IoT Smart Home

Muhammd Suleman Akram
Lab Of Future
Dubai, United Arab Emirates
sulemanakram806@gmail.com

Nicolas Antonini
Lab Of Future
Dubai, United Arab Emirates
nialantonini@gmail.com

Hassan Al Darwish
Lab Of Future
Dubai, United Arab Emirates
hassanaldarwish7@gmail.com

Adeeba Asif
Lab Of Future
Dubai, United Arab Emirates
adeebaasif07@gmail.com

Abdullah Ali
Lab Of Future
Dubai, United Arab Emirates
hima8dolli@hotmail.com

***Abstract*—**Smart home automation has become a key application of the Internet of Things, enabling intelligent control and monitoring of household environments. This paper presents the design and implementation of an AI enabled smart home automation system that integrates a physical IoT prototype with a mobile application and voice assistance. The system allows users to control home appliances such as lights, fans, doors, and curtains through both manual interaction and natural language voice commands. Voice inputs are transcribed using Apple's speech recognition framework and processed by a Python based AI model to identify user intent. The interpreted commands are transmitted through the MQTT protocol to an ESP32 microcontroller, which controls sensors and actuators in the physical prototype. In addition, environmental temperature data is collected and transmitted back to the mobile application for real time monitoring. The implemented system demonstrates reliable communication, intuitive user interaction, and effective integration of IoT, AI, and mobile technologies for smart home automation.

***Keywords*—**Internet of Things, Smart Home, Artificial Intelligence, Voice Control, MQTT, ESP32

I. Introduction

The Internet of Things has enabled the development of intelligent environments in which physical devices can communicate and operate automatically. Smart home automation systems represent one of the most common applications of IoT, offering improved comfort, security, and energy efficiency. These systems allow users to monitor and control household appliances remotely using connected devices.

Traditional smart home systems often rely on manual application interfaces, which may reduce usability and accessibility. Recent developments in artificial intelligence and voice recognition have introduced more natural interaction methods. Voice assistance allows users to control systems using spoken language, improving ease of use. This paper presents a smart home automation system that combines IoT hardware, a mobile application, and AI based voice assistance to create an interactive home environment.

II. System Overview

The proposed system consists of three main layers: the mobile application layer, the AI and backend processing layer, and the IoT hardware layer. The mobile application acts as the primary interface for user interaction and supports both manual control and voice commands. The AI backend processes user input and determines appropriate actions, while the IoT hardware layer executes commands and collects sensor data.

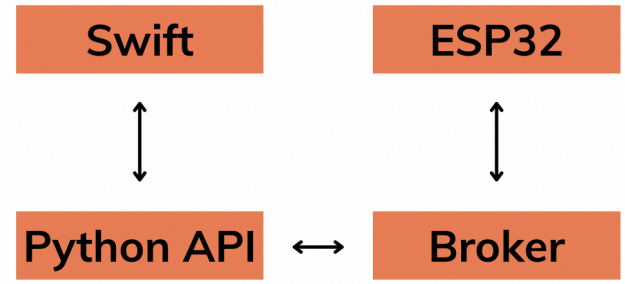


Figure 1: Overall system architecture diagram

As shown in Fig. 1, user commands originate from the mobile application and are processed by the AI backend before being transmitted to the ESP32 microcontroller using the MQTT protocol. Sensor data follows the reverse path to provide real time feedback to the user.

III. Hardware Design

A. Physical prototype

A small scale physical model of a home was constructed to demonstrate the proposed smart home system. The prototype includes controllable lighting, a fan, automated curtains, and a door mechanism.

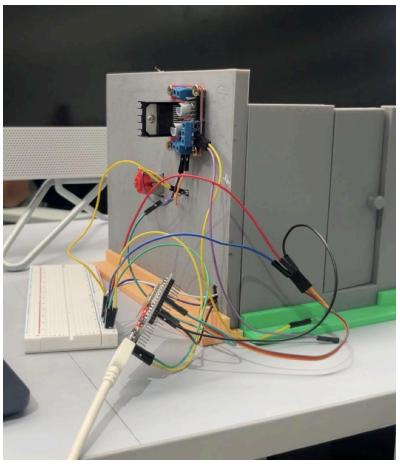


Figure 2: Physical smart home prototype photo

B. Microcontroller

An ESP32 microcontroller was selected as the central control unit due to its built-in Wi Fi capability, processing performance, and suitability for IoT applications. The ESP32 manages communication with the MQTT broker and controls all connected sensors and actuators.

C. Sensors

A temperature sensor was integrated to monitor environmental conditions inside the prototype. The ESP32 periodically reads temperature values and publishes the data to the MQTT broker for processing and display in the mobile application.

D. Actuators

The lighting system is controlled using digital output pins.
The fan is operated using a motor driver module.
The curtains are controlled using a motor and motor driver to enable opening and closing actions.
The door is operated using a servo motor to simulate locking and unlocking.

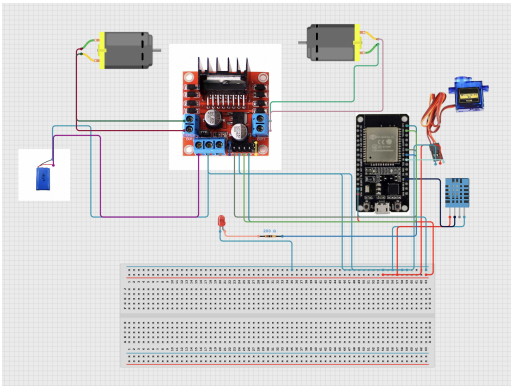


Figure 3: Overall circuit overview

IV. Mechanical Design and Fabrication

Custom mechanical components were designed using SolidWorks to support the physical implementation of the system. These components were fabricated using three dimensional printing to ensure proper alignment, stability, and smooth motion.

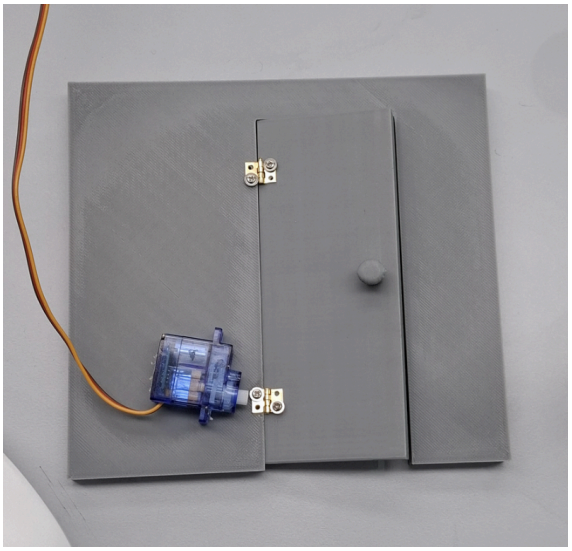


Figure 4: SolidWorks design of door mechanism

The measurements of the door roughly measure a length of 150mm and a width of 140mm, using a 5% fill on the inside. The door was printed with a length of 125mm and 50mm. To attach the door to the respective wall/door frame we used 10mmx8mm hinges which were attached using a hand drill and screwdriver to allow for the door to swing open.

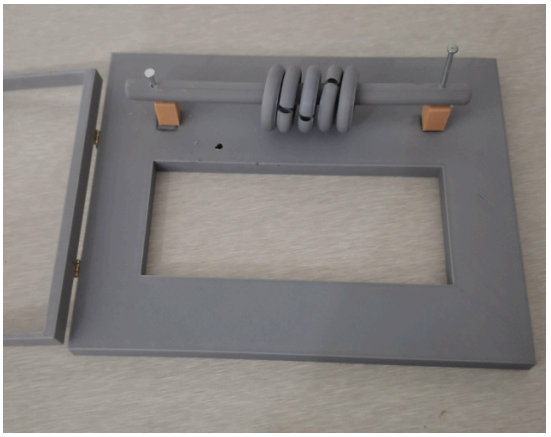


Figure 5: SolidWorks design of curtain mechanism

The first piece in was the window frame wall, a simple extruded rectangle with a window in the middle. We then created two brackets of equal height (roughly 20mm) to support the curtain pole.We then printed 5 rings with small slots to allow for fabric to be connected. Once all printed these were all assembled using a drill to connect the

brackets to the pole (attached by a nail) and then glued with a hot glue gun.

V. Software Architecture

A. Mobile Application

The mobile application was developed using Swift for iOS devices. The application includes a home screen that welcomes the user and provides manual controls for lights, fan, door, and curtains. Real time temperature data is also displayed.

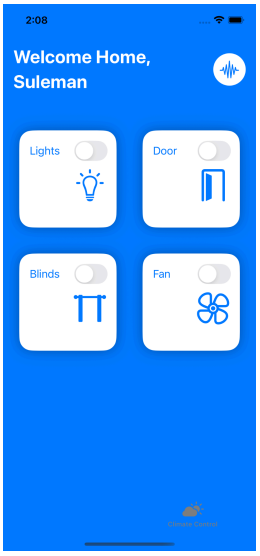
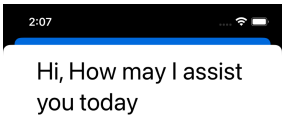
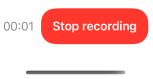


Figure 4: iOS application home screen

B. Voice Assistant Interface

The application includes a voice assistant feature accessed through a microphone button. When activated, a prompt appears stating “Hi, how may I assist you today”. Users can issue spoken commands to control devices.





VI. Artificial Intelligence and Communication

A. Voice Recognition and Intent Detection

User voice input is converted into text using Apple’s speech recognition framework. The transcribed text is sent to a Python based API, where an AI model analyzes the command and determines the user intent. For example, a command such as “Turn on the fan” is interpreted as a corresponding control action. The AI system also generates a response that is returned to the mobile application.

B. MQTT Communication

The MQTT protocol is used to transmit commands and sensor data between the backend system and the ESP32. Identified intents are published to specific MQTT topics, which the ESP32 subscribes to and executes. Temperature readings collected by the ESP32 are published back to the system and displayed in the mobile application.

VII. Results and Discussion

The system successfully demonstrated real time control of household components through both manual and voice interaction. Voice commands were accurately transcribed and interpreted, resulting in reliable execution on the physical prototype. The MQTT protocol provided efficient and low latency communication between components. Temperature monitoring improved system awareness and user feedback.

VIII. Conclusion and Future Work

This paper presented an AI enabled smart home automation system integrating IoT hardware, mobile application development, and voice assistance. The system provides intuitive control, real time monitoring, and a scalable architecture. Future work may include expanding AI intent recognition, adding additional sensors, improving security features, and integrating cloud based analytics.

Acknowledgments

The author(s) would like to thank Karam Ghazal for their guidance and support throughout the development of this project.

References

[1] Eclipse Foundation, *Eclipse Mosquitto™ – An Open Source MQTT Broker*, [Online]. Available: https://mosquitto.org/

[2] Apple Inc., *Speech Framework Documentation*, [Online]. Available: https://developer.apple.com/documentation/speech

[3] Espressif Systems, *ESP32 Series Datasheet*, [Online]. Available: https://www.espressif.com/en/products/socs/esp32