

National University of Sciences & Technology

School of Electrical Engineering and Computer Science Department of Computing

**CS 335: Internet of Things**

**Fall 2023**

**Class: BESE-11AB**

Project Report

Smart Home

Instructor: Dr. Rafia Mumtaz

**Names:**

**Alina Nasir [342350]**

**Faiq Jamal [344994]**

**Misbah Noor Awan [333811]**

**Mydah Nasir [333191]**

**Muhammad Ali Usman [332608]**

**Section: BESE-11 A**

Contents

[Project Description: 2](#_Toc154165520)

[Introduction 2](#_Toc154165521)

[Problem statement: 2](#_Toc154165522)

[Interoperability Challenges: 2](#_Toc154165523)

[User Interface and Analytics: 2](#_Toc154165524)

[Energy Efficiency Optimization: 2](#_Toc154165525)

[Objectives: 2](#_Toc154165526)

[Application Requirements 3](#_Toc154165527)

[Hardware Components 3](#_Toc154165528)

[Sensors 3](#_Toc154165529)

[Supporting Equipment 4](#_Toc154165530)

[More Power Optimization Techniques 5](#_Toc154165531)

[1. Sleep Mode 5](#_Toc154165532)

[2. Interrupt-Driven Operation 5](#_Toc154165533)

[3. Duty Cycling 5](#_Toc154165534)

[Practical Considerations 5](#_Toc154165535)

[Circuit Diagram 6](#_Toc154165536)

[Tinkercad Simulation 7](#_Toc154165537)

[Features and Logic Guide 8](#_Toc154165538)

[1. Proximity Sensing 8](#_Toc154165539)

[Threshold Distance 8](#_Toc154165540)

[2. Ambient Temperature Monitoring 8](#_Toc154165541)

[Threshold Temperature: 8](#_Toc154165542)

[3. Motion Sensing 9](#_Toc154165543)

[Threshold for Detection 9](#_Toc154165544)

[4. Smoke Detector using Gas Concentration Detection 9](#_Toc154165545)

[Threshold Gas Concentration: 9](#_Toc154165546)

[5. Street Light using Light Intensity Measurement 9](#_Toc154165547)

[Threshold Light Level: 10](#_Toc154165548)

[6. Automatic Door and Remoted Control Lights using Infrared Remote-Control Interaction 10](#_Toc154165549)

[Suitable Communication Technology: 10](#_Toc154165550)

[Individual Components: 10](#_Toc154165551)

[Automatic Doors: 10](#_Toc154165552)

[Automatic Light Control: 11](#_Toc154165553)

[Automatic Temperature Control: 11](#_Toc154165554)

[Fire Alarm: 11](#_Toc154165555)

[Network Topology: 11](#_Toc154165556)

[Mesh Topology: 11](#_Toc154165557)

[Physical Layer: 12](#_Toc154165558)

[Zigbee (Local Communication): 12](#_Toc154165559)

[Wi-Fi or Ethernet (Internet Connectivity): 12](#_Toc154165560)

[Internet Layer: 12](#_Toc154165561)

[Internet Protocol (IP): 12](#_Toc154165562)

[Application Layer: 12](#_Toc154165563)

[MQTT (Message Queuing Telemetry Transport): 12](#_Toc154165564)

[HTTPS (Hypertext Transfer Protocol Secure): 12](#_Toc154165565)

[Tools and technologies: 12](#_Toc154165566)

[Data Visualization/Analysis and system interface: 13](#_Toc154165567)

[Dashboard of System: 13](#_Toc154165568)

[Humidity 13](#_Toc154165569)

[Temperature 13](#_Toc154165570)

[Raw Ethanol 14](#_Toc154165571)

[Pressure: 14](#_Toc154165572)

[Average CO2: 15](#_Toc154165573)

[Fire Incidents: 15](#_Toc154165574)

[Predictive Analysis: 16](#_Toc154165575)

[Heat Map: 16](#_Toc154165576)

[Data Pre-processing: 17](#_Toc154165577)

[Data Splitting: 17](#_Toc154165578)

[Model Training and Hyper parameter tuning: 17](#_Toc154165579)

[Model Testing and Evaluation: 18](#_Toc154165580)

[Expected Impact: 18](#_Toc154165581)

[Automation and Convenience: 19](#_Toc154165582)

[Remote Access and Monitoring: 19](#_Toc154165583)

[Energy Efficiency: 19](#_Toc154165584)

[Cost Savings: 19](#_Toc154165585)

[Improved Quality of Life: 19](#_Toc154165586)

[Environmental Impact: 19](#_Toc154165587)

Smart Home System

# Project Description:

## Introduction

In the era of technological advancements, the concept of smart homes has gained significant popularity. Smart homes leverage Internet of Things (IoT) technology to automate and control various aspects of daily living, offering increased convenience, energy efficiency, and security. However, there remains a need for further innovation and integration to address existing challenges and provide a seamless, intelligent, and user-friendly smart home experience.

## Problem statement:

While existing smart home systems offer a range of functionalities such as remote control, energy management, and security, there are still notable limitations and challenges that hinder the widespread adoption and optimal performance of smart homes. The current landscape presents the following issues:

### Interoperability Challenges:

Many smart home devices and platforms operate in isolation, lacking standardized communication protocols. This results in compatibility issues and limits the ability to create a unified, interoperable ecosystem.

User Interface and Analytics:

The user interfaces of existing smart home applications are often complex and fragmented without much dashboards.

Energy Efficiency Optimization:

Although smart homes aim to enhance energy efficiency, there is room for improvement. The lack of intelligent algorithms to analyse usage patterns and optimize energy consumption remains a challenge.

## Objectives:

The objective of this project is to develop an integrated, user-friendly, and secure IoT-based solution that addresses the aforementioned challenges in smart homes. The proposed solution should focus on:

1. **Interoperability:** Establishing a standardized communication protocol for seamless integration of diverse smart home devices and platforms.
2. **Clean Circuit Design:** To build a clean working circuit that would enable the user to control the features of home automation
3. **Analytics:** Creating an intuitive dashboard displaying the analytics of data to the users
4. **Energy Optimization:** Developing intelligent algorithms to analyse user behaviour, device usage patterns, and environmental factors to optimize energy consumption within the smart home environment.

# Application Requirements

The following are the requirements for the of the project

1. Automatic Doors and Lights using Remote Control
2. Ambient Temperature Monitoring
3. Fire Alarm/Smoke Detector
4. Motion Sensing
5. Proximity Sensing
6. Analytics Dashboard
7. Machine Learning Model

# Hardware Components

## Sensors

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Function** | **Power Consumption** | **Optimizing Power Consumption** |
| Ultrasonic Distance Sensor HC-SR04 | Utilizes ultrasonic waves to measure distance by calculating the time taken for waves to bounce back from an object. | Around 15mA during active operation (triggering and receiving ultrasonic waves), approximately 2mA in idle mode. | Utilize a microcontroller to trigger the sensor only, when necessary, implement sleep modes, and use hardware interrupts to wake the sensor. |
| Temperature Sensor TMP36 | Measures temperature with an analogue output proportional to the temperature. | Typically, less than 50μA. | Use low-power modes of the microcontroller, sample temperature at longer intervals, and power off the sensor when temperature data isn’t required frequently. |
| PIR Sensor | Detects motion by sensing changes in infrared radiation within its field of view. | Around 50-100μA in idle mode, spikes to several mA when detecting motion. | Use interrupts to wake the microcontroller only when motion is detected, implement a time-based sleep strategy when no motion is detected for an extended period. |
| Gas Sensor | Detects specific gases in the environment. | Varies significantly based on the type and model of the sensor, typically ranging from a few milliwatts to a few watts during active sensing. | Power the sensor selectively and periodically for measurements, employing calibration and using low-power modes if available. |
| Photoresistor | Measures ambient light intensity by changing its resistance based on light exposure. | Very low, typically in the range of microamps. | Activate the photoresistor only when light intensity readings are required, utilize interrupts to wake the microcontroller when a certain light threshold is reached. |
| IR Sensor and IR Remote | Detects and transmits infrared signals used for communication or remote-control applications. | Varies but generally low, a few milliamps during operation. | Use power-down modes when not actively receiving or transmitting IR signals, activate the sensor only during specific events or commands. |

## Supporting Equipment

1. Arduino Uno R3: Functions as the main control unit, managing input/output from various sensors and controlling connected devices like lights, door locks, or displays.
2. Breadboard: Provides a platform for creating and testing circuits without soldering. It allows for quick prototyping and connecting various components for testing before final integration.
3. LCD (16x2): Displays information such as temperature readings, system status, or messages from the smart home system. It provides a visual interface for users to interact with the system.
4. Micro Servo: Employed to automate the opening and closing of doors or gates within the smart home. For example, it could control a door lock to grant access based on authorized inputs.
5. Light Bulbs: Controlled by the Arduino to provide lighting automation in different rooms. They can be turned on, off, or dimmed based on sensor inputs or programmed schedules.
6. Piezo: Used for generating audible alerts or notifications. For example, it can sound an alarm when a motion sensor detects movement in a secured area.
7. LEDs (Light-Emitting Diodes): Used as visual indicators for different statuses or alarms. For instance, they can indicate the system's operational status or signal when a specific condition is met.
8. Resistors: Essential in LED circuits to limit current and prevent LED damage. They are used in series with LEDs to control the current passing through them.

These components collectively contribute to the functionality and automation of a smart home system, allowing for control, monitoring, and automation of various aspects like lighting, security, and user interfaces.

## More Power Optimization Techniques

### Sleep Mode

Sleep mode in microcontrollers allows the system to enter low-power states when certain conditions are met. This mode effectively turns off or reduce power to various components of the microcontroller when not actively performing tasks. This reduces overall power consumption.

* Triggered by Inactivity: Implement sleep modes for the microcontroller when the system is idle or when sensors are not actively providing data.
* Configuring Sleep Modes: Utilize the microcontroller's features or libraries to put the system into sleep mode when necessary.

### Interrupt-Driven Operation

Interrupts enable the microcontroller to suspend its current tasks and respond immediately to specific events. This method is more power-efficient than continuously polling sensors as it allows the microcontroller to remain in a low-power state until needed.

* Event-Based Wake-Up: Configure sensors to generate interrupts when significant events occur (e.g., motion detection, data ready).
* Microcontroller Configuration: Set up interrupt handlers in the microcontroller to respond to these sensor-generated interrupts effectively.

### Duty Cycling

Duty cycling involves activating sensors periodically at scheduled intervals rather than keeping them continuously active. It helps in reducing the overall power consumption by limiting the sensor's active time.

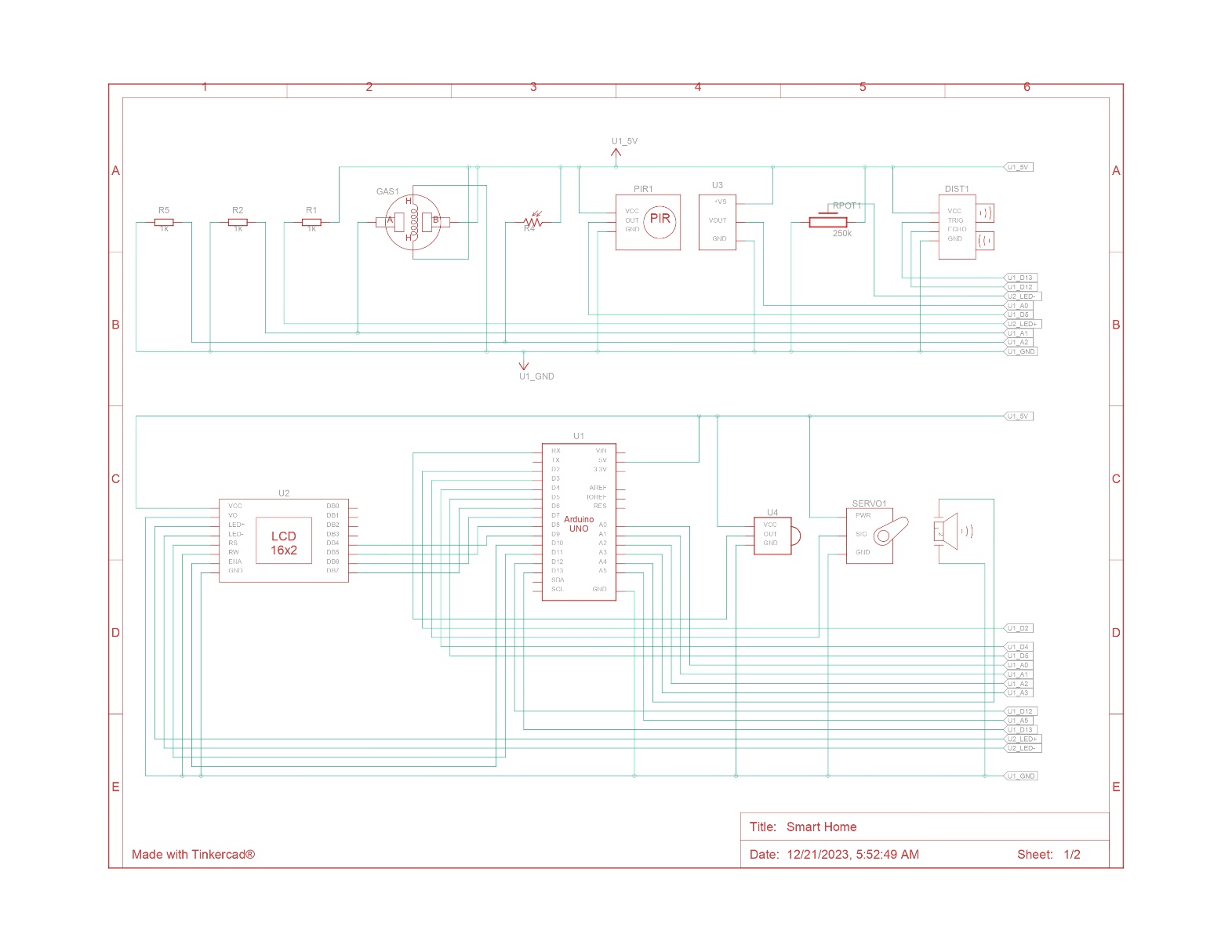
* Time-Based Activation: Determine suitable intervals for sensor activation based on the application's needs (e.g., temperature readings every 10 minutes, motion detection every 30 seconds).
* Dynamic Control: Implement adaptive duty cycles based on sensor data significance or user-defined settings.

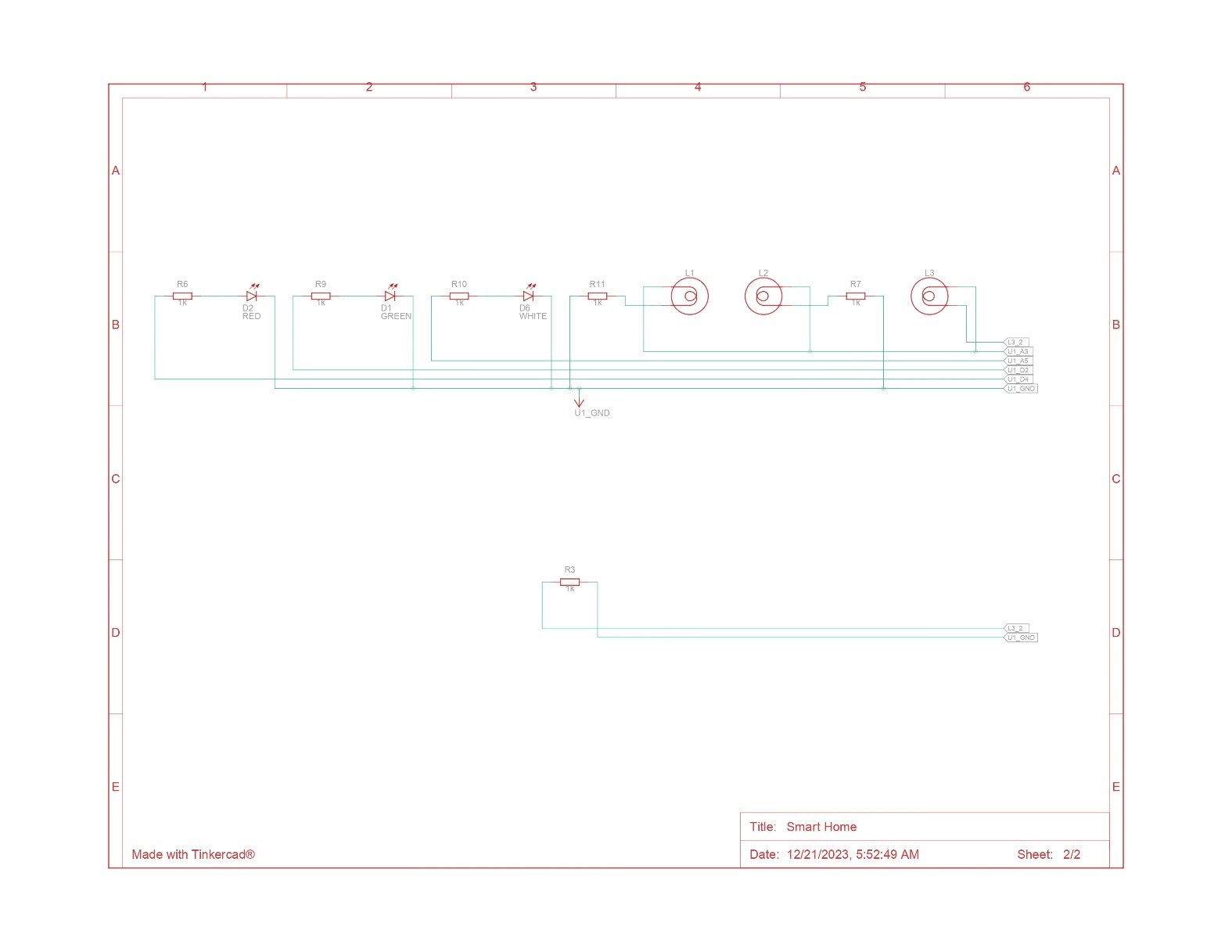
## Practical Considerations

Implementation Complexity: Each optimization technique requires careful consideration and integration within the system architecture.

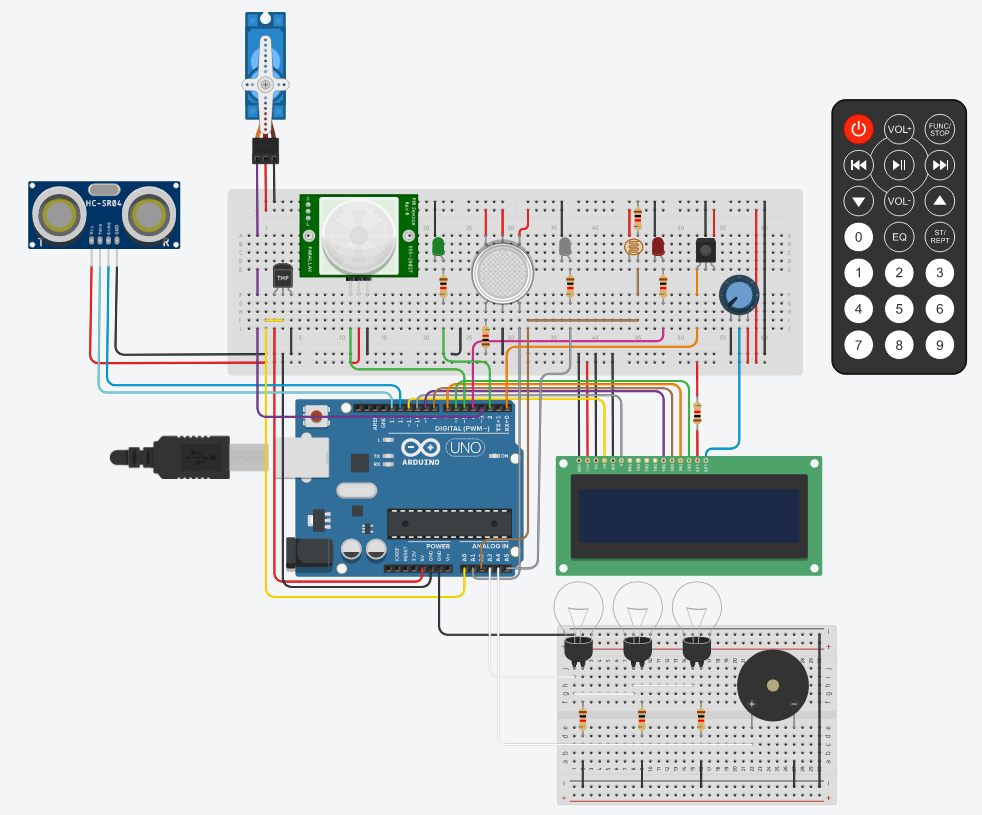
* Trade-offs: While these techniques save power, they might affect real-time responsiveness or data accuracy. Balancing power efficiency with system performance is crucial.
* Sensor Calibration: Adjust duty cycles and sleep modes according to sensor response times and the required frequency of readings.

## Circuit Diagram





# TinkerCad Simulation



Tinkercad Link: <https://www.tinkercad.com/things/erJSBQfE9cV-smart-home/editel?returnTo=%2Fdashboard%3Ftype%3Dcircuits%26collection%3Ddesigns&sharecode=vsbWzy9nYG-9O-1JBL7SJ9m2U8xwXkHKZA2ln9DgNvM>

# Features and Logic Guide

## Proximity Sensing

Utilizing an "Ultrasonic Distance Sensor" for proximity detection, this gadget sends out ultrasonic waves and measures how long they take to bounce back. By calculating this time, it figures out how far something is from the door.

This can be useful for security and monitoring. The measurement of distance from objects (parcels, unknown objects etc) or individuals using ultrasonic waves to determine their closeness to a designated area or door. In this project:

Threshold Distance**:** If the distance is less than a certain value (closingThreshold), it knows that an object or person is very close to the door.

Once an Object/Person is detected, the system sends a notification as well as turns on an LED Light to signify proximity sensing.

𝐶𝑙𝑜𝑠𝑖𝑛𝑔 𝑇ℎ𝑟𝑒𝑠ℎ𝑜𝑙𝑑 = 100

|  |  |  |
| --- | --- | --- |
| Distance | LED | Notification |
| < 100 |  | Detected |
| >100 | ✘ | Nothing Detected |

## Ambient Temperature Monitoring

This feature is like a tiny thermometer for the room, measuring the temperature. An "Analog Temperature Sensor" for environmental temperature assessment is used.

This can be used to monitor and control the temperature of the house to maintain optimum living conditions. In this project:

Threshold Temperature: If the temperature goes above or below a specific value (tempThreshold), it triggers an alarm to alert about extreme temperatures (low and high).

Collecting temperature data from the surroundings and displaying it on a screen. It triggers alerts if the temperature exceeds predefined thresholds.

|  |  |  |
| --- | --- | --- |
| Temperature | Buzzer | LCD |
| < 25 & > 0 |  | Fine |
| > 25 |  | High |
| > 0 |  | Low |

## Motion Sensing

The "Passive Infrared (PIR) Motion Sensor" is used for motion detection.This device detects changes in heat emitted by objects in its field of view, noticing when someone moves.

Threshold for Detection: If it detects a significant change in heat (motion) above a certain value, it triggers a signal.

Identification of motion or movement within a specific area and triggering responses such as illuminating an indicator LED.

|  |  |
| --- | --- |
| Motion | LED |
| Detected |  |
| Not Detected | ✘ |

## Smoke Detector using Gas Concentration Detection

A smoke detector is implemented using an "Analog Gas Sensor" to measure gas concentration. It measures the concentration of certain gases in the air.

Gas levels are monitored in the environment and alerts or notifications raised if gas concentrations exceed safe predefined limits. In this project, a notification is generated and the gas concentration is constantly updated on the LCD screen.

Threshold Gas Concentration: If the sensor detects a concentration higher than a specific level (gasThreshold), it signals a potential gas leak or presence of harmful gases.

|  |  |  |
| --- | --- | --- |
| Gas Concentration | Buzzer | LCD |
| < 10 |  | Low |
| > 10 |  | High |

## Street Light using Light Intensity Measurement

A "Light Dependent Resistor (LDR)" is used to measure light intensity. This sensor changes its resistance based on the amount of light falling on it. In this feature there is Assessment of ambient light levels and adjusting outputs or notifications based on predefined brightness thresholds.

Threshold Light Level: If the room/street becomes darker than a particular level (lightThreshold), it triggers the activation of lights to improve visibility.

|  |  |
| --- | --- |
| Light Intensity | LED |
| > 30 | ✘ |
| < 30 | **✓** |

## Automatic Door and Remoted Control Lights using Infrared Remote-Control Interaction

This feature uses interaction with an "Infrared (IR) Remote Control" for external command reception.

Execution of specific actions based on received commands from an IR remote, allowing remote control of certain functionalities within the system.

These functionalities collectively provide comprehensive monitoring, detection, and control capabilities within the smart home system, utilizing various sensors and technologies commonly integrated into IoT applications.

When any button is pressed, the action is performed and a notification is generated. In this project, mapped controls are given below:

|  |  |
| --- | --- |
| Remote Control Button | Action |
| 0 | Door Closed |
| 1 | Door Opened |
| 2 | Lights turned ON |
| 3 | Lights turned OFF |

# Suitable Communication Technology:

Following is the detail of communication technology that can be used for each feature individually.

## Individual Components:

### Automatic Doors:

* Physical Layer: RFID (Radio-Frequency Identification) technology can be used for communication between the door and the control system.
* Internet Layer: For internet connectivity, protocols such as Wi-Fi or Ethernet may be employed.
* Application Layer: MQTT (Message Queuing Telemetry Transport) or CoAP (Constrained Application Protocol) can be used for communication between the door control unit and the central application.

### Automatic Light Control:

* Physical Layer: Zigbee or Z-Wave for communication between light switches, sensors, and the central control unit.
* Internet Layer: Wi-Fi or Ethernet for connecting the central control unit to the internet.
* Application Layer: MQTT or CoAP for communication between the light control devices and the central application.

### Automatic Temperature Control:

* Physical Layer: Z-Wave or Zigbee for communication between temperature sensors, thermostats, and the central control unit.
* Internet Layer: Wi-Fi or Ethernet for internet connectivity.
* Application Layer: MQTT or CoAP for communication between temperature control devices and the central application.

### Fire Alarm:

* Physical Layer: Wired or wireless communication protocols for smoke detectors, heat sensors, and the central control unit.
* Internet Layer: Wi-Fi or Ethernet for internet connectivity.
* Application Layer: MQTT or CoAP for communication between fire alarm devices and the central application.

## Network Topology:

### Mesh Topology:

The sensors in our network such as temperature sensors, humidity sensors and motion sensors will be connected to ZigBee routers. These routers will in turn be connected to a ZigBee coordinator that in turn will be connected to the internet using Wi-Fi. The Zigbee coordinator can then send data to cloud for further processing. Below is the detail of how this can be achieved:

* The Zigbee coordinator collects temperature readings and door status information from Zigbee-enabled sensors within the home.
* Using MQTT, the coordinator publishes this data to specific topics on the cloud MQTT broker.
* The cloud platform's IoT service subscribes to these topics and receives the data in real-time.
* The cloud platform processes and stores the data, allowing users to remotely monitor and control their smart home through a web or mobile application.

The following application protocols can be used at each layer:

## Physical Layer:

### Zigbee (Local Communication):

Zigbee operates on the IEEE 802.15.4 standard, providing a low-power, short-range wireless communication protocol. It is suitable for connecting devices within a smart home, forming a mesh network.

### Wi-Fi or Ethernet (Internet Connectivity):

For connecting the Zigbee coordinator to the internet, standard technologies like Wi-Fi or Ethernet can be employed. This allows the coordinator to communicate with cloud services over the internet.

## Internet Layer:

### Internet Protocol (IP):

IP operates at the network layer and is used for routing data between devices on the internet. The Zigbee coordinator, when communicating with the cloud, will rely on IP for routing data packets.

## Application Layer:

### MQTT (Message Queuing Telemetry Transport):

MQTT operates at the application layer and is a lightweight and efficient messaging protocol used for communication between the Zigbee coordinator and the cloud.

### HTTPS (Hypertext Transfer Protocol Secure):

HTTPS also operates at the application layer and is used for secure communication between the Zigbee coordinator and the cloud platform. It is particularly relevant if the cloud platform exposes RESTful APIs for data transmission

# Tools and technologies:

The following tools and technologies will be used for the project:

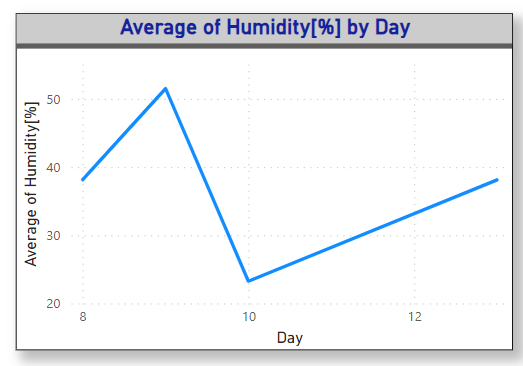
1. TinkerCad for designing the circuit
2. PowerBi for displaying analytics of data
3. Google Colab for training the machine learning model
4. ZigBee Technology for physical layer
5. Wi-Fi for internet layer
6. MQTT for application layer

# Data Visualization/Analysis and system interface:

## Dashboard of System:

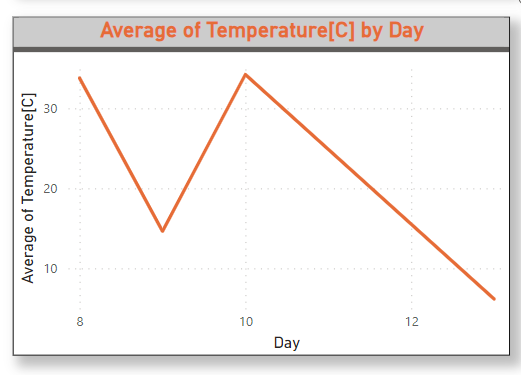
### Humidity

The following is the line graph of average humidity vs Day from the data of smoke detection system. It shows that the maximum average humidity over the span of 5 days was recorded on the 9th of June (~52%) while the minimum average Humidity was recorded on 10Th June (~23%).



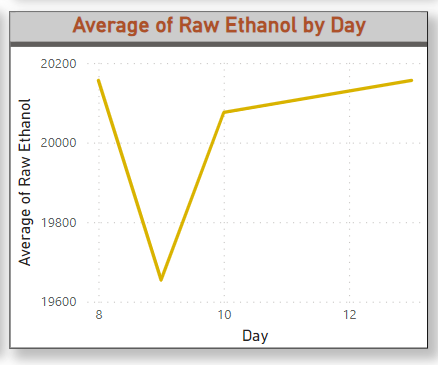
### Temperature

The following is the line graph of average temperature vs Day from the data of smoke detection system. It shows that the maximum average temperature over the span of 5 days was recorded on the 10th of June (~34 C) while the minimum average temperature was recorded on 9Th June (~15C).



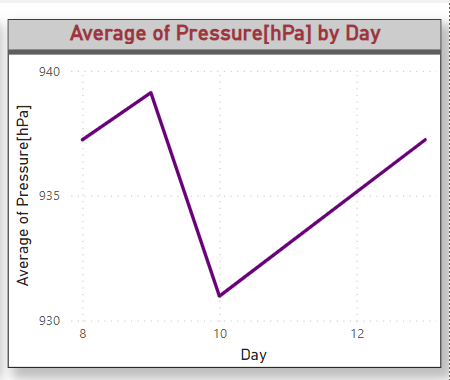
### Raw Ethanol

The following is the line graph of average Raw ethanol found in air vs Day from the data of smoke detection system. It shows that the maximum average Raw ethanol over the span of 5 days was recorded on the 8th of June (~20157 units) while the minimum average Raw ethanol was recorded on 9Th June (~19654 units).



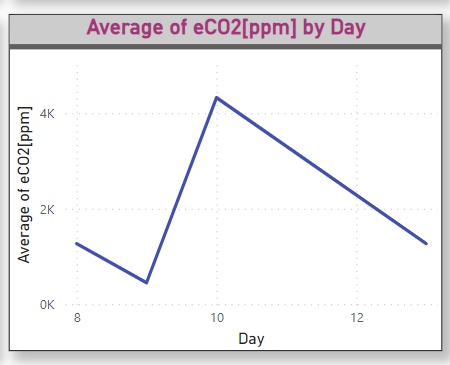
### Pressure:

The following is the line graph of average Pressure vs Day from the data of smoke detection system. It shows that the maximum average Raw ethanol over the span of 5 days was recorded on the 9th of June (~939 hPa) while the minimum average Raw ethanol was recorded on 10Th June (~930 hPa).



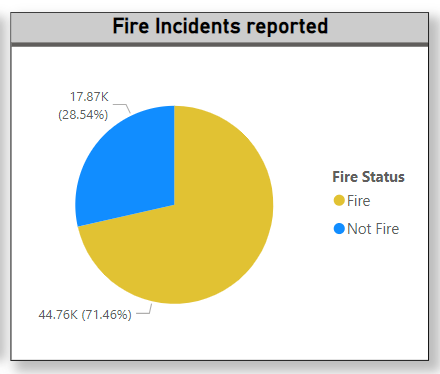
### Average CO2:

The following is the line graph of average equivalent carbon dioxide vs Day from the data of smoke detection system. It shows that the maximum average equivalent carbon dioxide over the span of 5 days was recorded on the 10th of June (~4328 ppm) while the minimum average equivalent carbon dioxide was recorded on 9Th June (~448 pm).



### Fire Incidents:

Following is the pie chart that shows the percentage of total number fire and non-fire incidents classified by the smoke detection system over the span of 5 days. It shows that total 28.54% of incidents were marked as not fire while rest 71.46% incidents were reported as fire incidents.



## Predictive Analysis:

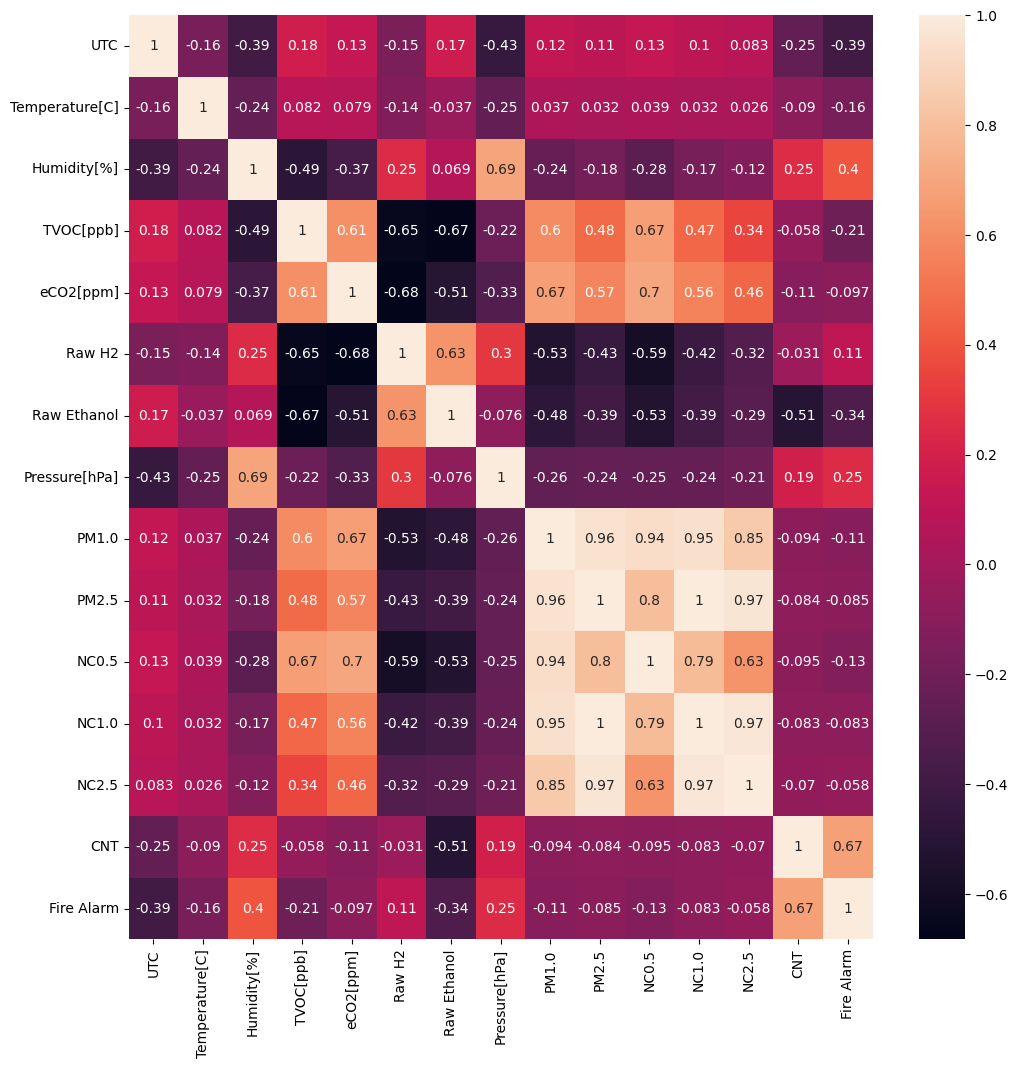
The predictive analysis of the data has been performed with the help of a machine learning algorithm. We have used the smoke detection dataset to perform our predictive analysis using XG Boost Model. Our trained model is capable of predicting whether smoke has been detected or not based on different feature values. Following features are present in this dataset:

1. Temperature
2. Humidity
3. TVOC[ppb]
4. eCO2[ppm]
5. Raw H2
6. Raw Ethanol
7. Pressure[hPa]
8. PM1.0
9. PM2.5
10. NC0.5
11. NC1.0
12. NC2.5
13. CNT
14. Fire Alarm

The Fire Alarm column represents whether fire alarm was turned on based on the detection of smoke.

### Heat Map:

Below is the heatmap that illustrates the correlation between different column values:



### Data Pre-processing:

The data was pre-processed by removing the outlier values from each column.

### Data Splitting:

The data was split in 9:1 indicating that 90% was used as training data and the rest was used as testing data.

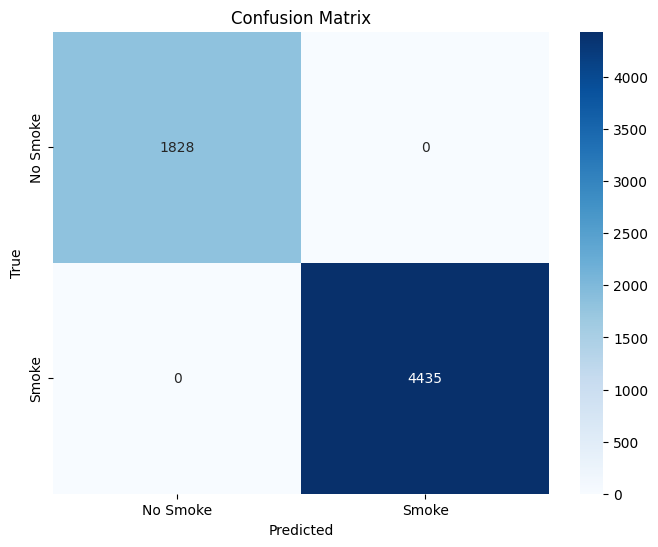
### Model Training and Hyper parameter tuning:

After the model was trained using XGBoost, hyper parameter training was performed using RandomizedSearchCV on the following hyper parameters:

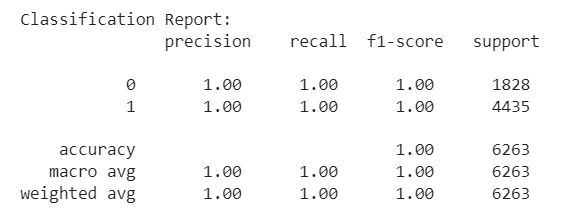
1. learning\_rate
2. n\_estimators
3. num\_leaves
4. max\_depth

### Model Testing and Evaluation:

After hyper parameter tuning the model was evaluated on the test dataset and following confusion matrix was obtained:



The classification report is as follows:



# Expected Impact:

The successful implementation of this project will result in a next-generation smart home system that offers enhanced interoperability, user experience, security, and energy efficiency. This solution has the potential to revolutionize the way residents interact with their living spaces, promoting the widespread adoption of smart home technology. The following benefits will be offered to users and society:

## Automation and Convenience:

Smart homes offer automated control of various devices and systems, providing convenience in daily tasks. This automation enables the scheduling and coordination of activities, enhancing overall efficiency.

## Remote Access and Monitoring:

The users can remotely monitor and control smart home devices using smartphones or other connected devices. This allows for real-time surveillance, adjustment of settings, and increased security.

## Energy Efficiency:

Smart home technology optimizes energy consumption through intelligent systems. Features like smart thermostats and energy-efficient appliances contribute to reduced energy costs.

## Cost Savings:

Energy-efficient devices and automated systems contribute to cost savings over time. Monitoring and optimizing resource usage led to reduced utility bills.

## Improved Quality of Life:

Smart home technologies contribute to a more comfortable and enjoyable living environment. Accessibility features can benefit individuals with mobility challenges or disabilities.

## Environmental Impact:

Energy-efficient practices and devices contribute to a reduced carbon footprint. Smart home technologies promote sustainability and environmental consciousness.

**GitHub Link:** [**https://github.com/Mydah-Nasir/Smart-Home-IoT-Project**](https://github.com/Mydah-Nasir/Smart-Home-IoT-Project)