Title: Concise and descriptive project title.

Home automation through voice control

Background: Overview of the problem domain and the motivation behind the project.

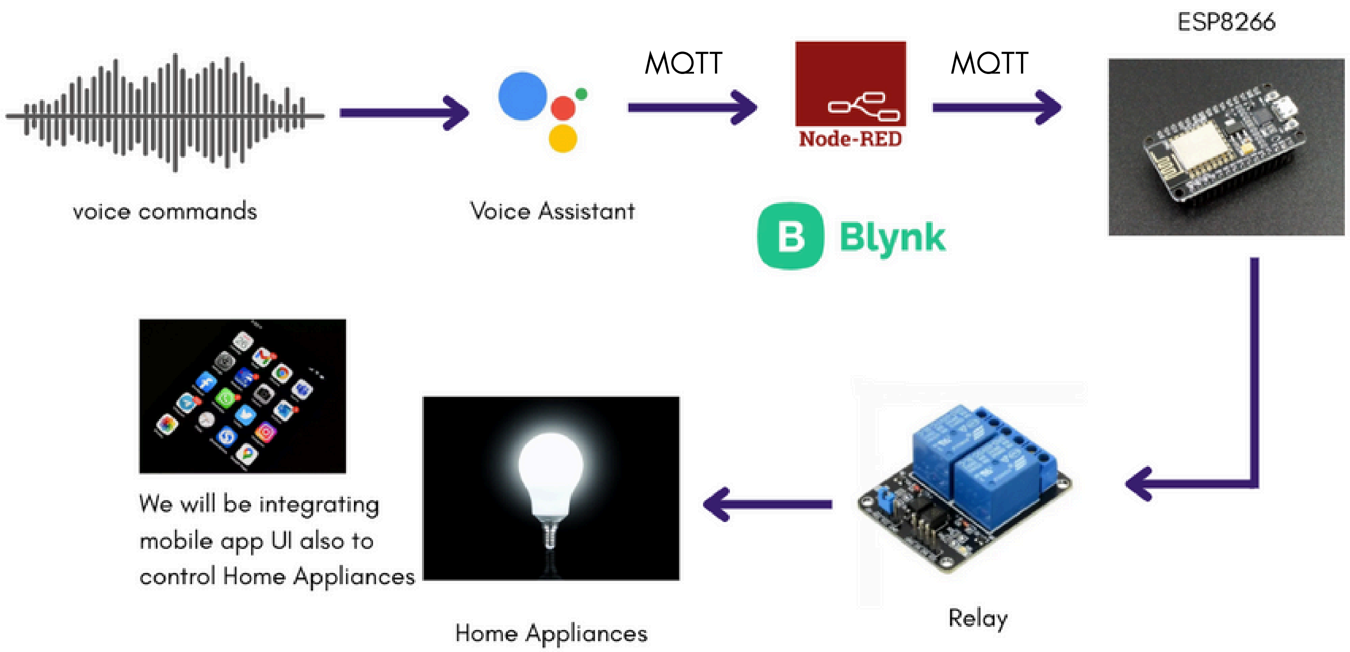
In our daily lives as students (even if we don’t consider student life), we often find ourselves too exhausted or too comfortable to get up and manually control basic appliances like lights and fans—especially late at night, during cold weather, or in the middle of intense study sessions. At times, we’re so sleepy while studying that even standing up to turn off the lights feels like a task. These everyday inconveniences sparked the motivation for this project: to design a smart, voice-controlled home automation system that enhances comfort, saves energy, and makes our student lifestyle more manageable.

Additionally, we plan to introduce an innovative safety feature: if someone shouts for help—by saying phrases like “Help! Help!”—the system will recognize it as an emergency and automatically initiate a voice call or send an alert message to a predefined emergency contact. Depending on the user's location, it could also call emergency services (e.g., 911 if user is outside Pakistan or 15 within Pakistan). This added layer of intelligent response can be especially valuable in situations where immediate help is needed and the user is unable to physically reach their phone.

Project Idea: Detailed description of the proposed solution and its objectives.

In this project, we will demonstrate the ability to control household appliances like lights, fans, and possibly other devices through voice commands and through phone UI, making daily tasks more convenient. Additionally, the system will feature an emergency response capability, where a emergency voice commands will trigger an automatic call or send a message to emergency contacts. This functionality aims to enhance both convenience and safety by enabling quick assistance in critical situations.

Home Automation Through Voice Control



We will be integrating mobile app UI also to control Home Appliances

Objectives:

- Comfort for Users: Make it easy for people, especially with mobility issues, to control appliances using voice and mobile app.
- Remote Control: Allow users to turn off lights or fans remotely if they forget when leaving home.
- Energy Saving: Reduce energy waste by enabling appliance control from anywhere.
- Safety and Accessibility: Make it easier and safer for users to control appliances without physical effort.

Approach: Methodology and technologies to be used in the project.

The project combines voice-activated automation with Google Assistant, IFTTT, Node-RED, and MQTT(HTTP as an alternative) to operate smart devices (lights, fans, etc.). Below is the step-by-step methodology:

1. Voice Command Input:

- The user issues a command through Google Assistant (e.g., "Turn on the light").
- Google Assistant invokes an IFTTT Applet via voice integration.

2. IFTTT (If This Then That) Webhook Integration:

- IFTTT triggers a Webhook, which sends an HTTP request (or MQTT message if that protocol is used) to Node-RED.

3. Node-RED Processing:

- Node-RED receives the HTTP (or MQTT) request through the HTTP In node (or MQTT In node if MQTT is used).
- A Function node in Node-RED processes the request, constructing a control message (e.g., turning on/off the appliance).
- The processed message is sent to the broker using the HTTP Out node (for HTTP) or MQTT Out node (for MQTT).

4. Device Control:

- The MQTT broker (or HTTP server, if using HTTP) receives the command and forwards it to the subscribed devices (e.g., NodeMCU/ ESP8266).
- The ESP8266 or other microcontrollers execute the action (e.g., turning on a light) through relays or GPIO pins.

Components Required: List of hardware and software components needed.

Hardware Parts

- Microcontroller (IoT Device)
- NodeMCU (ESP8266) or ESP32 (Wi-Fi enabled, MQTT compatible).
- Relay Module
- 5V Relay Module (for switching high-voltage appliances such as lights/fans).
- Power Supply
- 5V Power Adapter (for relay & ESP8266).
- Jumper Wires & Breadboard (for testing).
- Electrical Appliances (e.g., LED bulb, fan).

Software & Cloud Services

- Google Assistant: Provides voice input.
- IFTTT: Connects Google Assistant to Node-RED via Webhooks.
- Node-RED: Handles automation and logic processing, running on a Raspberry Pi OS, local server, or cloud (e.g., IBM Cloud).
- MQTT Broker: Uses Mosquitto (local) or cloud-based brokers like HiveMQ or Adafruit IO.
- Firmware for ESP8266: Programmed using the Arduino IDE with MQTT client logic

Challenges Foreseen: Potential obstacles and risks, along with mitigation strategies.

1.Workflow Errors in Node-RED

- Node-RED flows can get complicated quickly, which might lead to logic errors or unexpected behavior.

Mitigation: We'll keep our flows simple, test each step thoroughly, and document everything for better debugging.

2.MQTT Disconnection Problems

- If the MQTT broker (Mosquitto) disconnects, the whole communication between devices can break.

Mitigation: We'll ensure a stable mobile hotspot and enable automatic reconnects in our setup.

References:

- https://www.researchgate.net/publication/339543872_IoT_based_VoiceText_Controlled_Home_Appliances
- https://nodered.org/docs/
- https://mosquitto.org/documentation/

END