KLE Technological University

Creating Value, Leveraging Knowledge

Dr. M. S. Sheshgiri Campus, Belagavi

**Department of**
**Electronics and Communication Engineering**

**Minor Project Report**

**on**

# Advanced Home Security : Face recognition integrated with ESP32


**By:**

1. **Sairaj Patil** USN:02FE21BEC079
2. **Rohit Shekdar** USN:02FE21BEC076
3. **Sakshi Sadare** USN:02FE21BEC082
4. **Rachel Gadvi** USN:02FE21BEC071


**Semester: VI, 2023-2024**

Under the Guidance of

**Prof. Neelam Somannavar**

KLE Technological University,
Dr. M. S. Sheshgiri College of Engineering and Technology
BELAGAVI-590 008
2023-2024


Dr. M. S. Sheshgiri Campus, Belagavi

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

# CERTIFICATE

This is to certify that project entitled **"Advanced Home Security : Face recognition integrated with ESP32"** is a bonafide work carried out by the student team of **"Sairaj Patil 02fe21bec079,Rohit Shekdar 02fe21bec076,Sakshi Sadare 02fe21bec082, Rachel Gadvi 02fe21bec071"**. The project report has been approved as it satisfies the requirements with respect to the mini project work prescribed by the university curriculum for B.E. (VI Semester) in Department of Electronics and Communication Engineering of KLE Technological University Dr. M. S. Sheshgiri CET Belagavi campus for the academic year 2023-2024.

| | | |
|---|---|---|
| **Prof. Neelam Somannavar** | **Dr. Dattaprasad A. Torse** | **Dr. S. F. Patil** |
| **Guide** | **Head of Department** | **Principal** |

**External Viva:**

**Name of Examiners** **Signature with date**

1.

2.

# ACKNOWLEDGMENT

I Acknowledge my guide I extend my sincere appreciation to those who have contributed to the successful completion of this mini project. Their expertise, guidance, and support have been instrumental throughout the entire process.

I would like to express my deepest gratitude to,

Our mentor Neelam S , Your technical acumen and insightful feedback have greatly enriched the content of this report. Your dedication to precision and thoroughness has been a guiding force.I would also like to thank Dr . Arun .T. , Prof Amey .M ,Prof Hunagund .S .M.Your invaluable input and constructive suggestions have significantly improved the overall quality of this technical document. Your commitment to excellence is truly commendable.

I am also thankful to Dr Udayakumar Naik our subject coordinator for their constant support and guidance through out the program Collectively, their collaborative efforts have played a pivotal role in shaping the outcomes of this technical endeavor. This report stands as a testament to the synergy of our collective expertise.I would also like to acknowledge the collective efforts of the entire project team, including Sairaj .P,Sakshi .S,Rohit .S,Rachel .G for their collaboration and hard work have been pivotal in achieving our goals.

This project stands as a testament to the collaborative spirit and dedication of each individual involved. Thank you for your commitment to excellence.

-The project team

# 1 Abstract

The Internet of Things' (IoT) rapid advancements have made it possible to implement creative smart home solutions that improve security and convenience. This project describes the design and implementation of a smart door lock system that uses the ESP32-CAM module and a Telegram bot for remote control and monitoring. The system provides a strong security solution by allowing users to take pictures of people at their doorstep and remotely lock or unlock the door using Telegram commands.An ESP32-CAM module, a relay-controlled electronic lock, a flash LED, and a push button make up the system's essential parts. The core device for taking pictures and making internet connections is the ESP32-CAM, which has a camera and Wi-Fi capabilities. An easy-to-use interface is offered by the Telegram bot for giving commands and getting alerts.

Among the smart door lock system's primary features are:

Photo Capture and Notification: When the push button is hit or a certain Telegram command is received, the system takes a picture. The user's Telegram account receives the taken image, which provides real-time visual information about the visitor. Remote Door Lock Control: By giving the Telegram bot short commands, users can remotely lock or unlock the door. With this function, customers may manage who has access to their house from anywhere, increasing ease and security. Unauthorised Access Alert: To make sure that only authorised users are able to manage the door lock, the system will respond with a warning message if an unauthorised user tries to interact with the bot.This project showcases a practical IoT application for home security, providing real-time monitoring and control through internet-connected devices, and offers a cost-effective and user-friendly solution for smart home enthusiasts and security-conscious individuals.

# Contents

## 2 Problem Statement

Building a advanced home security solution the leverages cutting edge technologies like facial recognition integrated with ESP32-CAM micro controller.

## 3 Introduction

n today's rapidly evolving landscape of smart home technologies, the integration of advanced devices like the ESP32-CAM with Telegram bot functionality marks a significant stride towards enhancing both security and convenience for homeowners. This project focuses on developing a sophisticated smart door lock system designed to offer robust protection while enabling seamless remote access and monitoring capabilities. At its core, the ESP32-CAM module serves as a pivotal component, equipped with a high-resolution camera and Wi-Fi connectivity. This allows the system to capture images of individuals at the doorstep and transmit them securely to the user's Telegram account in real-time. Through the Telegram bot interface, users gain the ability to remotely control the door lock with simple commands, ensuring they can effortlessly grant or restrict access to their residence from any point. The system also includes tools to notify users of attempted illegal access, offering a complete security solution that puts user accessibility and safety first. In addition to meeting the growing need for smart home innovations, this project sets the standard for useful and efficient home security solutions catered to current lifestyles by utilising the power of IoT and user-friendly communication platforms such as Telegram.

# 4 Literature Survey

**[1] Smart home automation and security with ESP**

| | |
|---|---|
| **Objectives** | Implement a smart home system using IoT for automation and security, integrating ESP32-CAM, 8-Relay Module, and NodeMCU. |
| **Method Used** | Utilized hardware components and software integration for remote device control, facial recognition, and real-time alerts via a custom Android app, leveraging Wi-Fi connectivity and web-based APIs. |
| **Limitations** | potential security vulnerabilities in data transmission, dependency on continuous internet access, scalability issues, and user interface complexity for non-technical users. |

Table 1: Research Objective, Method, and Limitation

**[2] Wi-Fi Door Lock Using Esp32 Cam and Blynk**

| | |
|---|---|
| **Objectives** | Develop a smart Wi-Fi door lock system using the ESP32-CAM and Blynk app to enhance home security by enabling remote monitoring and control via smartphone. |
| **Method Used** | The system integrates the ESP32-CAM module with a solenoid lock, enabling facial recognition for authentication and sending captured images to the Blynk app for homeowner verification. |
| **Limitations** | Potential reliability issues with Wi-Fi connectivity affecting real-time notifications and remote access, as well as dependency on continuous power supply. |

Table 2: Objectives, Method, and Limitation

**[3] IoT based wi-fi fingerprint door lock system with raspberry pi and webcam**

| | |
|---|---|
| **Objectives** | To develop a secure and advanced door lock system using IoT technologies, integrating fingerprint authentication, live streaming via webcam, and remote access via WiFi for enhanced home security. |
| **Method Used** | The system integrates a biometric fingerprint sensor for user authentication, a Raspberry Pi for processing and connectivity, and a webcam for live streaming of the entrance area. |
| **Limitations** | Limited scalability in handling a large number of users due to hardware constraints of Raspberry Pi and potential bandwidth issues for live streaming. |

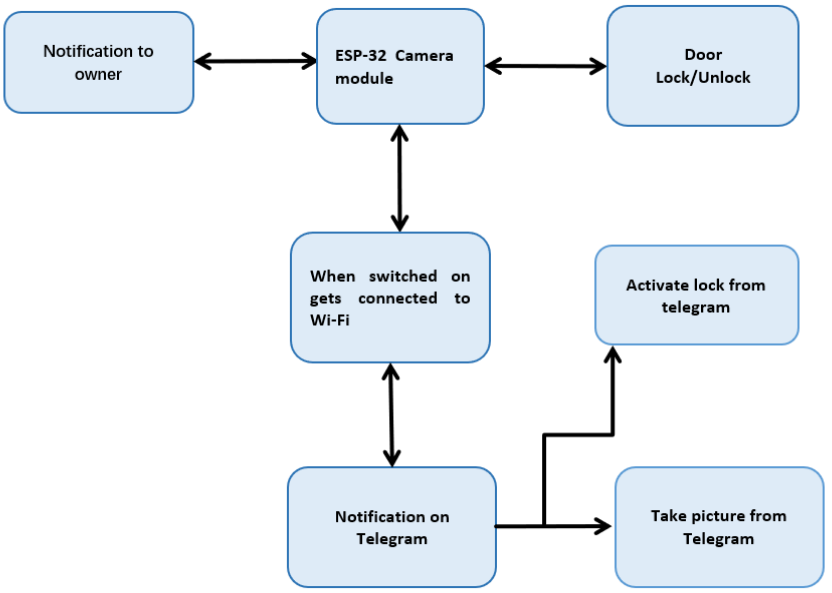
Table 3: Objectives, Method, and Limitation

**[4] IoT Based Smart Door Lock System**

| | |
|---|---|
| **Objectives** | To develop a smart door lock system using IoT technology, specifically utilizing ESP32 CAM and Blynk App, enhancing home security through remote monitoring and control. |
| **Method Used** | The system employs ESP32 CAM for video surveillance and Arduino Uno for door control, integrated with Blynk App for remote operation via Wi-Fi, ensuring secure access management. |
| **Limitations** | Limited reliability during power outages due to dependency on Wi-Fi connectivity for remote access, potentially compromising security in case of network disruptions. |

Table 4: Objectives, Method, and Limitation

# 5 Functional Block Diagram



# 6 Implementation details

1. **Hardware Components**
   - ESP32-CAM Module: Utilized for its integrated camera, Wi-Fi connectivity, and GPIO pins for controlling the electronic lock.
   - Electronic Lock: A relay-controlled mechanism to physically lock and unlock the door.
   - Flash LED: Provides illumination for the camera in low-light conditions.
   - Push Button: Triggered to capture a new photo upon physical interaction or via Telegram command.
2. **Software Components**
   - Configure the ESP32-CAM module to connect to the local Wi-Fi network using the provided SSID and password.
   - Initialize the camera settings (resolution, frame rate, etc.) using the esp-camera library.
   - Set up GPIO pins to control the electronic lock and flash LED.
3. **Telegram Bot Integration**
   - Create a Telegram bot using the BotFather on Telegram and obtain the bot token.
   - Configure the ESP32-CAM to communicate with the Telegram API using the UniversalTelegramBot library and secure HTTPS requests.
   - Handle incoming commands (/photo, /lock, /unlock, /start) from users to trigger actions such as capturing photos or controlling the door lock state.
4. **Image Capture and Transmission**
   - Implement a function to capture an image using the ESP32-CAM's camera module.
   - Convert the captured image to JPEG format and prepare it for transmission.
   - Use HTTPS POST requests to send the image along with appropriate metadata (e.g., chat ID) to the Telegram bot API endpoint (/sendPhoto).
5. **Door Lock Control**
   - Define functions to toggle the state of the electronic lock based on received commands (/lock to lock, /unlock to unlock).
   - Use GPIO pins to control the relay that operates the electronic lock mechanism.
   - Ensure robust error handling and safety precautions to prevent unintended access.
6. **Telegram Bot Message Handling**
   - Implement a function to continuously check for new messages using the Telegram bot API.
   - Process incoming messages to determine the appropriate action (e.g., capture photo, lock/unlock door) based on the command received.
   - Validate user permissions and send responses and notifications accordingly.
7. **Security Considerations**

- Secure Communication: Use HTTPS for all communication between the ESP32-CAM and Telegram bot API to protect user data and commands.
- Access Control: Implement authentication mechanisms to ensure only authorized users can interact with the bot and control the door lock.
- Error Handling: Incorporate robust error handling and recovery mechanisms to handle network interruptions, camera failures, or unexpected errors gracefully.

8. **Deployment and Testing**

   - Deployment: Install and mount the ESP32-CAM module securely near the door, ensuring optimal camera positioning and Wi-Fi connectivity.
   - Testing: Conduct thorough testing to validate each component's functionality, including camera capture quality, responsiveness of Telegram bot commands, and reliability of door lock control.

# 7 Optimization

1. **Code Optimization**

   Minimize Delay and Blocking Calls: Refactor code to reduce delays caused by delay() functions or blocking operations. Use non-blocking techniques like millis() for timing operations and implement state machines or finite state automata to manage asynchronous tasks effectively.

   Memory Management: Optimize memory usage by carefully managing buffers and dynamically allocating memory only when necessary. Use stack memory wherever possible instead of heap memory to reduce fragmentation.

   Reduce Image Processing Overhead: Adjust camera settings (resolution, frame rate) to balance image quality with processing and transmission speed. Use JPEG compression parameters (jpeg-quality) effectively to optimize image size without compromising quality.

2. **Network Optimization**

   HTTPS Optimization: Implement HTTP keep-alive connections to reduce overhead associated with establishing new connections for each HTTPS request. Utilize HTTP persistent connections (clientTCP.setReuse(true)) to reuse connections when sending multiple requests.

   Data Compression: Consider implementing data compression techniques (e.g., gzip) for reducing the size of HTTP payloads, especially when transmitting images or large data sets.

3. **Power Management**

   Deep Sleep Mode: Implement deep sleep modes between operations to minimize power consumption during idle periods. Configure the ESP32-CAM to wake up periodically or in response to external triggers (e.g., button press) to conserve battery life.

   Low-Power Components: Select and use low-power components where possible, such as low-power LEDs for status indicators or low-power electronic locks that consume less current during operation.

4. **Error Handling and Recovery**

   Robust Error Handling: Enhance error handling mechanisms to gracefully recover from network failures, camera capture errors, or Telegram API timeouts. Implement retry mechanisms with exponential backoff strategies to retry failed operations.

   Logging and Debugging: Implement logging mechanisms to capture debug information, errors, and exceptions. Use serial debugging output and logging libraries (e.g., Serial.println(), ESP-LOG for ESP-IDF) to monitor system behavior and diagnose issues.

5. **Security Enhancements**

   Secure Authentication: Implement secure authentication mechanisms (e.g., HMAC-based message authentication) for verifying commands and requests sent to the ESP32-CAM from the Telegram bot. Validate and sanitize user inputs to prevent malicious commands.

   Data Encryption: Use TLS/SSL (HTTPS) for secure communication between the ESP32-CAM and the Telegram API. Encrypt sensitive data (e.g., images) before transmission to protect against eavesdropping and data interception.

# 8 Result and Discussions

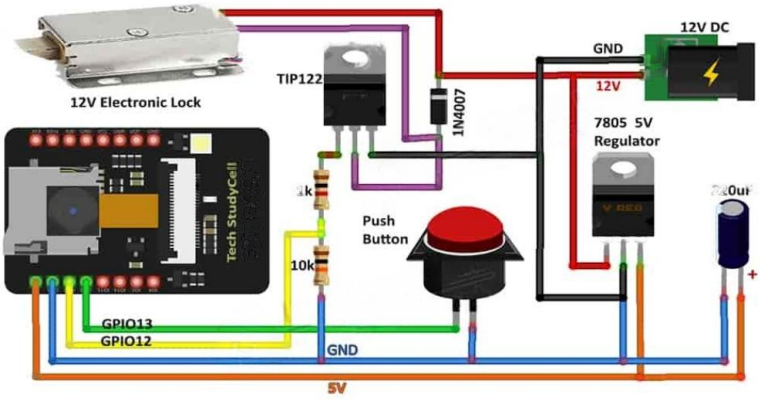



Figure 1: ESP32 door locking system circuit

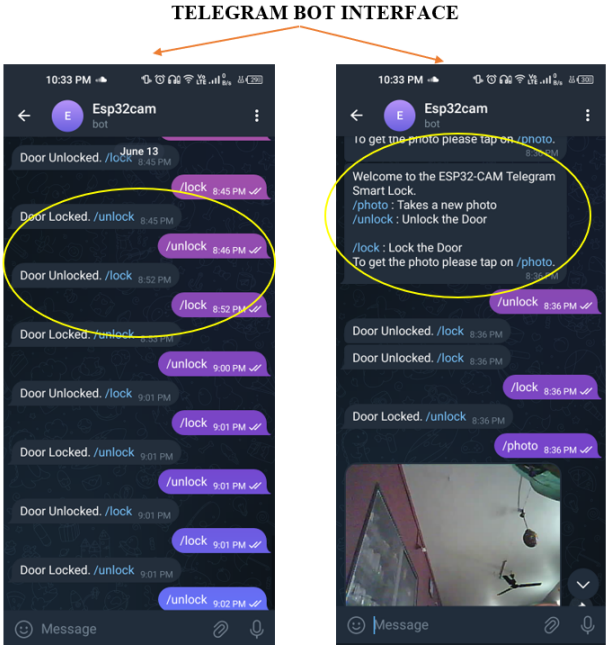



Figure 2: Telegram interface for controlling


Figure 3: Images captured for lock/unlock

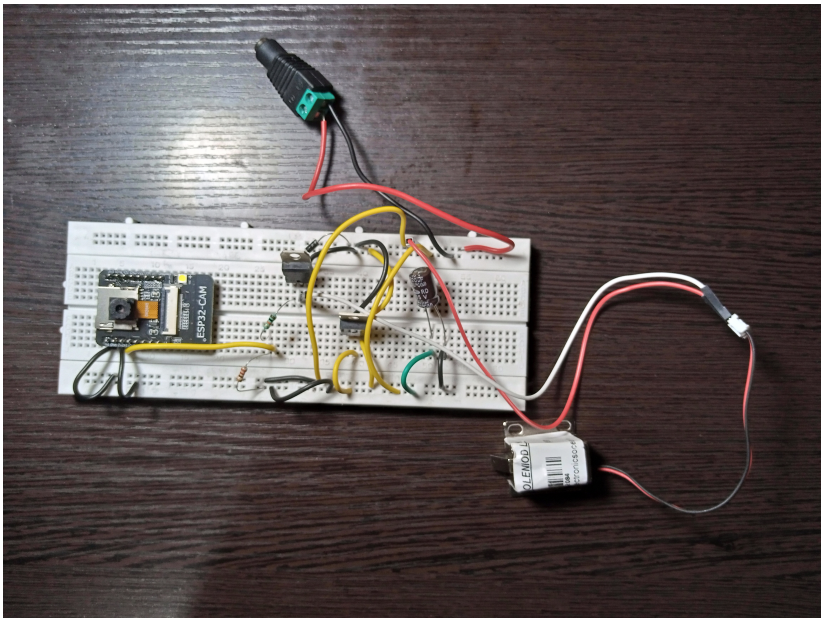


Figure 4: Real time model circuit

## 8.1 Working of model

**Telegram Bot Interface** The Telegram bot serves as the primary user interface for interacting with the door lock system. It allows users to send commands and receive feedback, making the operation of the system straightforward and user-friendly. The following key functionalities are provided through the Telegram bot interface:

- /start Command: When a user sends the /start command, the bot responds with a welcome message that includes instructions on how to use the system. This message lists available commands such as /photo, /unlock, and /lock.
- /photo Command: This command prompts the ESP32-CAM to capture a photo and send it to the user via Telegram. This is particularly useful for verifying who is at the door before unlocking it.
- /unlock and /lock Commands: These commands allow the user to unlock or lock the door remotely. The bot provides feedback on the current state of the door lock, ensuring that the user is always aware of the system's status.

**ESP32-CAM Image Capture** The ESP32-CAM is responsible for capturing images whenever the door lock state is changed (either locking or unlocking) or when the /photo command is issued. The images are captured and then transmitted over the internet to the user via the Telegram bot. This real-time image capture enhances security by allowing users to see who is at their door before deciding to unlock it.

- Image Quality and Format: The camera is configured to capture high-quality JPEG images, ensuring that the photos sent to the user are clear and detailed. This is crucial for security purposes, as it allows for accurate identification of visitors.
- Flash LED Activation: The system is designed to activate the flash LED when capturing a photo, ensuring that the images are well-lit even in low-light conditions. This feature is important for capturing clear images at any time of day.

# 9 Conclusions and Future Scope

## 9.1 Future Scope for Security Enhancement

1. **Enhanced Encryption and Authentication Protocols:** Implement advanced encryption algorithms and secure authentication protocols (e.g., OAuth 2.0) to protect sensitive data transmitted between the ESP32-CAM module and the Telegram bot. Strengthening encryption ensures confidentiality and integrity of data, guarding against unauthorized access and tampering.

2. **Multi-Factor Authentication (MFA):** Introduce multi-factor authentication techniques to verify user identities before granting access permissions. Combining something the user knows (e.g., password) with something they have (e.g., mobile device for Telegram authentication) enhances security against credential theft or unauthorized access attempts.

3. **Real-time Intrusion Detection and Alerts:** Implement AI-based algorithms or machine learning models to analyze live camera feeds for detecting suspicious activities or unauthorized access attempts. Integration with cloud-based intrusion detection systems can trigger immediate alerts to homeowners and authorities, mitigating potential security breaches.

4. **Physical Security Enhancements:** Evaluate and upgrade physical components of the smart door lock system, such as using tamper-resistant locks and reinforced enclosures to prevent unauthorized manipulation or tampering of the device.

5. **Regular Security Audits and Updates:** Establish a schedule for conducting regular security audits and firmware updates to address vulnerabilities and patch security flaws promptly. This proactive approach ensures that the system remains resilient against evolving cyber security threats.

## 9.2 Conclusion

A major development in home security technology is the use of ESP32-CAM in the implementation of a smart door lock system with Telegram bot integration. The solution offers homes strong safety and convenience by utilising the ESP32-CAM module's picture capture and Wi-Fi connectivity capabilities, in conjunction with Telegram bot functionality for remote control and monitoring. Users' security and peace of mind are increased by the integration, which enables them to safely monitor and manage access to their houses from anywhere in the globe.Through meticulous hardware configuration and software development, the project has successfully demonstrated the feasibility of combining IoT (Internet of Things) devices with cloud-based messaging platforms to create an intelligent and responsive home security solution. The system's efficacy in tackling contemporary security concerns is highlighted by its real-time image capturing and transmission capabilities, as well as its remote locking and unlocking capabilities through Telegram instructions.

# 10 References

[1] "Smart home automation and security with ESP", International Journal of Emerging Technologies and Innovative Research (www.jetir.org — UGC and issn Approved), ISSN:2349-5162, Vol.8, Issue 5, page no. ppb205-b209, May-2021.
[2] Divya, P Devaraj, V Aswin, V. (2023). Wi-Fi Door Lock Using Esp32 Cam and Blynk. Electrical and Automation Engineering. 2. 125-132. 10.46632/eae/2/1/18.
[3] Shirisha, P., G. Satish Kumarr, K. Shivanjan, K. Shiva Rama Krishna, and K. Vineela."IoT based wifi fingerprint door lock system with raspberry pi webcam." In MATEC Web of Conferences, vol. 392, p. 01066. EDP Sciences, 2024.
[4] Reddy, Kondamu Yashaswini, Ardha Jyothsna Reddy, K. Bhanu Prakash Reddy, and Mr B. Srinivasa Rao. "IoT Based Smart Door Lock System." (2022).