ElevateX:

The Future of Crowd - Aware Elevators

TEAM MEMBERS:

Shobhit Poddar: ______________________

Ujjawal Singh: ______________________

Raushan Kumar: ______________________

Suryansh Subham: ______________________

Ashutosh Kumar: ______________________

SUPERVISED BY

Prof. (Dr.) Sheli Sinha Chaudhuri

NETAJI SUBHASH ENGINEERING COLLEGE
(Affiliated to MAKAUT, Approved by AICTE)

____

Supervisor

Abstract:

The "IoT-Based Smart Lift System with Automatic Floor Gathering and Priority Sensing Acquisition" is a modern automation project that enhances traditional elevator systems using Internet of Things (IoT) technology. This system uses ESP32 microcontrollers along with camera-based sensors to detect the number of people waiting on each floor and prioritize lift movement based on real-time demand.

The smart lift collects data from multiple floors through strategically placed IR sensors or cameras, which count the number of people waiting. This data is processed by the ESP32, which calculates floor-wise priorities—such as higher urgency for hospitals, schools, or emergency floors. Based on this priority logic, control signals are sent to the motor driver and relay module to navigate the lift efficiently, minimizing wait time and energy usage.

The system also integrates Wi-Fi connectivity, allowing users or administrators to monitor the lift status via a mobile app or cloud dashboard (e.g., using Blynk or Firebase). The app provides real-time notifications and control commands. This intelligent lift mechanism is particularly useful in environments where crowd management and priority handling are essential.

This project not only demonstrates real-time automation and IoT communication but also focuses on improving accessibility, reducing energy consumption, and enabling remote control and monitoring.

Introduction :

In today's fast-paced world, automation and intelligent systems are playing a critical role in improving efficiency and user experience in both residential and commercial infrastructure. Elevators, or lift systems, are essential components of multi-story buildings, yet most traditional systems follow a fixed logic—serving calls in the order they are received without considering real-time occupancy or urgency.

The **IoT-Based Smart Lift System** aims to overcome these limitations by introducing **smart floor detection and priority-based control** using the **ESP32 microcontroller**. The system integrates **cameras or IR sensors** at each floor to **detect the number of people waiting**. This data is then transmitted via **Wi-Fi** to the ESP32, which processes it to determine **which floor should be served first**, based on **crowd density and pre-set priorities** (such as for hospitals, schools, or emergency areas).

five past academic and institutional projects on smart/elevator systems with IoT, AI, and contactless technologies.

1. Lift Me Please: Smart Elevator Development Using IoT

Institution: Southeast Asia University, Thailand
Summary:
A contactless elevator command system using Raspberry Pi and QR-code scanning via mobile app. Supports both online cloud mode and offline camera mode, records usage history, and achieved 100% accuracy in operation, with average wait time ~5.6 s (compared to ~3 s manually) arXiv+15ResearchGate+15Nature+15.
Link: ResearchGate PDF (May 2024)

2. A Study on the Non-Contact Artificial Intelligence Elevator System Due to COVID-19

Journal: *Electronics* (MDPI), 2024
Summary:
An AIoT elevator system featuring facial recognition and contactless floor selection. Applied in high-rise contexts to reduce wait and travel times, especially during COVID-19, by enabling touchless, face-activated elevator calls MDPI.
Link: MDPI Sciences

3. Connected Smart Elevator Systems for Smart Power and Time Saving

Publication: *Scientific Reports* (Nature), 2024
Summary:
This project integrates YOLO-based image detection with IoT elevator systems, achieving a **15% reduction in wait times** and **20% energy savings**. It includes object detection, energy analytics, predictive maintenance, and smart building connectivity ResearchGate+1Nature+1.
Link: Nature Scientific Reports

4. Towards Contactless Elevators with TinyML using CNN-based Person Detection and Keyword Spotting

Source: arXiv, 2024
Summary:
Describes a microcontroller-based *TinyML* elevator system utilizing CNNs for person detection and voice commands, achieving ~5 s latency, 83% detection accuracy, and 80% keyword spotting, enabling touchless, low-cost elevator calls with minimal infrastructure changes

ijeast.com+9arXiv+9MDPI+9.
Link: arXiv

5. Development of IoT Device for Human Activity Detection Near Smart Elevator Area

Institution: Tallinn University of Technology, Estonia (2020)
Summary:
A master's project designing a battery-operated IoT device with PIR sensor and NB-IoT connectivity. It automatically detects humans near elevators and sends reservation calls to the elevator system. Promising concept but faced power and latency challenges digikogu.taltech.ee.
Link: Tallinn University PDF

Problem Statement:

Traditional elevator systems operate on a **sequential or fixed logic**, responding to floor requests in the order received without considering the **urgency, number of waiting passengers**, or **real-time priority needs** such as emergencies or crowd management. This leads to:

- Increased **waiting time** for passengers
- **Inefficient energy usage** due to unnecessary lift movement
- Lack of adaptability in **sensitive environments** like hospitals or schools
- No real-time **data analysis or remote monitoring** capability

There is a clear need for a **smart, dynamic, and priority-based lift system** that can respond intelligently to real-world demands using **IoT technology and intelligent sensing**.

Objectives:

1. **To design and implement** an IoT-based lift system using ESP32 that detects the number of people waiting on each floor using IR sensors or cameras.
2. **To develop a priority-based algorithm** that decides the next floor based on real-time crowd data and predefined urgency levels (e.g., hospital > school > others).
3. **To enable wireless communication** between sensor nodes and the ESP32 using Wi-Fi (HTTP or MQTT protocols).
4. **To automate lift movement** using a motor driver (e.g., L298N) and relay modules, controlled by the ESP32 microcontroller.
5. **To integrate a mobile or web-based interface** (e.g., Blynk, Firebase, or a custom app) for remote monitoring and control of the lift system.

6. **To reduce energy consumption and improve efficiency** by optimizing lift movements based on demand and priority.
7. **To build a scalable and cost-effective prototype** that can be adapted to various building environments such as hospitals, educational institutions, and residential complexes.

Methodology:

1. Component Selection and Research

Selecting the right components is critical to ensure **functionality, compatibility, cost-efficiency**, and **reliability** of the smart lift system. Below is a categorized list of the main components along with their roles and selection criteria:

Components and their Purpose:

Component	**Purpose / Function:**
ESP32 Dev Board	Acts as the central controller for processing sensor data, running priority logic, and controlling the lift movement. Built-in Wi-Fi is used for IoT communication.
IR Sensors / Cameras	Placed on each floor to detect the number of people waiting for the lift. Data is sent to the ESP32. Cameras can be used for advanced crowd counting using image processing.
L298N Motor Driver	Drives the DC motor or stepper motor to move the lift up/down based on instructions from the ESP32. Controls direction and speed.
12V DC Motor / Stepper Motor	Provides the mechanical motion for lifting the elevator cabin between floors. Stepper motors allow precise positioning.
SG90 Servo Motor	Controls the door opening and closing of the lift cabin when the lift reaches a floor.
2-Channel Relay Module	Acts as a switch to turn ON/OFF high-power devices like motors or external power supplies safely through ESP32’s low voltage logic.

Power Supply (5V/12V)	Provides required voltage and current to power the ESP32, motor driver, sensors, and actuators.
Wi-Fi Router	Facilitates wireless communication between ESP32, floor sensors (cameras), and mobile/cloud dashboard.
Smartphone	Acts as a remote interface to monitor lift status, view live data (floor, crowd count), or control it manually via Blynk/Firebase app.
Buzzer & LED (Optional)	Provide audio/visual alerts (e.g., lift arrival, overload, emergency)
OLED Display (Optional)	Displays current floor, lift direction, or number of people detected – helpful for user feedback.
Laptop/Mobile (with OpenCV)	Used to process images from cameras and perform real-time person detection and counting before sending data to ESP32.

Table 1: Component list with their use

2. System Design and Planning

The system design of the IoT-based smart lift system focuses on integrating real-time sensing, wireless communication, and automated lift control using a modular and scalable approach. The design ensures smooth interaction between hardware, software, and IoT infrastructure.

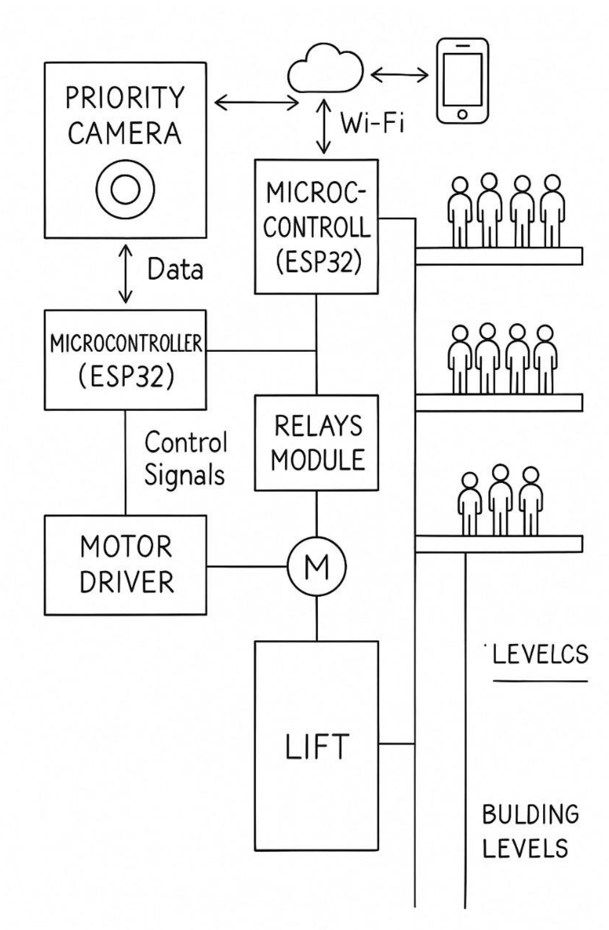


Fig1: Block Diagram of the Circuit

3. Sensor Integration and Calibration

Sensor integration is a critical part of the smart lift system, as it enables real-time detection of human presence or crowd size on each floor. These sensor inputs are used by the ESP32 to determine which floor needs to be served based on crowd density and assigned priority.

4. Circuit Assembly

The circuit assembly combines all core hardware elements: the **ESP32 controller**, **sensors**, **motor driver**, **servo motors**, and **power supply** to form a functioning prototype. Proper wiring and connectivity are crucial for safe and reliable operation.

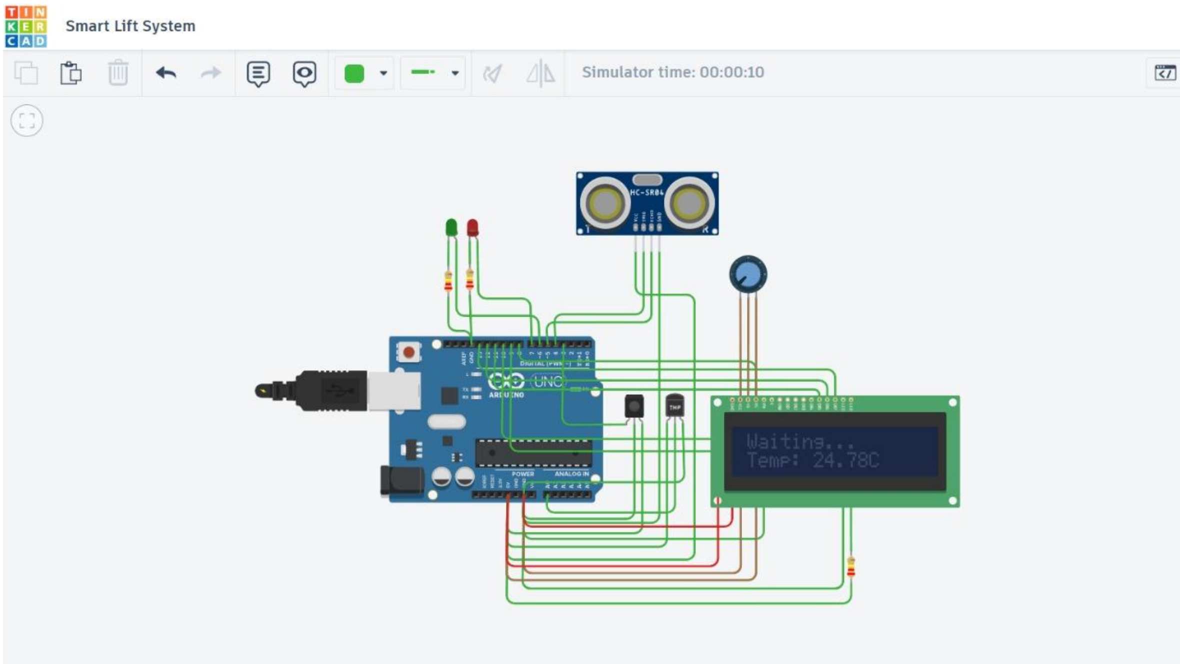
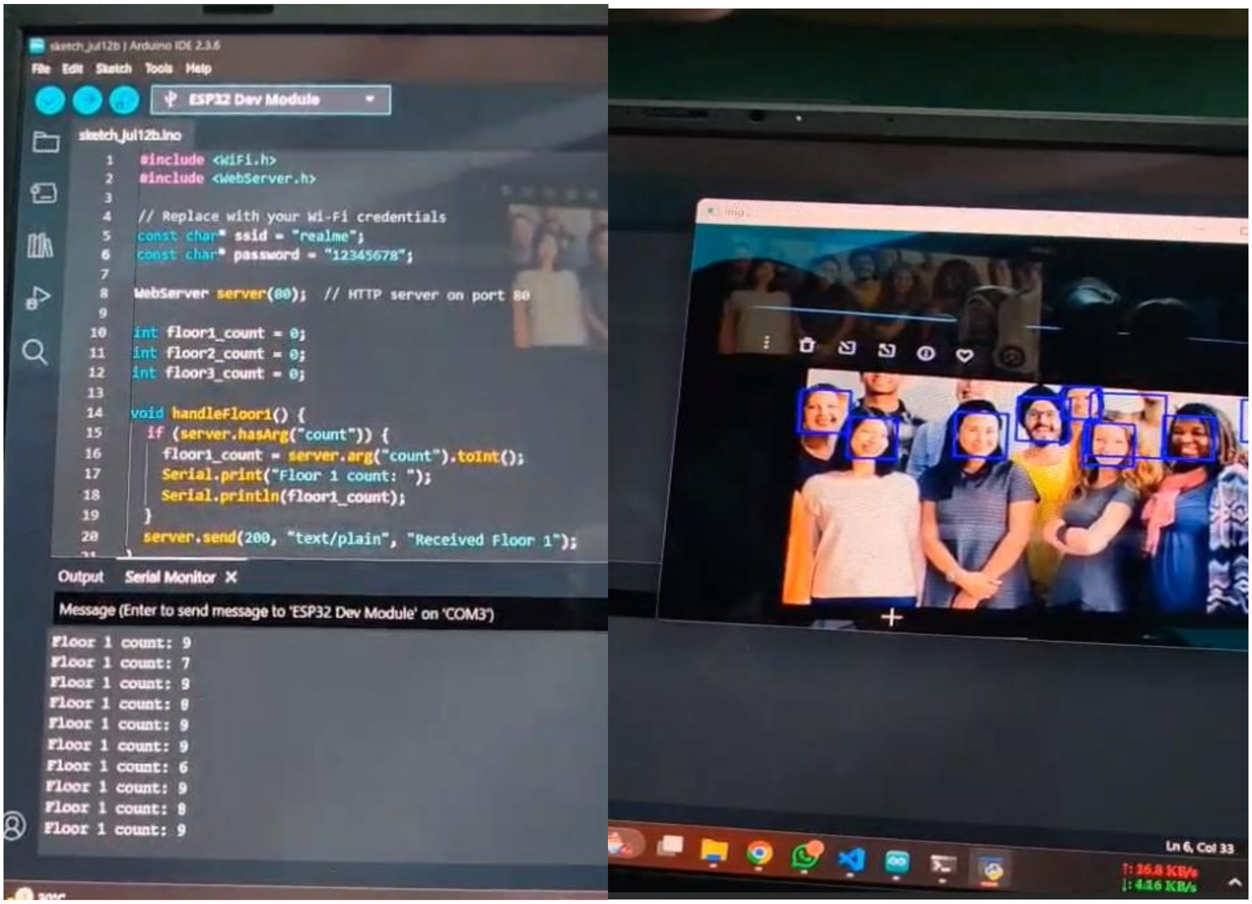


Fig 2: TinkerCAD Simulation Circuit

5. Software Development

The software development process involves writing and integrating the code logic that drives the entire smart lift system — from receiving sensor data to controlling lift motion and communicating with the user interface.



🧱 Key Software Modules

Module	Purpose
Sensor Data Processing	Reads IR or camera data (floor-wise person count) and stores it in memory.
Priority Calculation Logic	Computes which floor should be served next based on people count and floor priority.
Lift Control Logic	Sends motor commands to move the lift and controls the door mechanism.
Web Server / Communication	Receives real-time data from floor sensors (via HTTP) or from a camera system.
App / Cloud Integration	Optional module to show lift status and control remotely via Blynk/Firebase.

Core Logic Flow:

1. Start system

2. Receive people count from each floor (via IR/camera + HTTP)

3. Assign priority score:

   Priority = People_Count × Area_Weight

4. Select floor with highest score

5. Move lift to selected floor using motor

6. Stop at floor and open/close door (servo)

7. Update status to app/cloud

8. Repeat

6. Enclosure Fabrication:

To create a presentable and functional prototype, a miniature lift model was fabricated using cardboard, thermocol, or acrylic sheets. The structure included three visible floors with IR sensors or cameras installed on each floor for people detection. A pulley system driven by a DC motor was mounted to enable vertical motion of the lift cabin. All electronics including ESP32, motor driver, and relay module were placed neatly on a side panel or behind the model for easy access and wiring.

7. Testing and Optimization:

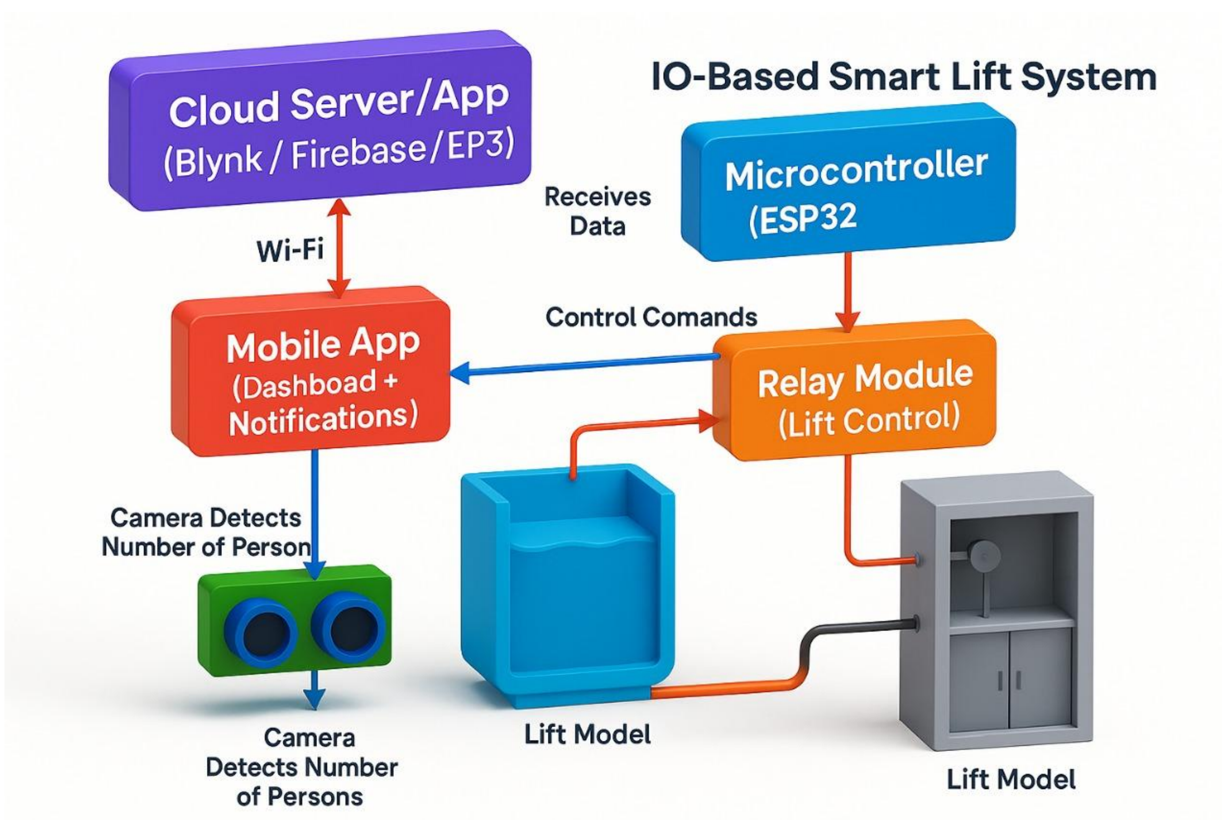
Each component was tested individually to ensure proper functionality. IR sensors were calibrated for accurate person detection, the motor was tested for smooth up/down movement, and the servo motor was tuned to operate the lift door correctly. The priority algorithm was tested with various combinations of crowd input to validate floor selection. Network stability, response time, and mechanical alignment were optimized for efficient operation and reliability.

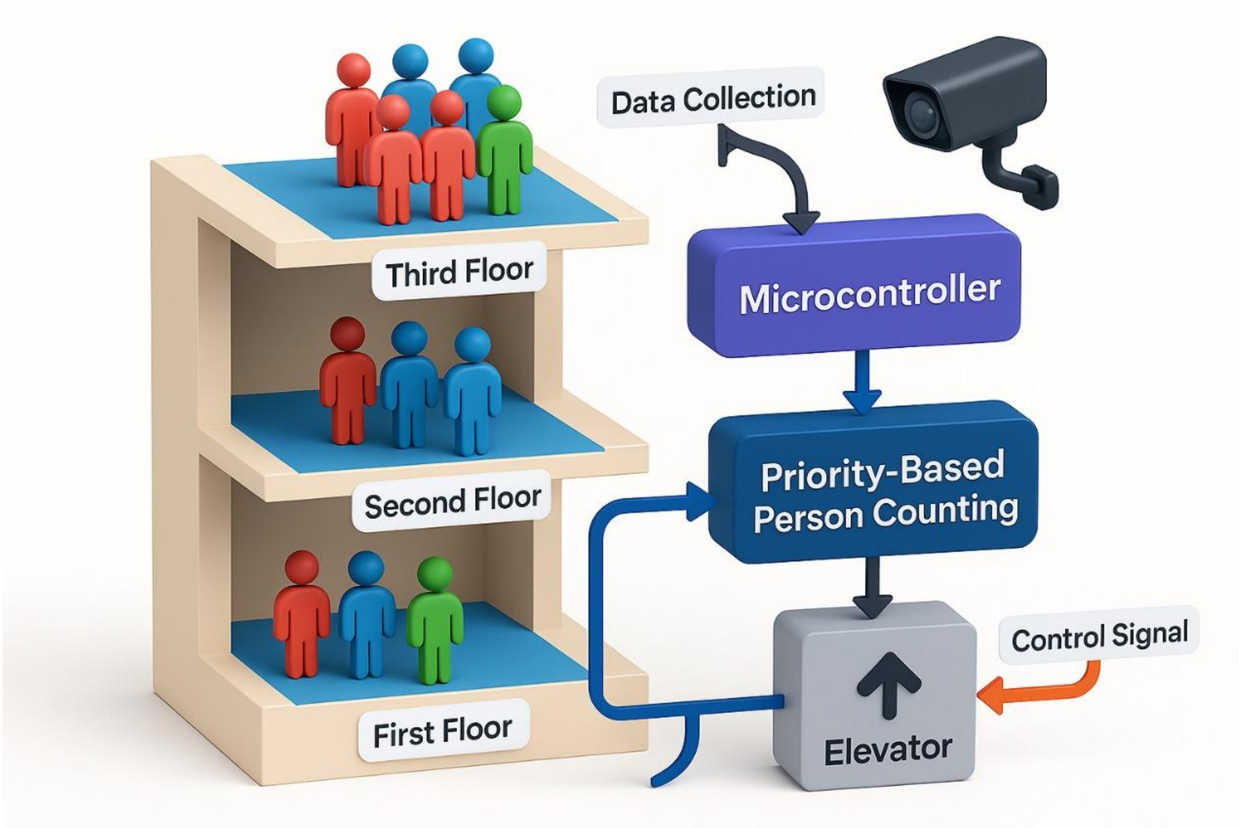
8. Final Integration:

After successful testing, all modules—sensing, logic, actuation, and communication—were fully integrated into a single system. The ESP32 was programmed to receive real-time input from floor sensors and operate the lift based on calculated priority. The motor and servo operated as per instructions, and live status updates were provided via a mobile app or display. The completed model successfully demonstrated the smart lift's ability to prioritize floors dynamically and function autonomously.

Blueprint:

The blueprint of the IoT-Based Smart Lift System includes a 3-floor lift model with IR sensors or cameras on each floor to detect crowd presence. An ESP32 microcontroller processes this data and prioritizes floors based on urgency. A DC motor and pulley system control the lift's movement, while a servo motor opens and closes the door. All components are mounted inside a compact enclosure, and the system is monitored or controlled via a mobile app, making it a fully automated and smart lift prototype.





Applications:

- **Hospitals** :Prioritizes emergency floors to ensure faster access for critical cases.
- **Schools and Colleges** : Manages student crowd efficiently during entry/exit times.
- **Office Buildings** : Reduces lift waiting time during peak hours by crowd-based routing.
- **Residential Apartments** : Provides smart and energy-efficient lift operation.
- **Shopping Malls** : Handles heavy foot traffic intelligently with real-time crowd detection.
- **Smart Cities** : Fits into IoT-based infrastructure for automated public transport systems.
- **Industrial Facilities** : Can be used to control material lifts based on load and usage frequency.

Future Scope:

- **AI-based Decision Making**: Integrate machine learning to predict crowd patterns and optimize lift scheduling automatically.
- **Advanced Camera Vision**: Use facial recognition or advanced object detection for better accuracy in people counting and access control.
- **Cloud Integration**: Store usage data in the cloud for analytics, remote diagnostics, and maintenance tracking.
- **Access Control Systems**: Add RFID, fingerprint, or facial ID to allow only authorized users to access certain floors.
- **Voice & Gesture Control**: Enable hands-free lift operation through voice assistants or gesture recognition for improved accessibility.
- **Energy Optimization**: Include regenerative braking and smart energy-saving algorithms to reduce power consumption in large buildings.
- **Multi-Lift Coordination**: Extend the system to manage and synchronize multiple lifts within a building for maximum efficiency.
- **Commercial Deployment**: Customize the system for different sectors like hospitals, airports, metro stations, and smart buildings.

Conclusion

The IoT-Based Smart Lift System provides an innovative solution to traditional lift management by introducing crowd-based floor prioritization using sensors and IoT technology. By integrating IR sensors or cameras with an ESP32 microcontroller, the system can detect the number of people on each floor and determine the most urgent floor based on predefined priorities. The automatic lift movement and door control reduce manual intervention, save time, and improve user experience. This prototype demonstrates the potential of combining real-time sensing, automation, and wireless communication for smarter urban infrastructure. Its cost-effectiveness, flexibility, and ability to scale make it suitable for environments like hospitals, schools, offices, and residential complexes. With further development, it can be enhanced using AI, cloud analytics, and multi-lift coordination for broader smart city applications.

References :

1. **ESP32 Datasheet** – Espressif Systems
   https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
2. **Arduino IDE Documentation** – Arduino Official
   https://docs.arduino.cc
3. **L298N Motor Driver Module Guide** – Electronics Hub
   https://www.electronicshub.org/l298n-motor-driver-module
4. **OpenCV Python Documentation** – OpenCV.org
   https://docs.opencv.org
5. **Blynk IoT Platform** – Blynk Official
   https://docs.blynk.io
6. **Firebase Realtime Database Guide** – Google Firebase
   https://firebase.google.com/docs/database
7. P. Kamble, A. Kotecha, and N. Shah, "Smart Elevator System using IoT," *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 5, May 2020.
8. R. Rani, S. Mishra, "Automatic Lift Control System Based on Crowd Monitoring," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, Issue 9, 2020.
9. TutorialsPoint – IoT and ESP32 Basics
   https://www.tutorialspoint.com/iot/index.htm