IoT based LPG Gas Utility system

Mitrajeet Golsangi*, Divija Godse†, Vivek Ghuge‡,
Vishwajeet Haralkar§, Adityaraj Honraopatil¶ and Dr. Vijay Gaikwad‖
dept. of Computer Science
Vishwakarma Institute of Technology
Pune, India
Email : *mitrajeet.golsangi20@vit.edu, †divija.godse20@vit.edu, ‡vivek.ghuge20@vit.edu,
§vishwajeet.haralkar20@vit.edu, ¶adityaraj.honraopatil20@vit.edu, ‖vijay.gaikwad@vit.edu,

***Abstract*—LPG is a powerful and efficient fuel that is primarily used in private cooking areas. LPG is highly combustible and can cause a fire even if it is far away from the source of the leak. The majority of fires are caused by a faulty rubber tube or when the regulator is not turned off. The proposed system is built around a microcontroller that includes gas sensors, GSM, a display, and a buzzer. The gas leak will be detected by the sensor, which will then send the information to the microcontroller. Based on that microcontroller makes a decision and displays a warning message. Another issue is the user not being able to tell the level of LP gas in the cylinder causing problems at the 11th hour. The proposed solution has brought a feasible solution to the same causing the users to monitor the Gas level from their mobile phones.**

***Index Terms*—LPG, IoT, LPG hazards, LPG monitoring , IoT embedded system**

I. Introduction

Our aim is to do a complete evaluation of the LPG gas systems which have been rooted in nearly 30 crore people in India. [1] It is a tedious process to check the level of gas cylinder at home without any special equipment and book the new one in anticipation. We aim to improve the safety, usability and effectiveness of the LPG gas cylinder.

LPG gas safety is a very important aspect of the complete system. Gas leakage leads to various accidents resulting in both financial loss as well as human injuries. [2] The risk of firing, explosion, suffocation all are based on their physical properties such as flammability, toxicity etc. According to the Ministry of Petroleum and Natural Gas, Government of India over 835 deaths have been recorded in 2018-19 in India alone. [3] Inspections by oil companies found that many LPG consumers are unaware of safety checks of gas cylinders. There is a need for a system to detect and also prevent leakage of LPG.

II. Literature Survey

In these papers, the authors have mentioned the use of an MQ-6 gas sensor, for the sole purpose of leakage detection. [4] The MQ-6 sensor senses LPG leakage in any case and sends alert signals to the microcontroller when it reaches a dangerous level. Calculation: power of sensitivity

$P_s = V_c^2 \times \frac{R_s}{(R_s+R_L)^2}$

Resistance of sensor (R_s):

$R_s = \frac{V_c}{V \times R_L - 1} \times R_L$ [5] [6]

From the microcontroller, the alert signals are then sent to the users via GSM. [7] Thus the users would be made aware of any kind of LPG leakage. Weight sensor is used for the purpose of monitoring and detecting the amount of gas present in the LPG cylinder. [5] Usually, the permissible net weight for LPG in the domestic cylinder is 14kg. The weight of an empty cylinder is approximately 15.3kg. The total gross weight of the cylinder can be rounded off to 29.5kg. Thus when this figure changes the weight sensor senses the smallest changes which are further displayed on the LCD. When it reaches 0.5kg the system sends an alert to the user device. [6]

To address the above mentioned issues our project uses the MQ-135 Gas sensor and alerts the user if the regular gas composition of the surrounding atmosphere changes. Our project would also use a 4.08mm force sensor in combination with the Gas Sensor for calculating the weight of the cylinder and notifying the user if a certain threshold is crossed. We intend to combine these two sensors and create a system that would prevent gas leakage and also serve as an alert sending device to prevent last minute shortage.

Many solutions have been in the industry for years like a solution created by students of KIIT [3]. The proposed solution aims to improve the safety, usability and effectiveness of the LPG gas cylinder usage. The advantages of the system are: As the Weight sensor output is converted to digital in the reference system there is a loss of accurate data on actually how much level of gas is remaining in the cylinder. The proposed systems avoids this problem By using the Node MCU which has a built in ESP8266 WiFi module the cost of the project has been reduced By giving the user an application to monitor their gas systems the accessibility and control over the system have been increased significantly

III. Methodology

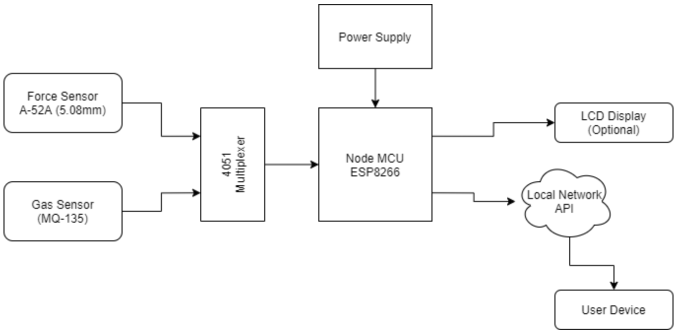

Fig.Block Diagram

The oversimplified block diagram (fig. III) allows the users to see the basic workflow of the project. The 4051 multiplexer toggles the analog input on the Node MCU (pin A0) which is controlled by three pins D1, D2, D3 on Node MCU. The maximum amount of time is given for the gas leakage detection whereas a finite amount of time (not sure exactly how much yet) is given to the Measurement of the level of gas in the cylinder. All the statistics is sent to a host on the local network and then is fetched from the user's application to give him/her the most accurate data in real time.

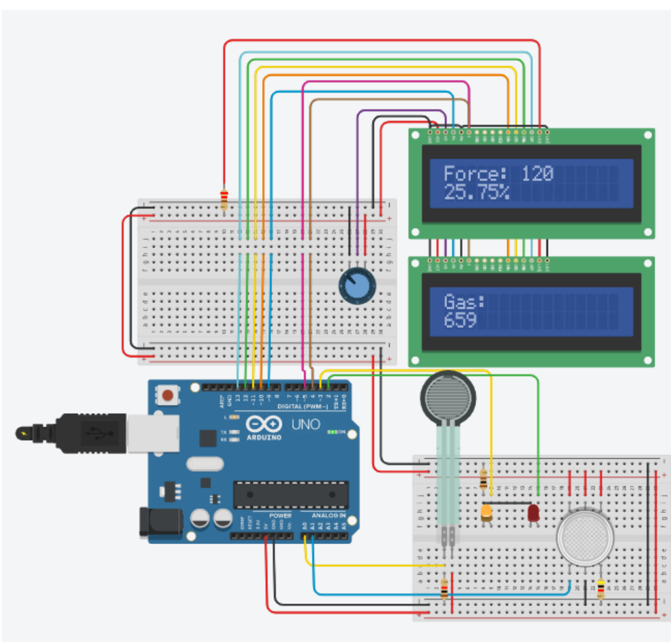

Fig.Simulation

figure III shows the simulation of the project that has been performed on Autodesk TinkerCAD®. Here 2 LCD monitors are used as wireless transmission cannot be simulated. As seen the readings of the force and gas Sensor are shown on the user device (LCD Display in this case) and as the gas has crossed the soft threshold of 470 PPM. The Yellow LED has been lit up to warn the user of some abnormal Gas concentrations in the surrounding atmosphere.

A. Hardware

1) Part Required:

- Node MCU 0.9 (ESP8266 Module)
- MQ-135 Gas Sensor
- A-52A 5.08mm Gas Sensor
- 4051N Multiplexer IC
- Resistors
- LED
- 15 pin female socket

2) PCB Creation: After the gathering of required components the team created a schematic for the required circuit.

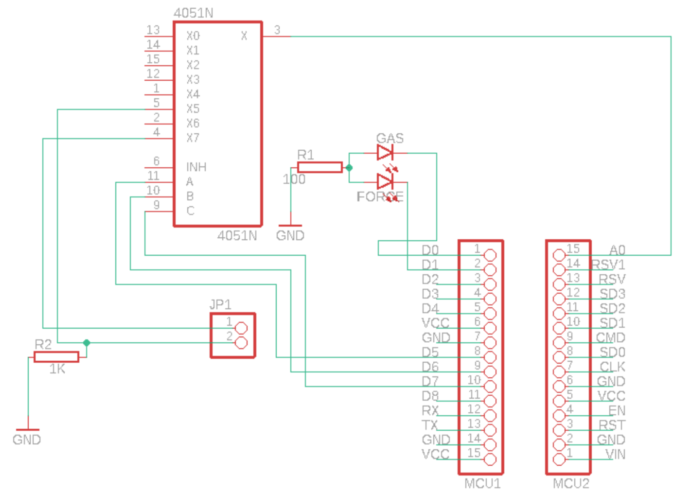

Fig.Schematics

After completion of the schematic and checking the connections the PCB board layout was designed and finalized as shown below

B. Software

1) Used Softwares:

- VS Code IDE
- Platform IO IDE
- Eagle

IV. Application Creation

A. Introduction

The project uses a mobile application as the means of connecting the device data and the end user. All the data collected from the sensors is uploaded to the local network in the form of a JavaScript Object Notation (JSON) format. This is useful as the mobile application then collects the data and directly processes it into useful information and displays it to the users. This exact process is explained in detail in the paper below

B. Structure

The application is segregated into 2 main parts

- Sensor readings
- Network Configuration

Sensor Readings : This section displays the collected sensor data to the user. The sensor data is displayed in the format of percentage and a graph which is refreshed at 200 ms for getting the new data.

Network Configuration : This is to ensure that the ESP8266 WiFi module is connected to the application and correctly configured.

C. State Management

In any application where the data needs to be shared from one screen to another a concept of state management is used. [8] This is needed in order to share data between the various screens or pages of the application. Flutter has several state management systems including redux framework and BLoC architecture [9]. In this application BLoC architecture is used.

V. BLoC Architecture ?

BLoC abbreviates to Business Logic of Components. This method is recommended in the official Flutter documentation by the Flutter's Development Team. The BLoC state management system is similar to the M-V-VM (Model View ViewModel) state management system, but uses various streams in order to achieve the same effect. The concept of BLoC is to have a corresponding effect for each state inside the application, this is better explained in the figure below.

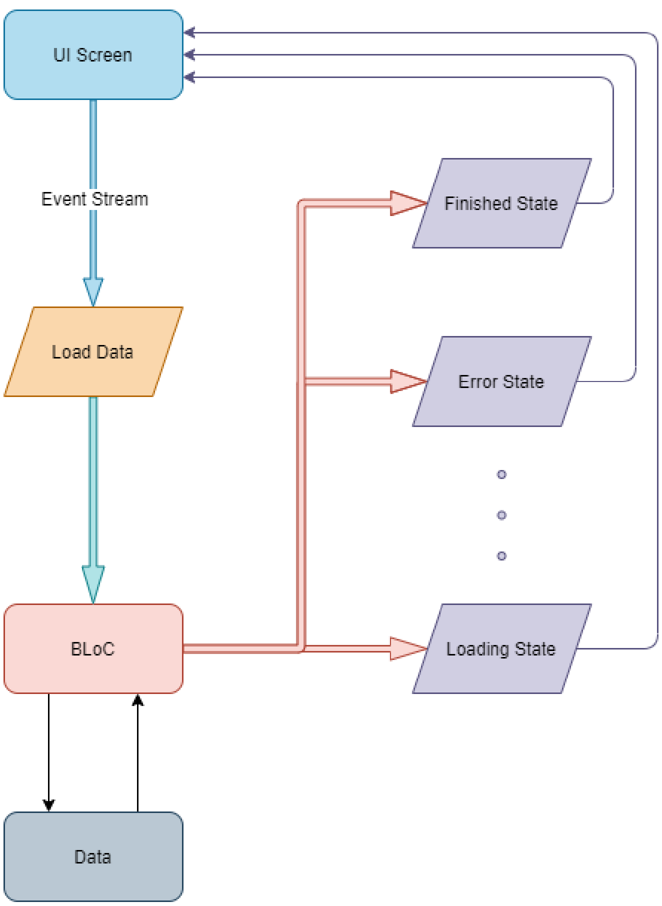

Fig.BLoC Workflow

Like displayed in the above figure the UI Screen loads an event on the event stream which is then carried on to the BLoC. According to this event package the BLoC interacts with the repositories and emits a stream of states on different stages of the interaction. This state is then again carried out to the UI Screen and required changes are made to indicate to the user some processes are going on in the backend. This methodology has several advantages.

A. Advantages of BLoC State Management

1) State of application can be known in any point of time
2) Testing of the application is easily possible
3) Data driven decisions can be made by recording the user behaviour
4) Components can be reused making code efficient

The application uses two cubits in order to manage its state. The main difference between a cubit and a bloc is that the cubits don't use event streams but send the states on a stream. Meaning only the response of the cubit is a stream and not the input.

VI. Application Workflow

The applications workflow is explained in accordance to the figure:

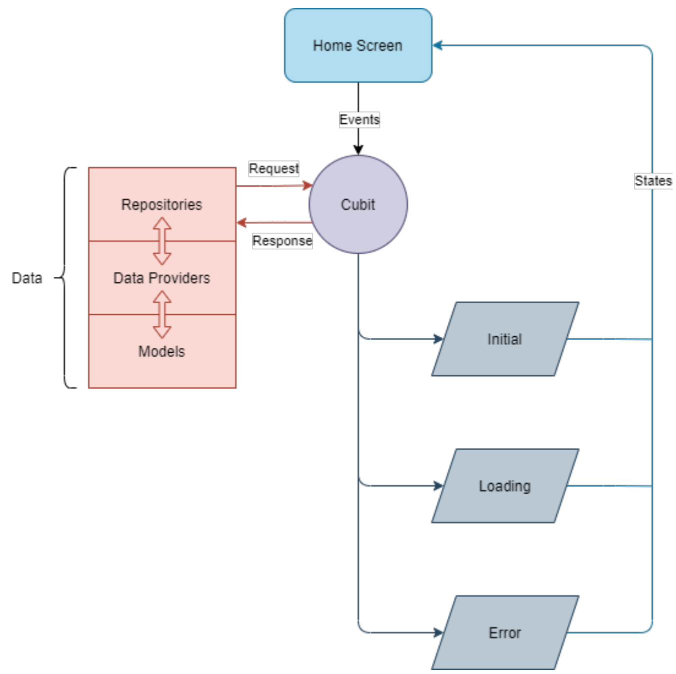

Fig.BLoC Workflow

The first cubit is to get the sensor readings from the API endpoint generated on the network IP generated by the ESP8266 WiFi module. This cubit emits 3 states. The initial state sets the force sensor and the gas sensor readings to zero and continues to send requests to the API endpoint. The next states are determined by the response emitted by the repositories. If the response is an error response code i.e. anything other than the 200 HTTP response code the cubit emits the Error state and notifies the user some error has occurred, else the force sensor and gas sensor values are updated giving the user the correct values of the same. The new requests are made in an interval of 200 milliseconds, thus the user gets the data in realtime. The repositories emit two kinds of data, percentage and actual values of the sensors, the percentage is displayed directly to the user and the actual values are used to map out a graph of the sensor readings, this gives a better understanding of the situation to the user.

A similar approach can be observed in the network configuration cubit where the cubit emits three states the same as the former. These emitted states are then caught by the UI components and then those components are rerendered to give the user the necessary information.

REFERENCES

[1] A. Ahsan, M. Z. lslam, and R. Siddiqua, "An iot based interactive lpg cylinder monitoring system with sensor node based safety protocol for developing countries," *IEEE Region 10 Symposium*, 2020.

[2] A. Macker, A. K. Shukla, S. Dey, and J. Agarwal, "Arduino based lpg gas monitoring and automatic cylinder booking with alert system," *IEEE*, 2018.

[3] A. K. Srivastava, S. Thakur, A. Kumar, and A. Raj, "Iot based lpg cylinder monitoring system," *IEEE International Symposium on Smart Electronic Systems (iSES)*, 2019.

[4] A. Siddika and I. Hossain, "Lpg gas leakage monitoring and alert system using arduino," *International Journal of Science and Research (IJSR)*, 2018.

[5] T.Soundarya and J. Anchitaalagammai, "Control and monitoring system for lpg detection," *nternational Journal of Innovative Research in Science, Engineering and Technology*, vol. 3, 2014.

[6] R. K. Kodali, T. D. B, and S. C. Rajanarayanan, "Iot based automatic lpg gas booking and leakage detection system," *IEEE*, 2018.

[7] J. Chaudhary and A. Mishra, "Detection of gas leakage and automatic alert system using arduino," *SSRN Electronic Journal*, 2019.

[8] L. H. Hoang, "State management analyses of the flutter application," Ph.D. dissertation, Metriploitia University of Applied Sciences, 2019.

[9] M. Szczepanik and M. Kedziora, "State management and software architecture approaches in cross-platform flutter applications," *ENASE*, 2020.