RFID & RF Based E-Voting System

A Report submitted in partial fulfillment of the requirements to complete Term Work of Project Based Learning (PBL) in the department of

Electronics & Telecommunication Engineering

As prescribed by

Savitribai Phule Pune University

By

Tejas Bhalchandra Kole	**PRN No: 72314248D**
Harsh Rakesh Kolhe	**PRN No: 72314249B**
Tejas Atul Kotgire	**PRN No: 72314254J**
Shraddha Vijay Kshirsagar	**PRN No: 72314261M**

Under the Guidance of
Ms. Archana Deokate

Sinhgad Institutes

E&TC Engineering Department
Smt. Kashibai Navale College of Engineering
44/1, Vadgaon (Bk), Off Sinhgad Road, Pune - 411041.

Sinhgad Institutes

Smt. Kashibai Navale College of Engineering

Sr. No. 44/1, Vadgaon (Bk), Off Sinhgad Road, Pune – 411 041.

Department of E&TC Engineering

Certificate

This is to certify that, following students,

1.	**Tejas Bhalchandra Kole**	Roll No.:	**E2301**
2.	**Harsh Rakesh Kolhe**	Roll No.:	**E2302**
3.	**Tejas Atul Kotgire**	Roll No.:	**E2304**
4.	**Shraddha Vijay Kshirsagar**	Roll No.:	**E2306**

has completed all the Term Work in the subject **Project Based Learning (PBL)** satisfactorily in the department of E&TC Engineering as prescribed by Savitribai Phule Pune University, in the academic year 2024 – 2025.

Faculty-In-Charge	Mr. V. P. Niwane/ Mrs. T. A. Mate **Coordinator**	Dr. P. S. Raskar **Program Coordinator (HOD)**

Date: 16/04/2025

Acknowledgement

We are deeply grateful to everyone who has supported and contributed to the success of this engineering project. First and foremost, I would like to express my sincere gratitude to my guide, **Ms. Archana Deokate**, for his/her invaluable guidance, encouragement, and expertise throughout the project.

We would also like to extend my heartfelt thanks to the department, particularly head of department, Dr. P. S. Raskar, and PBL coordinators Mrs. T. A. Mate and Mr. V. P. Niwane, for their tireless efforts and dedication. Their collaborative spirit and innovative ideas have greatly enhanced the quality and depth of this project.

In addition, we would like to thank I would also like to extend my heartfelt thanks to our esteemed Principal, Dr. A. V. Deshpande, and Vice Principal, Dr. K. R. Borole and Prof. R. H. Borhade, for their unwavering support and leadership. Their encouragement has been a cornerstone of this project's success.

Lastly, we are truly thankful to my family and friends for their unwavering support and encouragement. We also wish to acknowledge the administrative staff at Smt. Kashibai Navale College of Engineering, Pune who have provided seamless departmental support and resources.

Tejas Bhalchandra Kole
Harsh Rakesh Kolhe
Tejas Atul Kotgire
Shraddha Vijay Kshirsagar

Abstract

• In today's digital era, the need for secure and transparent election processes has become critical due to the limitations of traditional paper-based voting systems, which are prone to human error, duplication, and manipulation.

• This project proposes a smart, IoT-enabled **RFID & RF Based E-Voting System** using the **ESP32 microcontroller**, designed specifically for **small-scale elections** in institutions and organizations.

• The system uses **RFID technology (RC522)** to uniquely identify each voter through a card or tag, ensuring only authorized users can vote, thereby preventing fake or duplicate entries.

• Upon successful card scan, the voter is guided through a **10-second timed voting window** where they can cast their vote using **one of four physical buttons**, each mapped to a political party: **BJP, INC, AAP, and NOTA**.

• Feedback is provided through **LED indicators**, a **buzzer**, and a **16x2 LCD display** which shows dynamic messages such as voter greeting, countdown timer, confirmation of vote, and system status.

• The **ESP32**, leveraging its built-in **WiFi capability**, logs each vote in **real-time to a Google Sheet** using HTTP requests, including the voter's name, UID, selected party, and status (Casted, Invalid, Already Voted).

• A special **master/admin RFID card** is implemented to conclude the voting session, which displays the **result summary and winner** on the LCD and sends a formatted summary to the Google Sheet — handling conditions like **tie** or **no votes cast**.

• The system incorporates edge-case handling for **invalid cards**, **duplicate scans**, and **voting timeouts**, making it **fail-safe and robust**.

• This project serves as a **proof-of-concept** for a smart, eco-friendly, and tamper-proof voting system — enhancing reliability, usability, and transparency in organizational elections while showcasing practical applications of **RFID**, **WiFi IoT**, and **embedded systems**.

🔑 **Keywords – List of Keywords**

- RFID
- RC522
- ESP32
- E-Voting System
- Google Sheets Integration
- Master Card
- Real-Time Data Logging
- 16x2 LCD Display
- IoT
- WiFi Connectivity
- Buzzer
- Voting Buttons
- LED Indicators
- UID Authentication
- Voting Timeout
- Arduino IDE
- Cloud-based Voting
- Secure Vote Casting
- Embedded Systems
- Project Based Learning (PBL)

Table of Contents

Index

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1
Introduction

1.1 Background

• In today's fast-paced digital era, the traditional voting system is increasingly seen as outdated due to its reliance on manual processes such as paper ballots, physical verification, and hand counting.

• These traditional methods are **time-consuming, prone to human error, and vulnerable to fraud** such as multiple voting or identity manipulation.

• To address these challenges, **technological solutions using RFID and IoT** have emerged as promising alternatives for secure, fast, and reliable voting systems.

• This project presents a **smart RFID & RF Based E-Voting System** using **ESP32**, which provides **real-time vote logging** and **transparent result monitoring** via **Google Sheets** using WiFi connectivity.

1.1.1 Problem Statement

• The major problems in conventional voting systems include:

- Manual registration and validation.
- Risk of **duplicate or fake votes**.
- Delays in vote counting and result declaration.
- Lack of transparency and security.

  • There is a need for a system that enables **secure authentication**, **one-person-one-vote logic**, **tamper-proof logging**, and **quick result access** — especially in small-scale elections like colleges, clubs, or organizations.

1.1.2 Significance of the Project

• This project demonstrates how **RFID and embedded systems** can be used to automate and secure the voting process, eliminating manual errors.

• It ensures **only authorized users can vote once**, and all data is stored on the **cloud (Google Sheets)** in real time, providing **transparency and traceability**.

• The system also introduces features such as **admin summary view**, **10-second voting timeout**, and **user feedback** via **LCD, buzzer, and LEDs** — simulating a **real-world EVM** setup on a

compact prototype scale.

1.2 Project Objectives

• The primary objectives of this project are:

- To design a secure and portable RFID-based voting prototype.
- To ensure real-time vote logging using **WiFi + Google Sheets integration**.
- To implement a **Master Card feature** to display and send final results.
- To eliminate the chances of **duplicate voting or invalid card entries**.
- To improve usability with **LCD guidance**, **LED/buzzer feedback**, and **vote timeout logic**.

1.3 Organization of Report

• This report is organized into the following chapters:

- **Chapter 1** provides the background, problem statement, and objectives of the project.
- **Chapter 2** presents a literature review of existing RFID-based and IoT-enabled voting systems.
- **Chapter 3** explains the system design, hardware/software architecture, and implementation details.
- **Chapter 4** shows the results with circuit diagrams, output snapshots, and discussion.
- **Chapter 5** concludes the project with a summary of findings, limitations, and future work.

Chapter 2

Literature Survey

Chapter 2

Literature Survey

2.1 Related Work

• Various smart voting systems have been proposed in recent years to replace manual ballot-based elections.

• Most of these projects focus on RFID authentication, real-time data logging, and secure vote counting.

• Below is a summary of some significant prior works:

2.1.1 Existing Solutions and Technologies

S.N.	Year	Title	Methodology
1	2017	RFID-Based Electronic Voting Machine	Used RFID cards to authenticate voters and restrict duplicate voting.
2	2019	IoT-Enabled E-Voting with ESP32 and Cloud	ESP32 used to log vote data to Google Sheets for transparency.
3	2020	Multi-Level Authentication Using RFID & Biometrics	Combined RFID with fingerprint scanning for higher security.
4	2021	Blockchain-Based Result Storage	Used RFID with blockchain to store vote logs in a tamper-proof ledger.
5	2023	IoT Voting with LCD Feedback and Remote Monitoring	Provided LCD confirmation and stored data in real-time Google Sheets via WiFi.

2.2 Theoretical Foundations

This section covers the core technologies and principles used in the design of our system.

2.2.1 Core Concepts

• RFID (Radio Frequency Identification)

- A wireless system that uses tags and readers to identify objects via unique UIDs.
- Enables secure and contactless identification in voting systems.

• **ESP32 Microcontroller**

- A powerful dual-core WiFi-enabled chip used for edge computing and IoT solutions.
- Handles all voting logic, data transmission, LCD, RFID, and I/O operations.

• **Google Apps Script**

- A scripting language used to connect ESP32 with Google Sheets via HTTP requests.
- Allows real-time logging of vote data and results on the cloud.

2.2.2 Underlying Principles

• **One-Person-One-Vote Logic**

- Ensures each card (UID) can only be used once, avoiding multiple voting.

• **Event-Driven Embedded Programming**

- Code execution based on actions (like card scan or button press) using interrupts and conditions.

• **Cloud-Based Logging**

- Votes are sent via URLs (HTTP GET) to a Google Sheet, making data transparent and easily accessible.

• **User Feedback Loop**

- LCD shows instructions and messages, while LEDs and buzzer confirm actions or errors.

Chapter 3

System Design

Chapter 3

System Design

3.1 Design and Development

3.1.1 System Architecture

• The proposed RFID-based voting system is built on a **modular architecture** combining hardware and software elements to ensure secure vote casting and transparent result logging.

• The **ESP32** acts as the brain of the system, handling:

- RFID tag scanning
- Button-based voting input
- Real-time Google Sheet communication
- LCD display for instructions
- LED and buzzer feedback

• The entire process is **event-driven** — activated by card scans and button presses.

✅ **Block Diagram Overview**

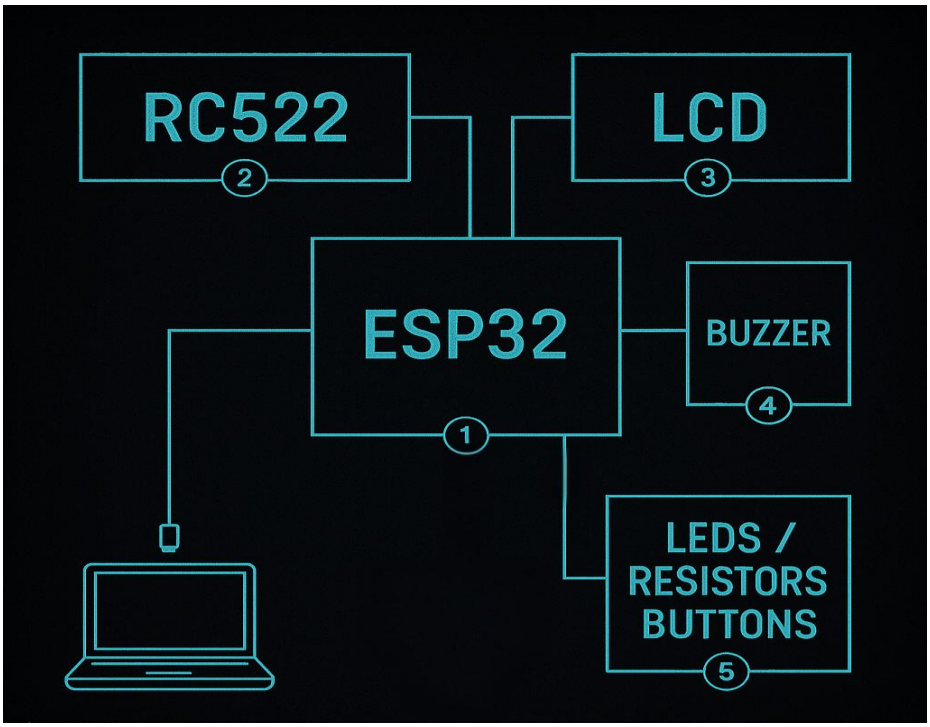


Figure 1.1 – Block Diagram of RFID & RF Based E-Voting System

🧱 **Main blocks**:

- ESP32 Microcontroller
- RC522 RFID Reader
- Push Buttons (BJP, INC, AAP, NOTA)
- 16x2 LCD with I2C
- 4 LEDs + Buzzer
- Google Sheet Integration (via Apps Script)

3.1.2 Hardware Design

• The hardware is built on a **breadboard prototype** powered by USB. All components are interfaced with the ESP32 using jumper wires.

💡 **Key Modules Connected:**

- RFID Reader → SPI Communication
- LCD → I2C Communication
- Buttons & LEDs → GPIO Inputs/Outputs
- Buzzer → GPIO Output
- Power Source → USB (5V)

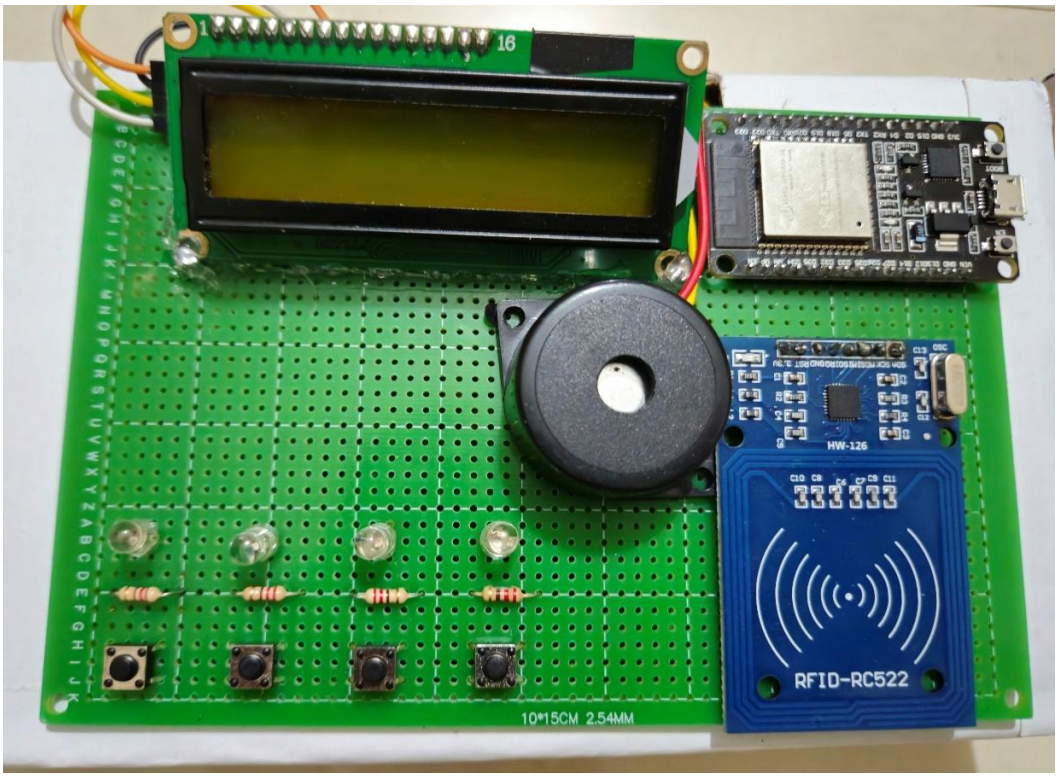


Figure 1.2 – Actual Hardware Setup of the Voting System

🧰 **Technology Stack Used**

Layer	Technology / Component	Purpose
Microcontroller	ESP32 Wroom-32	Core controller handling all input/output and WiFi
RFID Module	RC522	Scans and identifies RFID cards
Display Interface	16x2 LCD with I2C	Shows user instructions, status, and results
User Input	Push Buttons (x4)	Used for party-wise vote selection (BJP, INC, etc.)
Feedback Output	LEDs + Buzzer	Indicates vote status (casted, invalid, etc.)
Software Platform	Arduino IDE	Code development and uploading to ESP32
Data Logging	Google Sheets + Apps Script	Stores real-time votes and results in the cloud
Communication	HTTP GET Requests via WiFi	Sends data from ESP32 to Google Sheet
Programming Language	C / C++	For Arduino coding logic

3.1.3 Software Development

• The system is programmed using the **Arduino IDE** with libraries:

- WiFi.h, Wire.h, SPI.h, MFRC522.h, HTTPClient.h, LiquidCrystal_I2C.h

• The logic includes:

- **RFID card UID checking**
- **Vote tracking using arrays**
- **10-second voting countdown**
- **Google Sheet logging using URL encoding**
- **Admin UID triggers vote summary and winner display**

3.2 Implementation

3.2.1 Experimental Setup

• The system was assembled using:

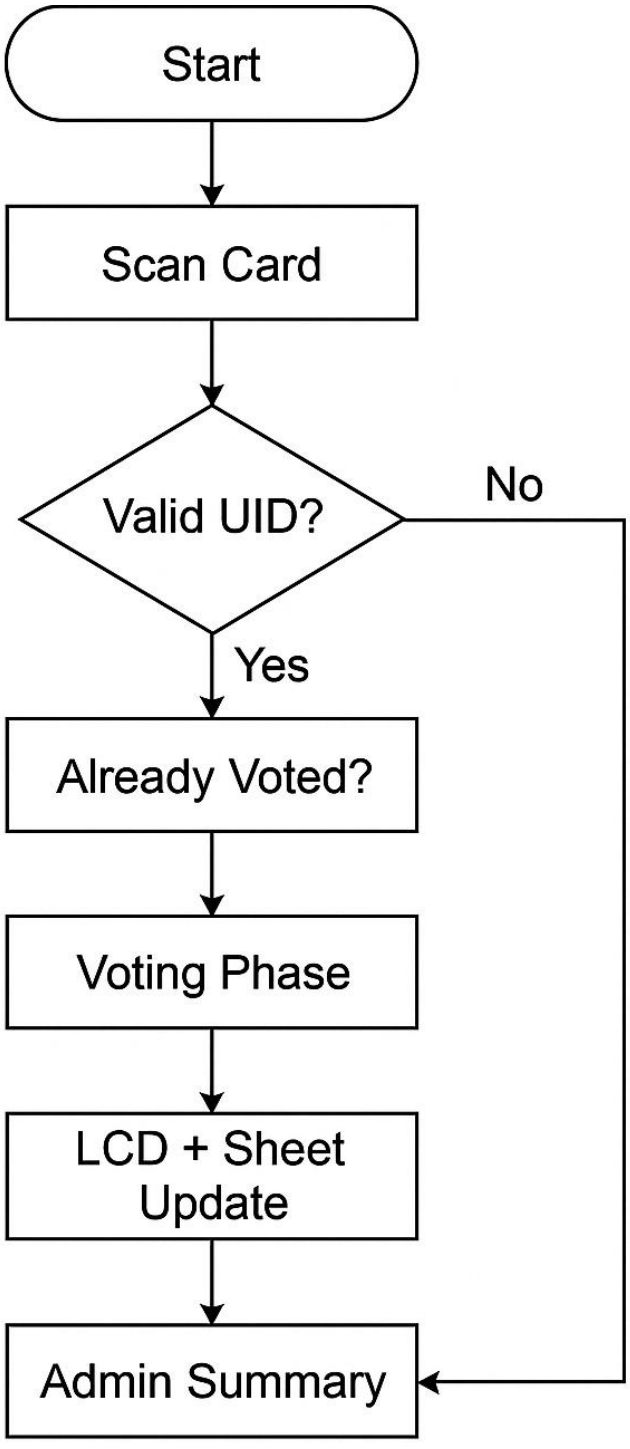
- **5 registered RFID cards** (each linked to a voter name)
- **1 master RFID card** (for admin)
- **Push buttons** to simulate party selection
- **Google Sheet** and **Apps Script** for backend

✅ All features were tested with real hardware and logs were verified on the sheet.

- **The GPIO pin connections of the ESP32 to external components are listed in Table 3.2 below:**

Component	Pin Description	ESP32 GPIO Pin
RC522 (RFID)	SDA (SS)	GPIO 5
	SCK	GPIO 18
	MOSI	GPIO 23
	MISO	GPIO 19
	RST	GPIO 27
	VCC	3.3V
	GND	GND
LCD (I2C 16x2)	SDA	GPIO 21
	SCL	GPIO 22
	VCC	3.3V
	GND	GND
Buttons	BJP	GPIO 33
	INC	GPIO 32
	AAP	GPIO 25
	NOTA	GPIO 4
LEDs (All Green)	BJP Indicator	GPIO 12
	INC Indicator	GPIO 13
	AAP Indicator	GPIO 14
	NOTA Indicator	GPIO 15
Buzzer	Output	GPIO 26

3.2.2 Flow Chart



Chapter 4

Results and Discussion

Chapter 4

Results and Discussion

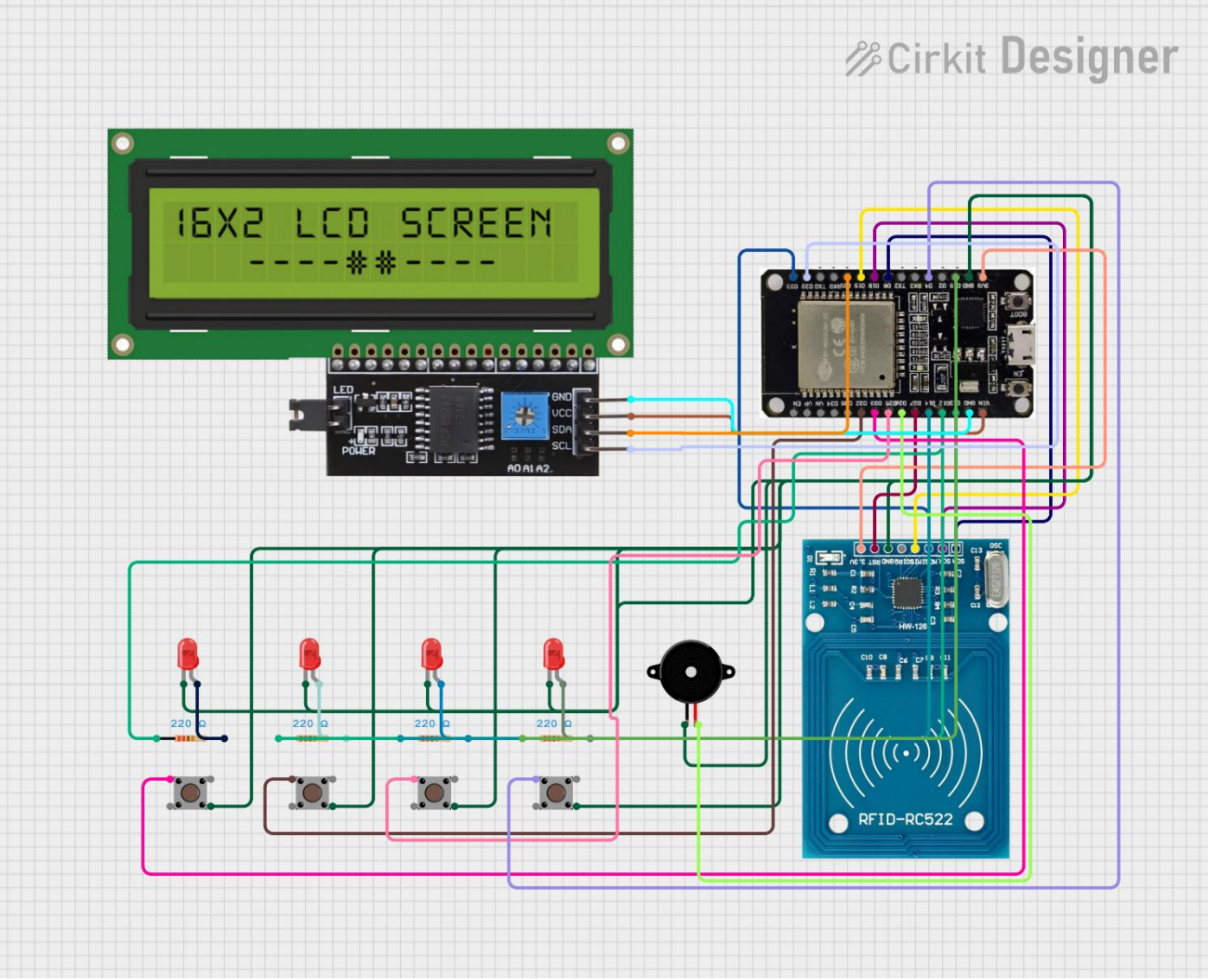
4.1 Results

4.1.1 Circuit Diagram

• The circuit diagram represents all physical connections between the ESP32, RFID reader, push buttons, LEDs, buzzer, and LCD display.

• It shows the **SPI connections for the RC522 RFID module**, **I2C interface for the LCD**, and GPIO pins assigned to each party button and corresponding LED.

Figure 4.1 – Circuit Diagram of the E-Voting System



4.1.2 Circuit Working Description

• The heart of the system is the **ESP32 microcontroller**, which acts as the central controller managing inputs, outputs, and cloud communication.

• The **RC522 RFID Reader** is connected to the ESP32 using the **SPI interface** (GPIOs: 5, 18, 23, 19, 27). When a card is brought near the reader, it scans the **UID** and sends it to the ESP32 for authentication.

• If the card is valid and hasn't been used before, the LCD displays a **"Welcome"** message followed by a **10-second countdown** for voting.

• The user can then press one of the **four push buttons**, each representing a political party:

- **BJP → Button on GPIO 33**
- **INC → GPIO 32**
- **AAP → GPIO 25**
- **NOTA → GPIO 4**

• Once a vote is cast, the corresponding **LED (Red, Green, Blue, Yellow)** glows briefly using GPIOs 12 to 15, and a short beep is played via the **buzzer** on GPIO 26 to indicate successful voting.

• Simultaneously, the selected party, voter's name, UID, and status are logged in **real-time to a Google Sheet** via an HTTP request using ESP32's **WiFi capability**.

• If the card has **already been used**, or is **invalid**, the system responds with a proper **error message on LCD** and a **double beep + blink** for feedback.

• Scanning the **Master Card (UID: 33EDA6D)** triggers the **vote summary display**, both on the LCD and on the Google Sheet, including party-wise counts and the declared winner. Tie/no-vote cases are handled smartly in both display and logging.

4.2 Discussion

• The system was successfully tested using **5 unique RFID cards** and **1 Master Card**, each assigned to a known voter or function.

• On scanning a valid card, the user was prompted to vote within a **10-second countdown window** using physical push buttons.

• The vote casting process worked flawlessly, with **LED feedback**, **buzzer alert**, and **Google Sheet logging** visible in real-time.

• Special cases were also handled correctly:

- **Already voted cards** triggered an LCD warning and double-beep.
- **Invalid/unregistered cards** were rejected gracefully with a message and blink.
- **Master card** displayed the party-wise summary and winner/tie/no-vote messages on both LCD and Sheet.

4.2.1 Interpretation of Results

• The system successfully met all design goals — including **accuracy**, **user guidance**, and **real-time logging**.

• Vote data appeared correctly in the Google Sheet with:

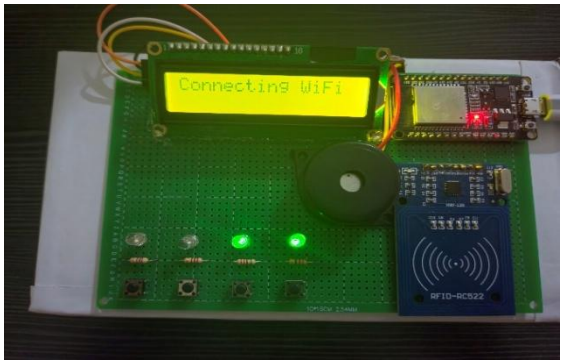
- Voter name
- UID
- Selected party with emoji
- Voting status (✅ Casted, ❌ Already voted, ❌ Invalid)

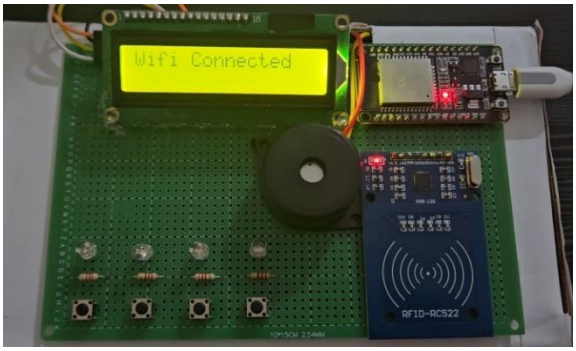
• The **summary row**, generated using the Master Card, included a **multiline party-wise vote count** and the declared **winner** — proving the end-to-end functionality.

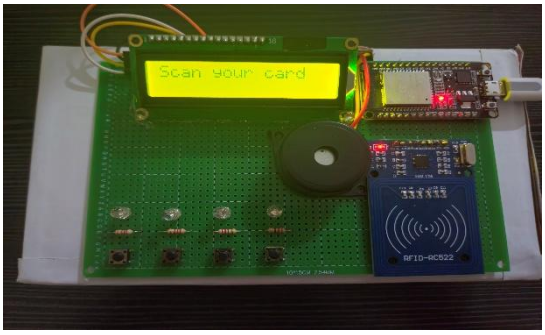
• The system's **user interaction** (LCD messages, timeout, feedback) made the prototype **engaging, error-resistant, and ideal for small-scale voting scenarios**.

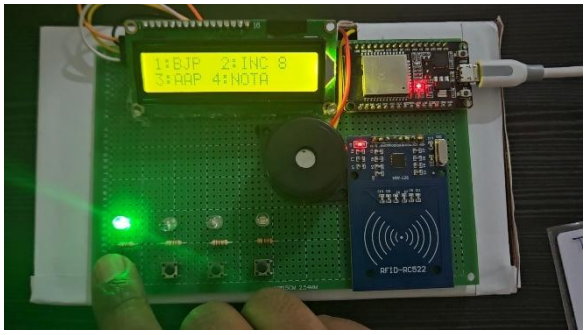
4.3 Working Images of Project Execution

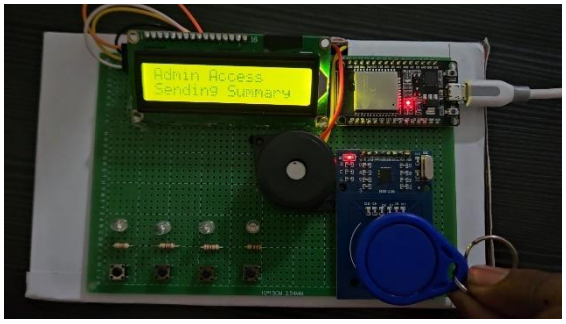
> The following images showcase the live working of the RFID & RF Based E-Voting System during testing and demonstration :

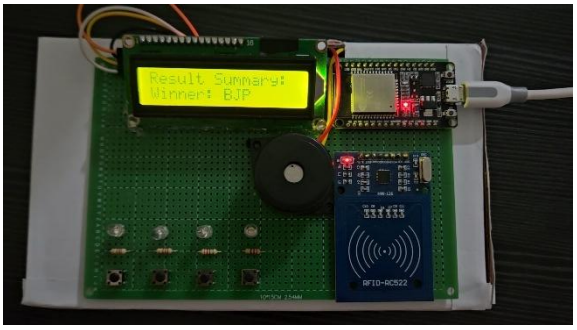


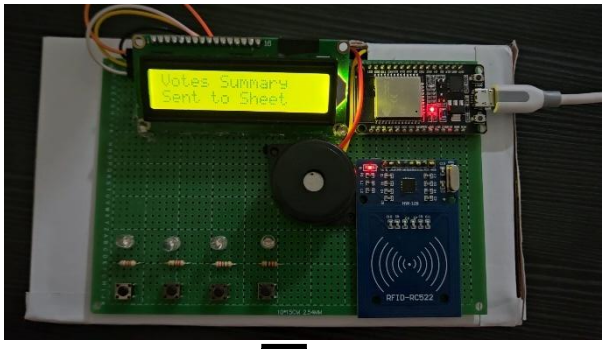






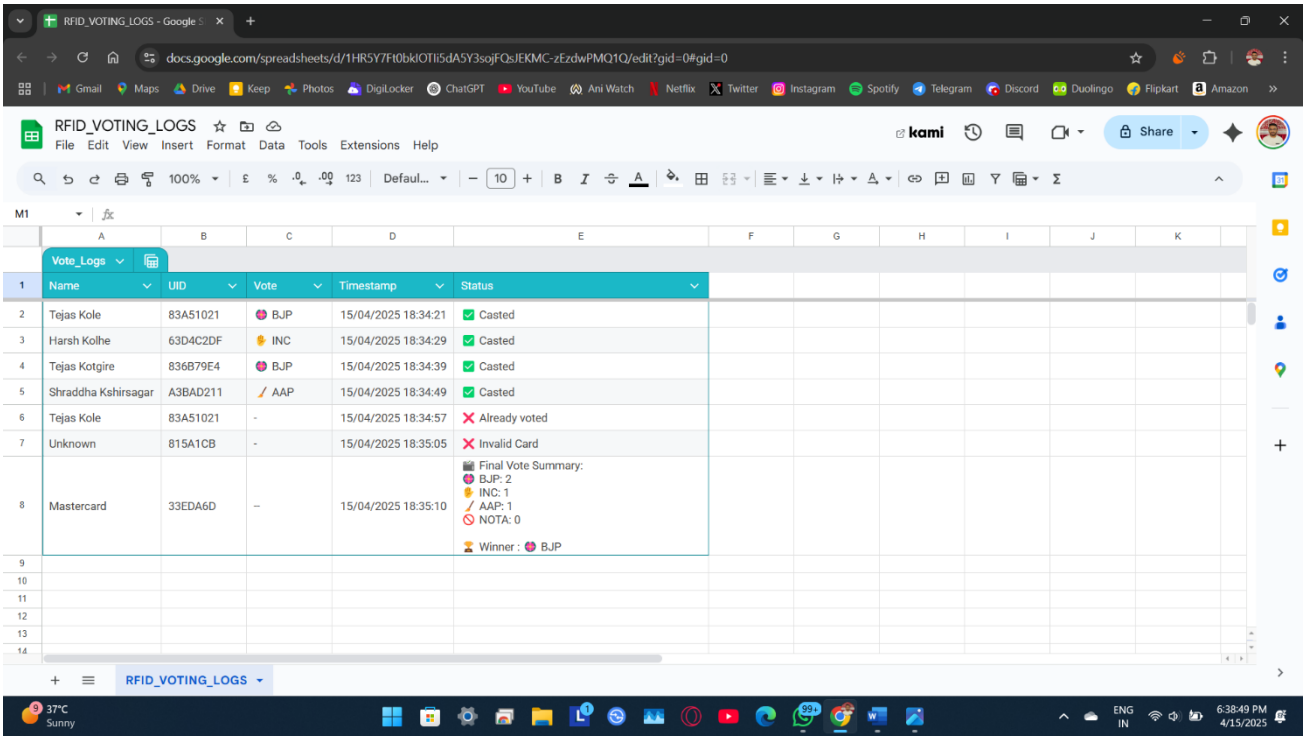






Real-Time Vote Logging on Google Sheet via ESP32

- The below sheet displays real-time logs of RFID-based votes, including voter name, UID, selected party (with emoji), and voting status sent through WiFi :



• The last row in the sheet represents the admin summary triggered by the Master Card, showing party-wise vote counts and the final winner.

Chapter 5

Conclusion

Chapter 5
Conclusion

5.1 Summary of Findings

• The RFID & RF Based E-Voting System achieved its goal of providing a **secure, real-time, and tamper-proof voting solution** using RFID authentication and cloud-based logging.

• All core functionalities were successfully implemented:

- Voter authentication through RFID card scanning.
- One-vote-per-person logic to prevent duplication.
- Vote selection using physical push buttons.
- Real-time data logging to a **Google Sheet** with status, UID, and vote info.
- Display of **live feedback** through LCD, LEDs, and buzzer.
- Admin/master card functionality to display **vote summary and winner**.

• The system also included intelligent features like:

- 10-second timeout with reverse countdown.
- Error handling for **invalid** or **duplicate cards**.
- Emoji-based party representation in both **LCD** and **Google Sheets**.
- Proper formatting of results including **tie cases and no-vote conditions**.

5.2 Limitations and Future Work

• **Limitations:**

- The system supports only **5 RFID cards** due to static array allocation.
- No encryption is used in the communication between ESP32 and Google Sheets.
- Power supply is dependent on USB; not battery-optimized for portability.

• **Future Enhancements:**

- Integration of **biometric verification** along with RFID for dual-layer security.
- Development of a **dashboard or app** to visualize live votes and results.
- Use of **secured HTTPS with encryption** for transmitting sensitive data.
- Expansion to support **multiple voting booths** through a centralized database.

5.3 Conclusion

• This project successfully demonstrates the use of **IoT and embedded systems** to create a **modern electronic voting machine** prototype.

• It eliminates common issues in traditional systems such as **manual errors, fake votes, and delayed results**, and offers a **transparent and user-friendly** voting experience.

• With features like **Google Sheet integration**, **RFID-based validation**, and **admin-controlled result generation**, the system is ideal for **internal elections in educational institutes, clubs, or organizations**.

• Overall, this project serves as a **reliable, efficient, and secure e-voting prototype**, showcasing the practical applications of **RFID technology, microcontrollers, and cloud integration** in real-world systems.

References

References

1. **Rajeshwari Sundararajan, *Microcontroller-Based Electronic Voting Machine*, International Journal of Engineering Science Invention (IJESI), Vol. 2, Issue 2, 2013.**
2. **Klaus Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*, Wiley.**
3. **D. L. Chaudhari, *Smart Electronic Voting Machine Using RFID*, International Research Journal of Engineering and Technology (IRJET), Vol. 5, Issue 4, 2018.**
4. **Espressif Systems, *ESP32 Technical Reference Manual*, Version 4.0.**
5. **NXP Semiconductors, *MFRC522 RFID Module Datasheet*.**
6. **Arduino.cc, *LiquidCrystal_I2C Library Documentation*. https://www.arduino.cc/en/Reference/LiquidCrystal**
7. **Google Developers, *Apps Script Documentation*. https://developers.google.com/apps-script**
8. **William Stallings, *Wireless Communications and Networks*, Pearson Education.**
9. **Arduino.cc Community Forums – Discussions on RFID integration and real-time cloud logging.**
10. **GitHub Repository: *RFID-RF-Based-E-Voting-System*, TejasK-02. https://github.com/TejasK-02/RFID-RF-Based-E-Voting-System**
11. **YouTube Project Demo: *RFID Voting System – Real-Time Demo*. https://youtube.com/shorts/w2tXMks05K4**
12. **Project Resource Drive Folder (Code, Images, Working Proof): https://drive.google.com/drive/folders/1dlD695RuSaaR76-CfT7QR9z3M7TtpGgr**

Appendix A

Cost Sheet

Cost Sheet

• **Table A.1 :**

S.N.	Component	Specification / Model	Quantity	Unit Price (₹)	Total (₹)
1	ESP32 Dev Board	Wroom-32	1	380	380
2	RFID Module	RC522	1	80	80
3	RFID Cards/Tags	UID-based	6 (5+1 admin)	10	60
4	I2C LCD Display	16x2 with I2C backpack	1	190	190
5	Push Buttons	Tactile buttons	4	5	20
6	LEDs	Green	4	2	8
7	Buzzer	5V	1	18	18
8	Resistors	220Ω	4	5	20
9	Jumper Wires	Male-to-male	50	2	100
10	PCB	General purpose	1	120	120
11	Soldering Wire	Standard 22 SWG (Rosin Core)	1 Roll	100	100
12	Soldering Paste (Flux Wax)	Standard flux (small box)	1	40	40
13	Glue Stick	Medium size (for glue gun)	1	10	10
14	Electricity	Power consumption (approx.)	-	10	10

| | Total Estimated Cost :: ₹1156 | |

Appendix B

Code

Code 1: Arduino Code

The following code is used to implement RFID authentication, voting logic, LCD feedback, and Google Sheet integration using ESP32 :

```
#include <WiFi.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SPI.h>
#include <MFRC522.h>
#include <HTTPClient.h>

#define RST_PIN 27
#define SS_PIN 5
#define BUZZER 26

int leds[] = {12, 13, 14, 15};
int buttons[] = {33, 32, 25, 4};
String parties[] = {"BJP", "INC", "AAP", "NOTA"};
int votes[4] = {0, 0, 0, 0};

String masterUID = "33EDA6D";
LiquidCrystal_I2C lcd(0x27, 16, 2);
MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
  Serial.begin(115200);
  SPI.begin();
  mfrc522.PCD_Init();
  lcd.begin();
  lcd.backlight();
  pinMode(BUZZER, OUTPUT);
  for (int i = 0; i < 4; i++) {
    pinMode(leds[i], OUTPUT);
    pinMode(buttons[i], INPUT_PULLUP);
  }
}
```

..

…

….

Note: Complete 425 lines of code is available in soft copy and on the GitHub repository.

https://github.com/TejasK-02/RFID-RF-Based-E-Voting-System

Code 2: Serial Interface

The following is a sample output observed in the Serial Monitor during real-time operation of the RFID & RF Based E-Voting System :

Card UID: 83A51021

Please vote using buttons (BJP/INC/AAP/NOTA)...

Vote Casted for Candidate BJP

Sending to Google Sheet:

https://script.google.com/macros/s/AKfycbxp1hC_1_Kb9F1buncDfLuXfGl2yaUmO_cj3gbC80ti2uvO_sBUgnSSMxHilJh-n2VD/exec?name=Tejas%20Kole&uid=83A51021&vote=%F0%9F%AA%B7%20BJP&status=%E2%9C%85%20Casted

Response: <HTML>

<HEAD>

<TITLE>Moved Temporarily</TITLE>

</HEAD>

<BODY BGCOLOR="#FFFFFF" TEXT="#000000">

<!-- GSE Default Error -->

<H1>Moved Temporarily</H1>

The document has moved <A HREF="https://script.googleusercontent.com/macros/echo?user_content_key=AehSKLgmiLQKfOWBP1AAZbpE9gFLMEzy5G9mbTQ3buBmO1f_0z7a6hh8hhuWpTjNnI-Bsik_qjZCqH2CJVHCFXCVV02OA2YOE8tOchiJmR9Lnv1SCqey-zxBDOduU8MJtZmsW4KKHFneskltAEMRZvOAqW0FjYFhJU7ZdTF0mlcZebSog_uthrxh_nJ5r9NaJqqLjZOImUEb56vd2waCXWQ617rB1cGxUaCF7cG6yzuncIQpty0Jz6mSX_mXIWty9Va2b3VOXP6rI-BtbM4Fe8kW5t_bM8D9HO2S1UFcCskF2vOt1eZiFSDz6u7DD7ovP_TiWhz7sl0RS5Y0rtrGngE9ylThaAJqmhN7OomtUV6GBSOBkE1Gy4A3rxBYAgeUm6h9zpbImphj&lib=ME9_pg7NCfV4qphC__D4MUMWr1zeZllwf">here</A>.

</BODY>

</HTML>

Card UID: 33EDA6D

Sending Summary to Google Sheet...

https://script.google.com/macros/s/AKfycbxp1hC_1_Kb9F1buncDfLuXfGl2yaUmO_cj3gbC80ti2uvO_sBUgnSSMxHilJh-n2VD/exec?name=Mastercard&uid=33EDA6D&vote=--&status=%F0%9F%97%B3%EF%B8%8F%20Final%20Vote%20Summary%3A%0A%F0%9F%AA%B7%20BJP%3A%201%0A%E2%9C%8B%20INC%3A%200%0A%F0%9F%A7%B9%20AAP%3A%200%0A%F0%9F%9A%AB%20NOTA%3A%200%0A%0A%F0%9F%8F%86%20Winner%20%3A%20%F0%9F%AA%B7%20BJP

Summary sent to sheet.

Appendix C

Data Sheet

Data Sheet for Arduino (ESP32 Dev Board)

- **The ESP32 Wroom-32 is a dual-core microcontroller with built-in WiFi and Bluetooth, used as the central controller in the project.**

Key specifications are listed below:

Parameter	Value
Microcontroller	Tensilica Xtensa LX6 (dual core)
Clock Speed	Up to 240 MHz
Flash Memory	4MB
SRAM	520 KB
WiFi & Bluetooth	Built-in (802.11 b/g/n)
Operating Voltage	3.3V
GPIO Pins	34

- **Official Datasheet**:

https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

Data Sheet for LED (Green 5mm)

- **LEDs are used for visual feedback in the voting process. Green 5mm LEDs are used for all four party indicators :**

Parameter	Value
Color	Green
Diameter	5mm
Forward Voltage	2.0V – 2.2V
Forward Current	20mA
Viewing Angle	60°
Package Type	Round, Diffused

- **Typical Datasheet**: https://components101.com/sites/default/files/component_datasheet/LED-Green.pdf

Data Sheet for Buzzer (5V Passive Type)

- **A passive buzzer is used to indicate system actions such as vote cast, invalid attempt, or admin access :**

Parameter	Value
Type	Passive
Rated Voltage	5V DC
Operating Voltage	3V – 7V
Sound Output	85 dB (at 10cm)
Resonant Frequency	2.0 ± 0.5 KHz
Pin Type	PCB Mount

- **Buzzer Datasheet :** https://components101.com/sites/default/files/component_datasheet/Buzzer-Datasheet.pdf

Datasheets for Other Components

- **RC522 RFID Reader Module**
  - Manufacturer: NXP Semiconductors
  - Datasheet: https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf

---

- **LCD Display (16x2 with I2C)**
  - Manufacturer: Hitachi-compatible
  - Datasheet: https://components101.com/sites/default/files/component_datasheet/LCD16x2.pdf

---

- **Tactile Push Buttons**
  - Standard 6mm button used in breadboard circuits
  - Datasheet: https://components101.com/sites/default/files/component_datasheet/Tactile-Switch-Datasheet.pdf

---

- **220Ω Resistor**
  - Standard carbon film resistor
  - Datasheet: https://components101.com/sites/default/files/component_datasheet/Resistor-Datasheet.pdf

---

Appendix D

Technology Readiness Levels

Technology Readiness Levels Summary

• Table D.1 :

TRL	Description	Criterion	Evidence	Assessment (Y/N)	Supporting Evidence
1	**Basic Principles Observed**	Have the underlying principles of RFID, ESP32, microcontrollers, sensors, etc. been understood?	Literature review on ESP32, RFID, WiFi, buzzer, LCD	Yes	Literature Review Report, Datasheets, Project Proposal
2	**Technology Concept Formulated**	Has a conceptual design for the project been developed?	Block diagram, flowchart, list of components, working strategy	Yes	System Design Diagram, Flowchart, Project Sketches
3	**Critical Function and/or Physical Principle Validated**	Have basic modules been tested?	Testing of RFID card detection, button input, buzzer, and LCD	Yes	Breadboard Tests, Code Snippets, Photos
4	**Component Validation**	Have the components been integrated and tested together?	RFID + ESP32 + LCD + Google Sheet integration	Yes	Circuit Diagram, Pin Mapping Table, Video Demo
5	**System/Subsystem Validation**	Has the system been tested in a representative environment?	Full test in college lab with cards and buttons, Google Sheet updates	Yes	Serial Monitor Logs, Sheet Screenshots, Lab Notes
6	**System/Subsystem Demonstration**	Has the system been successfully demonstrated with all features?	Admin summary, error handling (invalid/already voted), vote summary and LCD feedback	Yes	Master Card Response, Summary Table, Google Sheet Logs

Technology Readiness Level (TRL) = 6