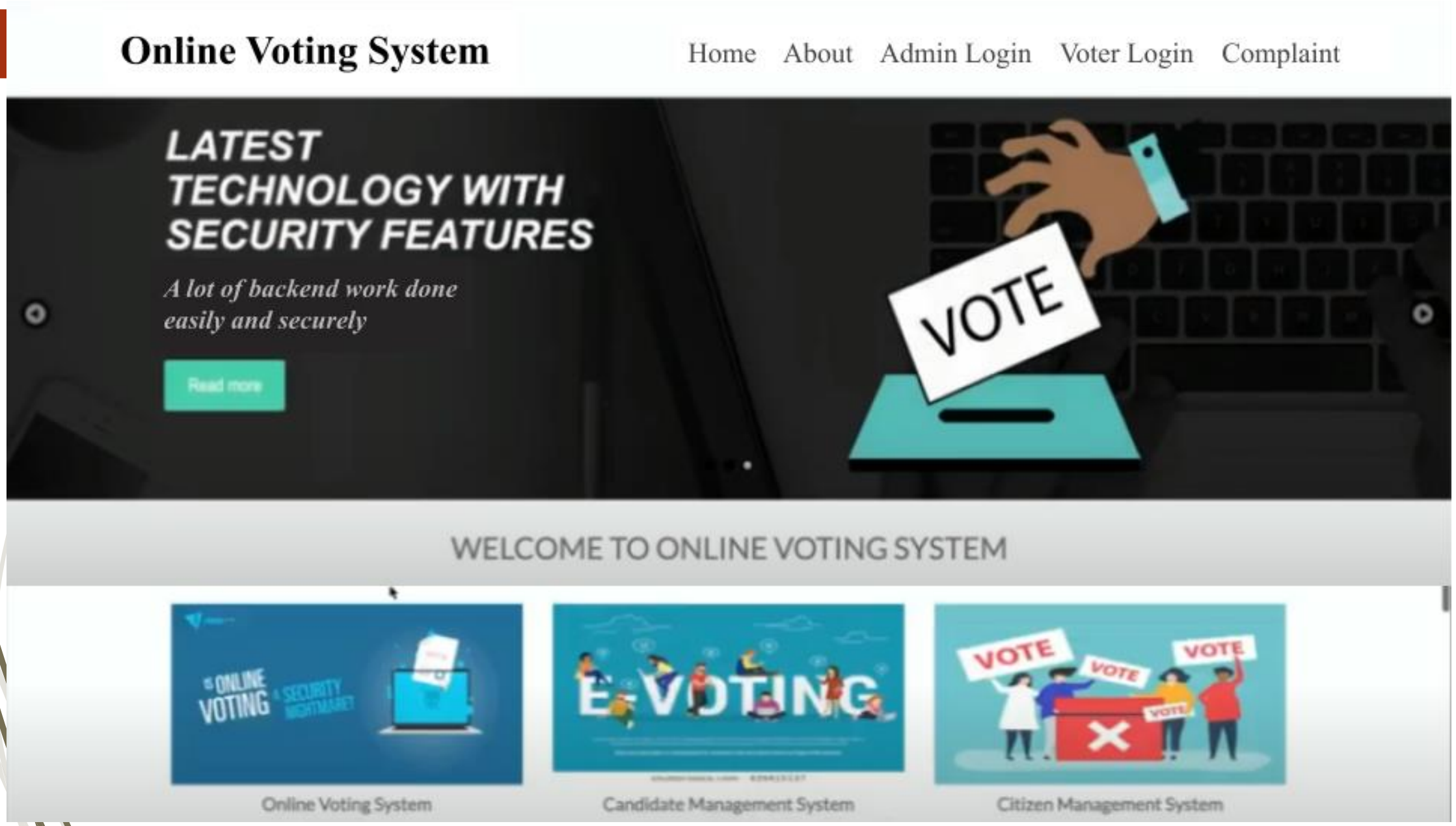
```
Sketch uses 5134 bytes (15%) of program storage space. Maximum is 32256 bytes.
Global variables use 529 bytes (25%) of dynamic memory, leaving 1519 bytes for local variables. Maximum is 2048 bytes.
```

COM3

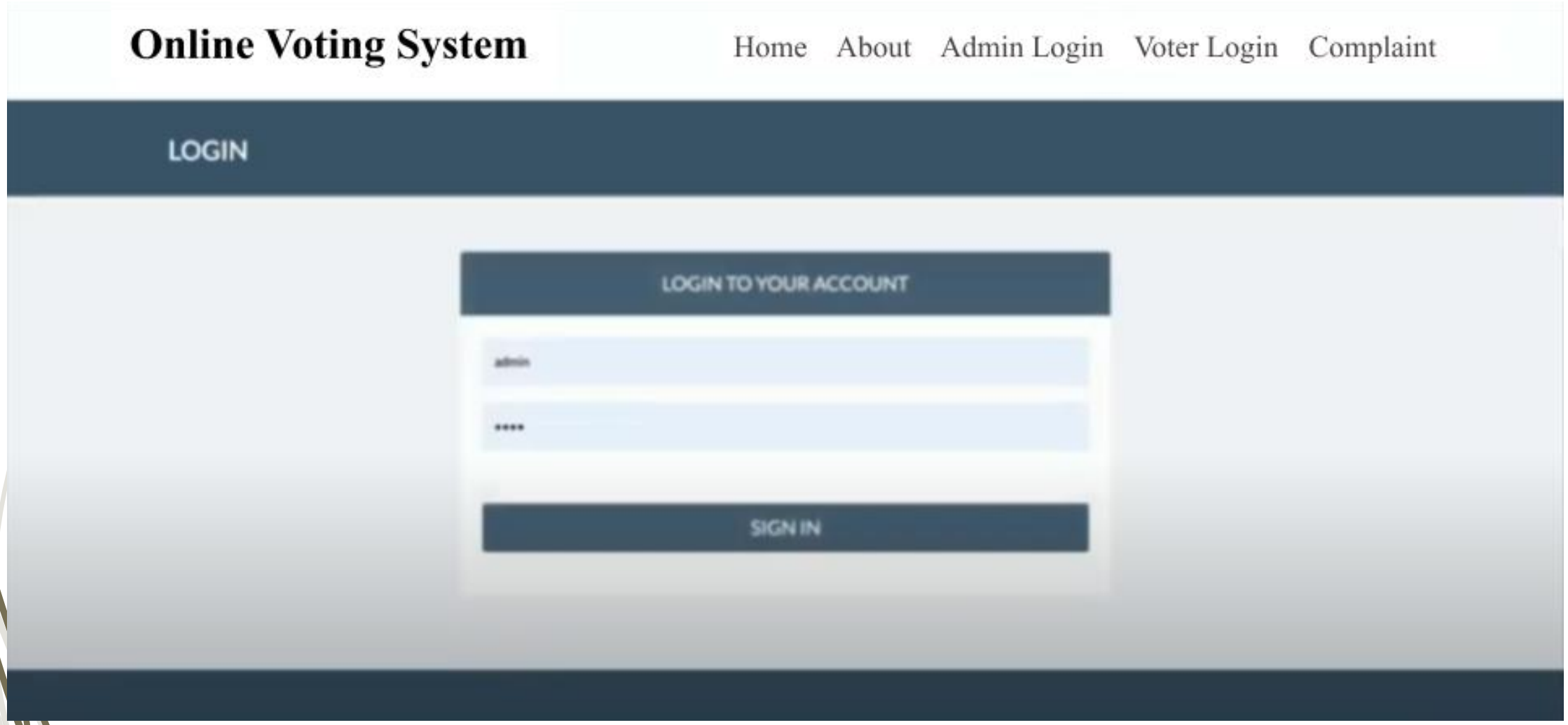
Send

```
Adafruit finger detect test
Found fingerprint sensor!
Sensor contains 2 templates
Waiting for valid finger...
Found ID #1 with confidence of 75
Found ID #1 with confidence of 120
```

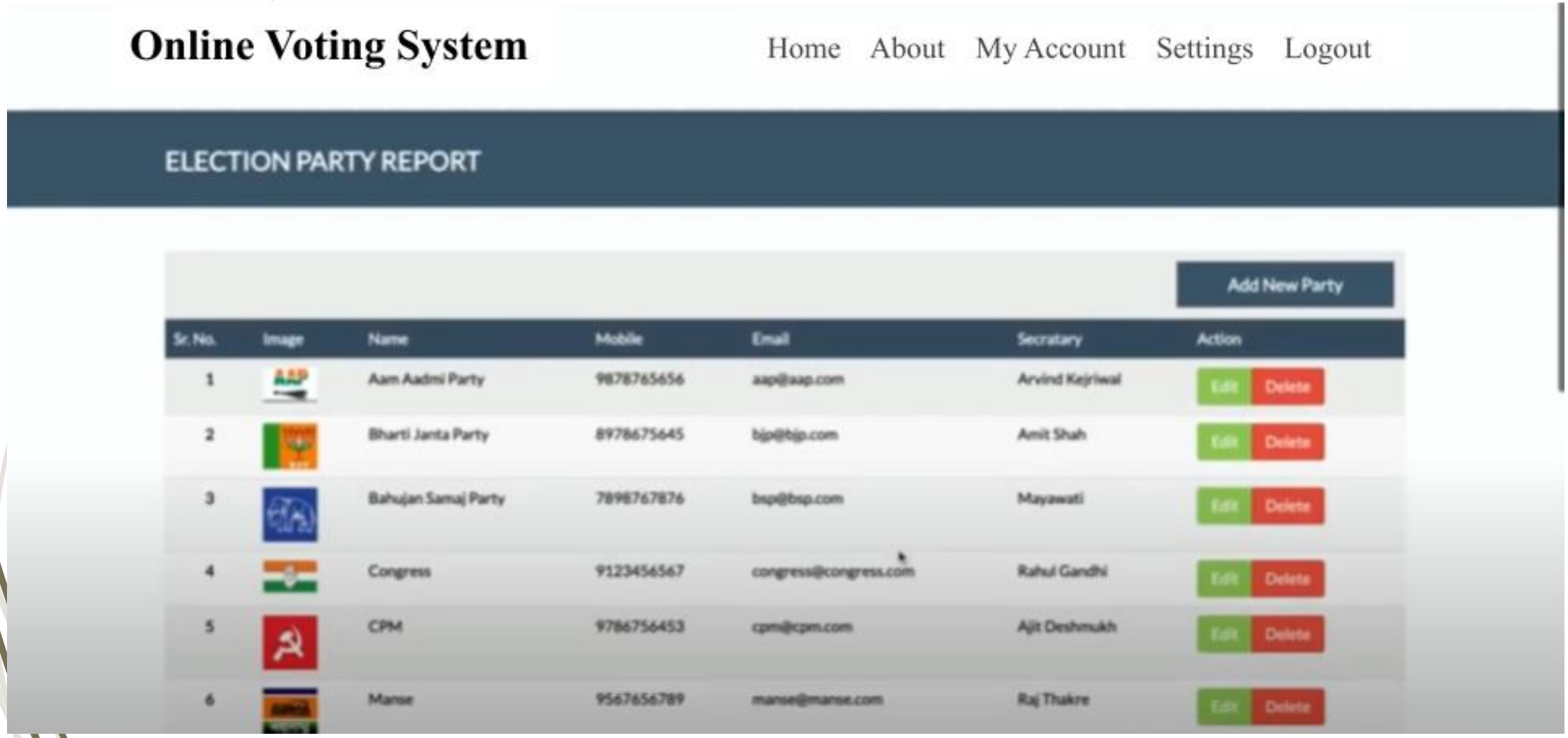
Home Page for Online Voting System



Login Page


Online Voting System
Home
About
Admin Login
Voter Login
Complaint
LOGIN
LOGIN TO YOUR ACCOUNT
admin
••••
SIGN IN

Dashboard for Admins



CONCLUSION AND FUTURE SCOPE

- This project can be used as a voting machine to prevent rigging, during the elections in the polling booths.
- Fast track voting which could be used in small scale elections like resident welfare associations, ‘panchayat’ level election and other society level elections, where results can be instantaneous.
- It could also be used to conduct general assembly elections, on a small-scale basis.

REFERENCES

[1] Brian Evans, Beginning Arduino Programming, Technology in Action.

[2] Santosh Kumar Shaw, Sashank Poddar, “Design and Implementation of Arduino Based Voting Machine”, IEEE Electron Device Kolkata Conference, 24-25 November, 2018, Kolkata, India.

[3] Atharva Jamkar, Omkar Kulkarni, “Biometric Voting Machine Based on Fingerprint Scanner and Arduino”, IOSR-JEEE, 2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT), Manipal University Jaipur, Sep 28-29, 2019.

[4] Rahil Rezwan, Md. Abdur Rahman, “Biometrically secured electronic voting machine”, IEEE R10 Humanitarian Technology Conference, Dhaka, 21-23 Dec, 2017.

THANK YOU!

Any Questions?