iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** XI **Month of publication:** November 2022

DOI: https://doi.org/10.22214/ijraset.2022.47615

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue XI Nov 2022- Available at www.ijraset.com

IoT-based Smart Bag Using Arduino

Shazia Talib[1], Tanvi Save[2], Arwa Ujjainwala[3], Arjun Jaiswal[4]
[1, 2, 3, 4]*Department of Information Technology, Dwarkadas J. Sanghvi College of Engineering, Mumbai, India*

Abstract: ***According to statistics, about $1.25 billion worth of baggage was potentially lost or stolen in 2018. About 77% of mishandled bags were delayed, 18% were damaged or pilfered, and 5% were completely lost or stolen. Baggage being lost, mishandled, and stolen is a widespread problem many of us face. It becomes especially hard for solo travelers and the impaired to lug their luggage around every day. Traveling is such an important aspect of everybody's lives, yet with the world undergoing a technological revolution, our bags are still not smart. Hence, to help make journeys comfortable and safe, the implemented system proposes the concept of an intelligent suitcase. This project comes under the domain of IoT and is developed using Arduino Uno to make it cost-effective. The features of this system include motion detection, obstacle avoidance, and location tracking. The system has also been tested using multiple test cases.***
Keywords: IoT, Internet of Things, Smart Bag, GPS, GSM SIM900A, L298N motor driver, Bluetooth HC05, IR sensor, Automated motion, Obstacle detection, Location tracking.

I. INTRODUCTION

In our everyday life, we travel to multiple locations for work and vacations. Generally, people use simple luggage bags or suitcases for traveling purposes, but in today's tech-savvy world, thieves have also gotten tech-savvy! Missing pieces of baggage, damage to customers' belongings, and carrying around huge loads of luggage everywhere are the common problems faced by passengers around the world. This problem needs an efficient and smart solution. The proposed concept of a smart bag can be a potential solution. Smart Bag is based on the simple fact that people always suffer from all these issues when they go traveling on a trip, and there will always be some or the other circumstances when people forget to bring some important items, maybe changes of clothes, charging cables, and even their passports which is extremely crucial for traveling. However, no one will forget to bring their phones on the trip with the current world scenario. Thus, these phones can be used to take care of baggage and keep it safe. The proposed system uses Arduino Uno which is cost-effective and simple to work with. The system has multiple functionalities which can help users to manage their trips and let their luggage automatically follow them. The luggage can be tracked remotely using GPS. It is an IoT application integrated into luggage. A motor and a motion detection sensor are used to drive the suitcase.
The proposed model can be built on a large scale with stronger and more efficient machines. This project is a prototype product for IoT-based Intelligent bags. This paper lays down the scope and project resources necessary for the implementation of the project. System architecture and Implementation details are thoroughly mentioned in the subsequent sections.

II. LITERATURE SURVEY

A. Luggage Tracking System Using IOT

This project consists of a GPS module implemented with Arduino and an alarm sensor. We can always track the location of the bag. The alarm sensor in the system is integrated to prevent theft or losing the bag. The users get notifications as the bag moves out of a particular range like 10m, 20m, and 30m. The alarm gets triggered if the bag goes out of a certain range.[2]

B. Smart Luggage Tracker

This system implements a luggage tracking & handling system using RFID.
Radio Frequency Identification – or RFID does two of the following things:
1) Automatically identify an object
2) Capture data about that object that has been stored in a small microchip tag and attached to the object.
The RFID tag has a built-in antenna that communicates to a scanning device that reads the data remotely. RFID uses radio frequency waves to transfer data and hence they are wireless. This RFID algorithm is used for generating tags that are attached to printed luggage labels. The details of passengers and airlines are stored in it. A prototype at the two locations of check-in and check-out is to be developed. RFID readers at the check-out have step-by-step tracking of luggage. Location is also tracked.

©IJRASET: All Rights are Reserved

Each boarding passenger has a unique RFID code which has to be entered on the website to know the exact location and status of their luggage. Details include the exact time of arrival of luggage, location, and net weight before and after loading. This information lets the passenger take necessary action if the luggage has been misplaced, stolen, or tampered with. [1]

C. Smart Luggage Tracking And Alert System Using Arduino

This system is a lightweight device with advanced security technology. The android software provides the location status of luggage. It also has a Fingerprint Sensor to keep their belongings safe and secure. In addition, it also stores unauthorized and authorized location details in the cloud.[3]

Hence to summarize, the existing systems use Arduino or Raspberry Pi as a microcontroller. They usually have a GSM-GPS integrated module to track and send the location to the user. They might have a wi-fi module attached to send messages over the wi-fi. They also have protection against theft and security such as an alarm sensor that gets triggered as soon as the bag is stolen. They have a fingerprint module integrated into the system to protect the bags from being opened by an unauthorized user. There are different reasons for not including certain technologies in the proposed system. RFID might not be a very reliable technology as the materials like metal and liquid can very well impact signals. It is difficult for an RFID reader to read the information in case the tags are installed in liquid or metal products. The problem here is that liquid and metal surfaces tend to reflect radio waves, which makes the tags unreadable. It is also not very accurate. Implementation of RFID is very time-consuming and not efficient. RFID tags are also larger than barcode scanners. Interference has been observed to take place in RFID systems when devices such as forklifts and walkie-talkies are in the vicinity. The presence of mobile phone towers has also been found to interfere with these radio waves. RFID signal frequencies across the world are non-standardized. RFID tags are also very expensive. Raspberry Pi is not very efficient either. It does not replace the computer, and the processor is not as fast. It is time-consuming to download and install the software and that makes it unable to do any complex multitasking. Raspberry Pi is also not compatible with other operating systems such as Windows. Raspberry Pi is good for developing software applications using Python, while Arduino is good for interfacing Sensors and controlling LEDs or Motors. If the system is a single-purpose system then Arduino is better and way more cost-effective than Raspberry Pi.

The alarm sensor's role in existing systems is also faulty. The alarm sensor getting triggered after the bag goes out of a certain range is ambiguous. Technically, it should get triggered on its own since the owner might be in absolute distress with their bag being stolen and have no sense to trigger the alarm, but, if it does get triggered on its own, then how can one make sure that the bag is actually being stolen or just getting away from the radius simply because of something else? The alarm sensor also causes a lot of noise pollution and can be harmful to the elderly and small children.

Hence the proposed system was built without using these technologies.

III. PROPOSED METHODOLOGY

Our proposed methodology is implemented by integrating the IR sensors with the Bluetooth module and the motor drivers as well.

We have 2 functionalities in the system:

1) Tracking of the suitcases always (even if the user has no wi-fi in his phone)

2) Motion of the suitcase by giving it commands through your phone.

The tracking is done by obtaining the location coordinates of the bag. The proposed system needs the user to have Bluetooth on their phone. It does not need wi-fi at all. The system also expects the user to have a network in their phone to send/receive messages at all times. The motion control also works only until a certain range due to the Bluetooth constraint.

Features of the Proposed System

a) Motion Detection

- Motion Detection is to be done with the help of an IR motion detector.
- Should follow a user manually by using an app.
- The user should have Bluetooth connected to the app.
- The Carrier follows the user based on the motor which is connected to the app through Bluetooth.

b) Speed Control

- By using the motor driver, it will control the speed and also give a high voltage to the motor.

c) Obstacle Avoiding

- Has to be done with the help of the IR sensors.
- The carrier detects a value of closeness of an object with the help of IR sensors.
- When it detects any hindrance, it will stop.

d) Tracking the Location

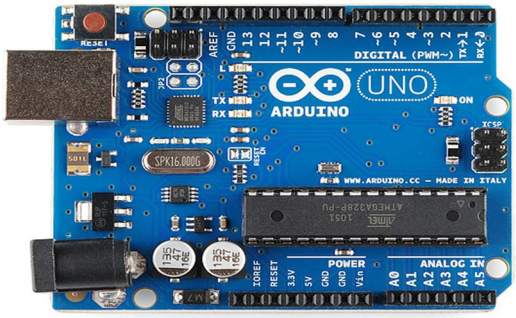
- The carrier is built with smart features that allow the user to get the location of the luggage wirelessly on the mobile phone.

The whole system integration is controlled through an application installed in Android and hence controlling the device will be very easy for the user.

IV. PROJECT REQUIREMENTS

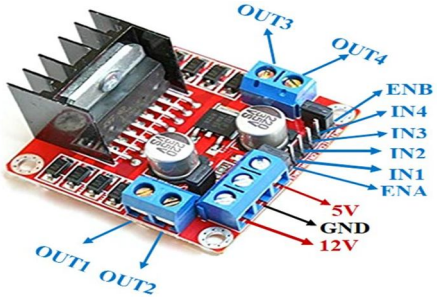
A. Hardware Requirements

1) *Arduino:* It is a microcontroller board that is based on the ATmega328. Arduino is the heart of the proposed system. USB cable can be used to connect Arduino to the laptop. Arduino is programmed using the Arduino IDE (Language: C). Arduino Uno contains a set of analog and digital I/O data pins which are used to connect all the other electronic components. It consists of 14 digital I/O pins and 6 analog I/O pins. It has a USB port from which we can connect it to a laptop and upload the program. This port is also used to power the board. Along with the USB port, it also has a DC power jack. An external battery or adaptor of 9V or 12V can also be used to power Arduino boards. It can take in voltages between 7 and 20 V. [8]



Arduino

2) *Motor Driver(L298N):* The wheels in the system are powered by motors that are interfaced with the help of a motor driver. Motors require a high measure of current while the controller circuit takes low current signals. So, the motor driver takes the low input from the Arduino and converts it into high voltage so that our motor can work accordingly. The role of motor drivers is to take a low-current control signal and transform it into a higher-current control signal that can drive the motor. The motor driver requires an external battery. Two motors can be connected to one motor driver. One motor goes into the terminal labelled OUT1 and OUT2 and the second goes to the terminal labelled OUT3 and OUT4. Using a motor driver, we can control the speed of the motor and change its direction as well. The row of pins on the bottom right of the L298N control the speed and direction of the motors. IN1 and IN2 control the direction of the motor connected to OUT1 and OUT2. IN3 and IN4 control the direction of the motor connected to OUT3 and OUT4. The L298N can be powered with a battery of up to 12V via the pin labelled "12V". [15]



Motor Driver

L298N Module Pinout Configuration

Pin Name	Description
IN1 & IN2	It is used to control the spinning direction of motor A
IN3 & IN4	It is used to control the spinning direction of motor B
ENA	It enables PWM signal for motor A
ENB	It enables PWM signal for motor B
OUT1 & OUT2	These pins are the output pins for motor A
OUT3 & OUT4	These pins are the output pins for motor B
12V	In this pin external battery is connected like 9V
5V	This pin is used for switching the logic circuitry inside L298N IC
GND	Ground pin which connected to the ground of Arduino

3) *Motor:* As discussed above in the motor driver, the motor is used to move the bag. Two motors need to be used so that the motion can be started and stopped as well as to enable the direction that can be changed using the Bluetooth app.



Motor with wheels

4) *IR Sensor:* It is a sensor that is used to determine certain characteristics of its surroundings using infrared radiation. An IR sensor can be used to detect whether an obstacle is present in front of it or not, i.e. the IR sensor is used to detect the proximity of the obstacle. It consumes low power and is cost-efficient. Working with this sensor is very simple. It consists of two main components: the first is an IR transmitter and the second is an IR receiver. In the transmitter, IR (infrared) led is used and in the receiver, a photodiode is used to receive an infrared signal and after some signal processing and conditioning, an output is generated.

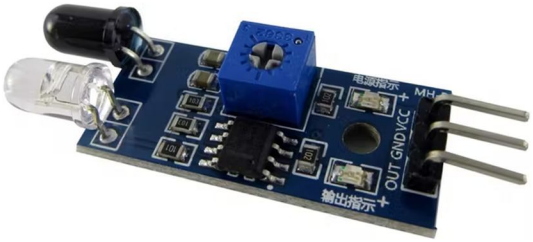


Figure 3.5.1.d IR sensor

Table 2 - IR sensor Pinout

Pin Name	Description
IR Emitter LED	This LED is used to emit infrared light
IR Receiver	A photodiode with resistance depending on the amount of IR light falling on it
Vcc Pin	3–5 V input voltage
Gnd Pin	Connected to the ground
Out Pin	This pin transmits the output signal from the sensor to Arduino.
Obstacle LED	If the light falls on the IR Receiver it gets turned on.
Power LED	When it is connected to the voltage source the LED will turn on
IR Emitter LED	It emits infrared light

5) *Wires:* Wires are of different types. There are male-to-male, female-to-female, and male-to-female wires. They are used to connect different sensors with our Arduino Uno.



Figure 3.5.1.e Jumper Wires

6) *GPS:* Global positioning system (GPS) is a module utilizing which the location of a device can be ascertained. GPS navigation is based on a satellite system. It gives latitude and longitude by which we can track and be informed of the position of the module. GPS works by receiving unique signal and orbital parameters transmitted by the satellite that is then decoded by the GPS device to compute the precise location of the satellite and thereby calculate the exact location of the user. The GPS receiver bases its distance calculation on the amount of time it takes to receive a transmitted signal from each satellite. With distance measurements from a few more satellites, the receiver can triangulate a user's position and display it. A minimum of three satellites are required to lock onto a 2-D position (latitude and longitude) and track movement. As soon as the GPS sensor finds the satellites, a blue light starts blinking continuously. This is an indication that GPS is working correctly. [16]



Figure 3.5.1.f GPS Module

GPS Pinout

Pin Name	Description
GND	Ground Pin
TxD	TxD (Transmitter) pin is used for serial communication.
RxD	RxD (Receiver) pin is used for serial communication.
VCC	It supplies the power to the module. It can be directly connected to the Arduino 5V pin.

7) *GSM:* To send and receive messages, a GSM - Global System for Mobile Communication SIM card is put into the mobile handset. The number of the GSM SIM card is stored in the system. Data is sent from the control unit to the base unit through GSM. It is used for long-distance data transmission using GSM technology in areas where there is no access to the internet. This makes it handy in projects that require data transfer over long distances. This module can be used to create a variety of applications, such as call or text message-based triggers. The GSM module communicates wirelessly with another device using GSM and GPS technology. It connects to the internet via a 2G network and supports Quad-band (EGSM 900, GSM 850, DCS 1800, PCS1900). The versatility of this module is very high due to its ability to read and send messages without any hassle. It is very easy to configure using AT commands. The module can also be used in low signal areas because of the presence of an external antenna. [17]

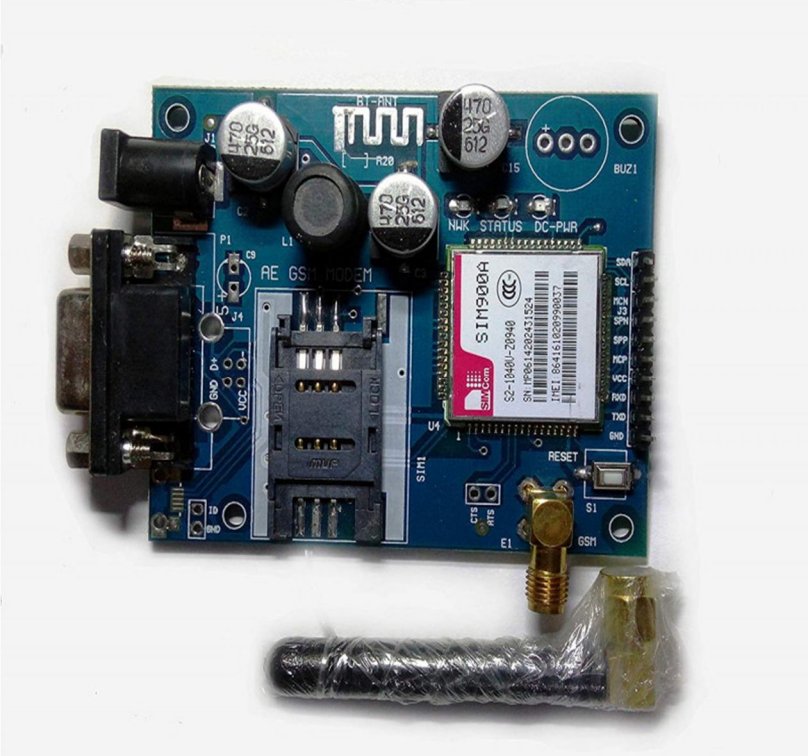


Figure 3.5.1.g GSM Module

8) *Bluetooth:* It establishes a wireless connection between devices. In this system, the Bluetooth module helps to connect the app to Arduino. The Bluetooth-connected app in turn enables the movement of the wheels. It has a range of less than 100m which further depends on the transmitter, receiver, atmosphere, geographic and urban conditions. It uses serial communication to communicate with devices. It communicates with a microcontroller using a serial port (USART). The specific model used in this system is the HC-05 Bluetooth module. This module can be used in a master or slave configuration.

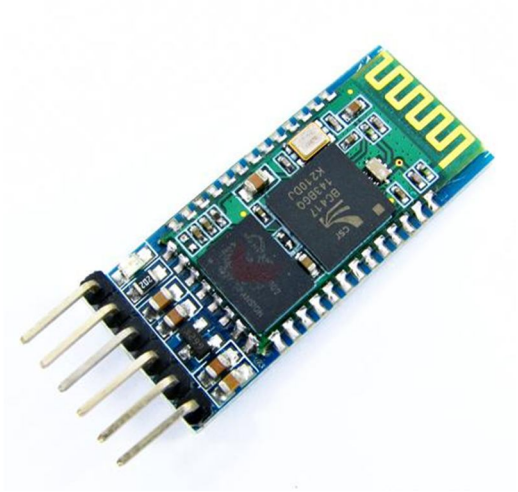


Figure 3.5.1.h.1 Bluetooth Module



Fig 33.5.1.h.2 Pin Description of Bluetooth Module

Bluetooth Pinout

Pin Name	Description
VCC	Connect with 5V or 3.3V
GND	Ground Pin
TXD	Transmit Serial Data (data received wirelessly by Bluetooth module is transmitted out serially via TXD pin)
RXD	Receive data serially (received data will be transmitted wirelessly by Bluetooth module).

B. Software Requirements

1) Arduino IDE: Arduino IDE (Integrated Development Environment) is a software platform used to write programs to operate Arduino or any other ATmega controller. The user can write code to program the Arduino in C language but the backend is developed using JAVA. It is an interface that acts like a bridge to upload code to the Arduino and hardware and communicate with them. The following functions have been used from various Arduino libraries in the code for the system:

a) Void Setup (): This space is used to initialize any variables required for the program and include libraries of various sensors. This function is used to set up an Arduino before adding functionality to the circuit.

b) Pinmode: This function is used to declare the pins of Arduino as input or output.

c) Serial.Begin: This function initiates the Arduino's communication with other sensors or devices. It allows a specific baud rate to be set for the purpose of communication.

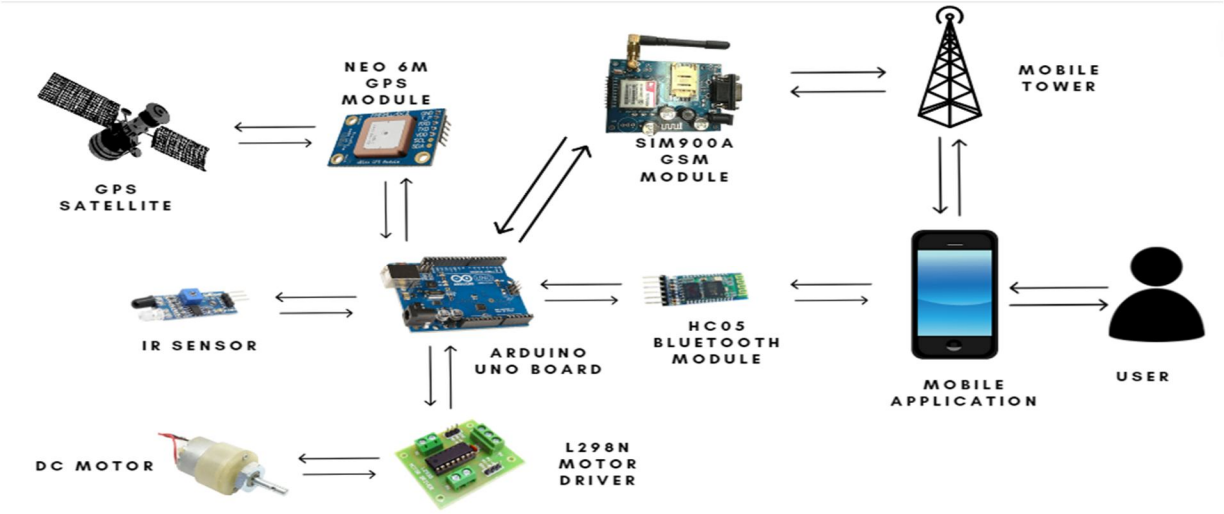
d) Void Loop (): Any code written in this section will loop again and again unless Arduino is interrupted either by issuing an interrupt or disconnecting the USB cable from the port.

e) Digitalwrite: This function is used to make a specific pin on Arduino logically HIGH or LOW.

f) Digitalread: This function has two uses: reading digital data from sensors and controlling something using a switch.

2) Android Mobile Application: This project requires the user to install an android application on their smartphones.

V. IMPLEMENTATION

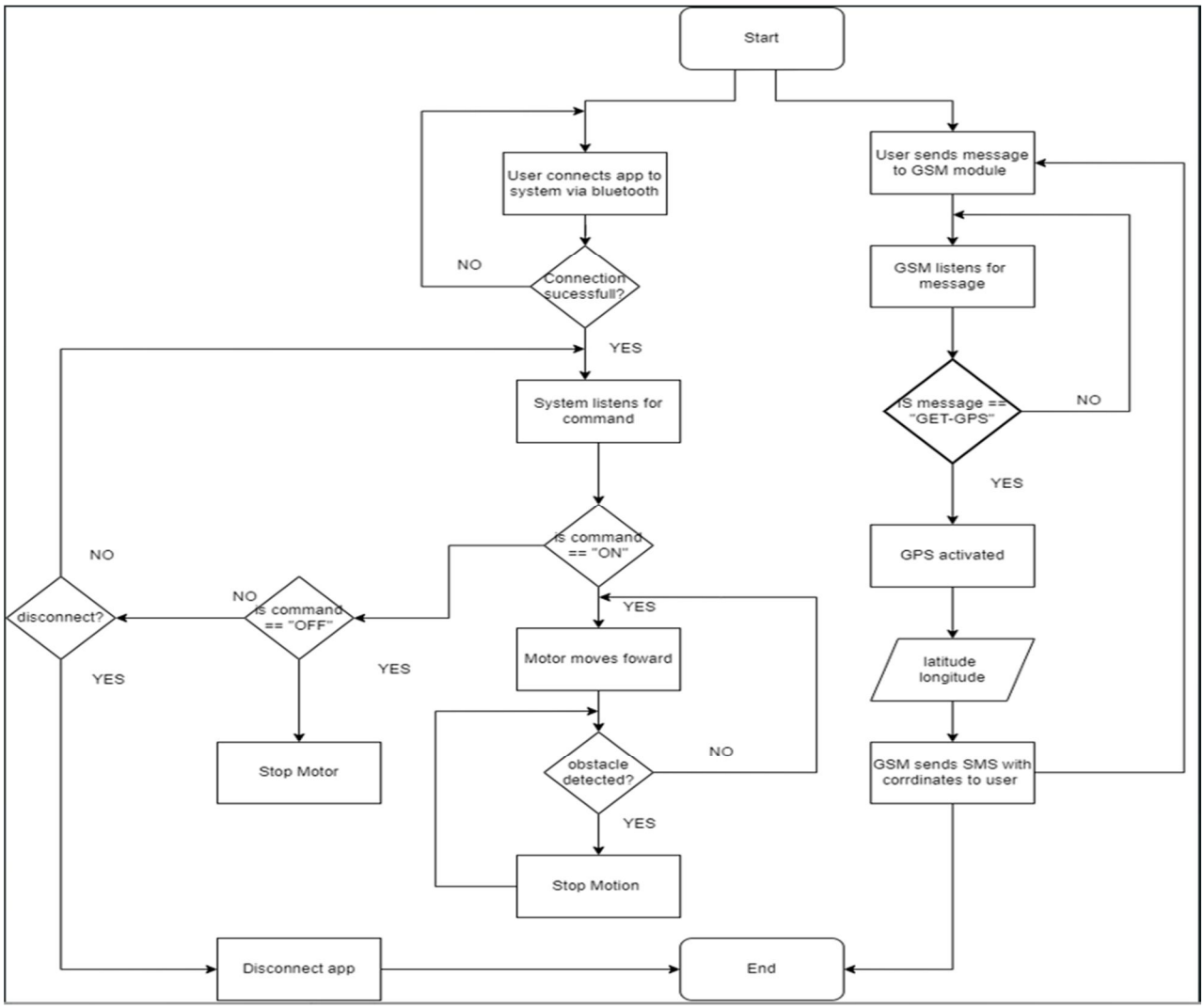


System Architecture

The proposed system has 2 main functionalities: Location Tracking and Automated Motion.

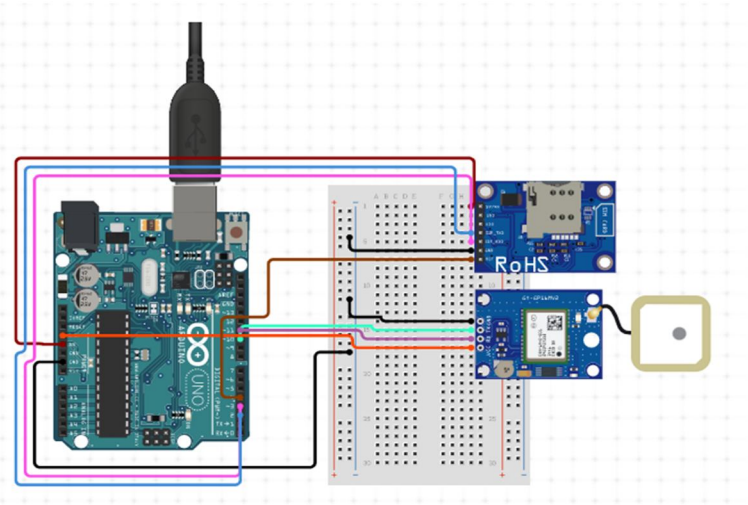
Location Tracking: This functionality is used for obtaining the geographical location of the bag in the form of latitude and longitude coordinates using the NEO 6M GPS module and the SIM900A GSM module. The user can send a trigger message from their smartphone to obtain the location of the system. The trigger used is “GET-GPS". Coordinates will only be sent after the system receives this particular trigger message and not for any other message. After the system receives the trigger message via the GSM module, it activates the GPS module which connects with GPS satellites and obtains latitude and longitude coordinates. The system then creates a google map link and appends these coordinates to the URL as parameters. This link is sent to the phone number registered by the user via SMS. The user can click the link received and will be redirected to the google maps page which will display the location of the system. The Tiny GPS library is used to get cleaned data from the GPS module and the Software Serial library for creating multiple serial ports on the Arduino board. Standard AT commands are used to communicate with the GSM module.

Automated Motion: The system is capable of detecting obstacles and stopping motion when it encounters a barrier. DC motors are attached to an L298N Motor driver to control the motor. The motor driver helps control the voltage provided to the motor and provides speed control facilities. It enables the motor to be connected to and controlled by the Arduino. An IR sensor is used to detect obstacles in the path of the system. It is connected to a digital pin of the Arduino board and provides a continuous stream of high or low output to the pin. A HIGH output means no obstacle is present and a LOW output means an obstacle is detected. An HC 05 Bluetooth sensor is used to connect the system to the user's smartphone application. The application has functionalities to connect to the Bluetooth sensor and buttons to give commands for starting and stopping the motors. The application is developed for android devices for this version of the project. Once the user connects to the system using Bluetooth via the app, they have the option to start or stop the motion of the motors. On pressing the ON button, the motors begin forward motion. If an obstacle is detected, the motion stops. Once the obstacle is removed motion resumes. Motion is stopped when the user clicks OFF. The Simple Timer library has been used to simulate the parallel execution of IR and Bluetooth loops as well as the Software Serial library to add a serial port. Both modules work together in parallel but are independent of each other. Location tracking has a wide range of operations. It can work in the entire geographical area that the sim network provider operates in. The motion detection functionality will work only in the range of Bluetooth connectivity.



The flow of System

Location tracking:



Location Tracking

The GPS and GSM modules are connected with the Arduino as portrayed in the above circuit diagram. Using the Software Serial library, we create two sets of serial ports using which each of the two modules can communicate with the Arduino Uno board. [5] The GSM modules operate on a 5V input supplied by the Arduino board and it has a ground connection as well. Arduino pins 2 and 3 are converted to RX and TX pins respectively using the software serial library. The TX pin of the GSM is connected to pin 2 and RX pin to Arduino pin 3. The GPS module operates on a 3.3V input supplied by the Arduino and is provided with a ground connection. Arduino pins 10 and 11 are converted to RX and TX pins respectively. The TX pin of the GPS module is connected to pin 10 and the RX pin is to pin 11 or the Arduino. The system initializes the GPS module to get the current coordinates of the bag. These are parsed through the Tiny GPS Plus library to get latitude and longitude coordinates from raw GPS NMEA data. These coordinates are stored as global variables.

Next, the serial ports of the GSM are activated. The GSM is set to listen mode using AT + CMGF =1 and AT + CNMI = 2,2,00 commands wherein it listens for any text messages sent to it. Once the module receives a text message, it checks to see if it matches the trigger message which has been set as “GET-GPS". Only if the text message received matches the trigger message, the system calls the sendSMS() function. Once the sendSMS() function is called, the GSM is set to send mode using the AT + CMGS command, and the recipient's (in this case the user’s) mobile number is provided. The body of the messages is a google map API link to which the latitude and longitude values, retrieved from the global variables, are passed as arguments. The google map API URL is as follows: “https://maps.google.com/maps?q=" and latitude and longitude values are concatenated as comma-separated values at the end of the URL. This URL is then sent to the user following which the GSM is again set to listen mode to wait for future messages.

Automated Motion with obstacle detection:

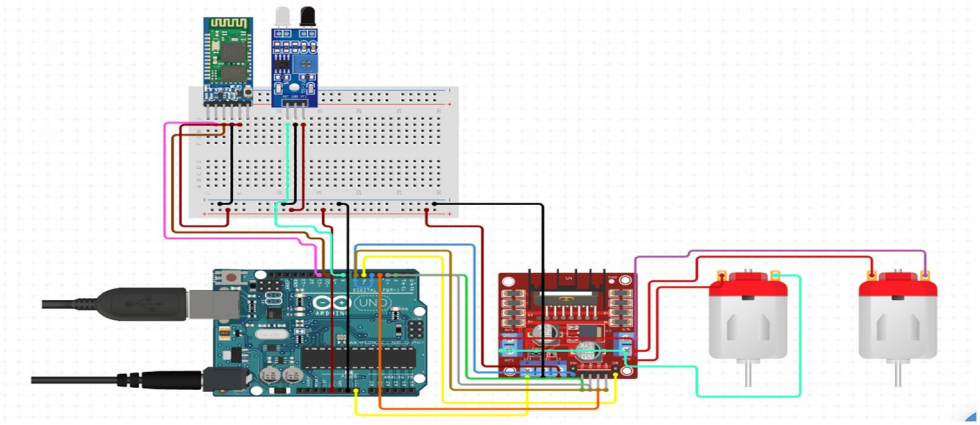


Figure 5.1.c. Automated Motion with obstacle detection

This subsystem consists of three components, the motors and motor driver, an IR sensor to detect obstacles, and a Bluetooth module to connect to the android application.

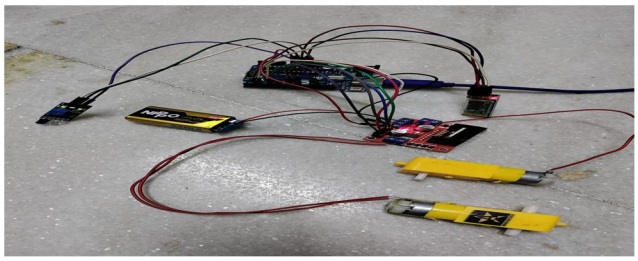
The two DC motors are connected to a motor driver which is an essential interface between the Arduino and the motors. The motor driver enables us to start and stop the motors on command. It also regulates the voltage going to the motors. It makes speed control possible by controlling the voltage supplied and is also capable of changing the direction of motors. It has two types of pins, EN and IN pins. The EN pins are used for speed control and IN pins are for start, stop, and direction control.

A maximum of two motors can be connected to one motor driver. The ENA and ENB are connected to Arduino pins 5 and 6 respectively while the IN1, IN2, IN3, and IN4 pins are connected to pins 2,3,4, and 7 respectively.

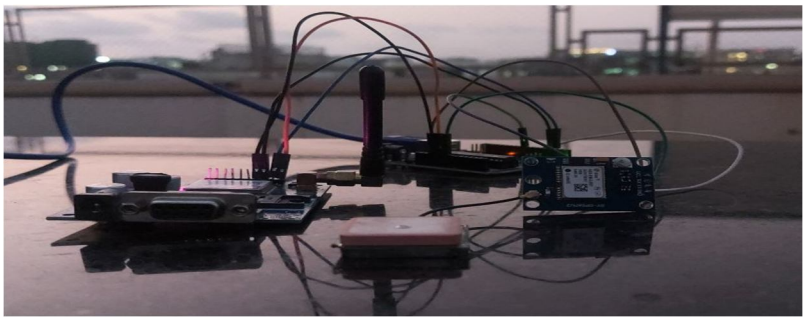
The Bluetooth module requires a serial port to be set up which has been set as 10 and 11 along with a 5V power connection and a ground connection. The android application connects to the Bluetooth module. Upon successful connection, the user can press buttons on the app which will send specific commands to the Arduino via the sensor. On pressing the “ON” button, the integer value 78 is sent, and on pressing “OFF” the value of 70 is sent. Based on these values, a loop has been set up in the system with an if-else condition. If the value received is 78, the motor pins are set to a combination of HIGH and LOW to enable clockwise or forward motion and they start moving and if the value is 70, the motor pins are set to LOW to stop motion

The IR sensor only requires one digital pin connection apart from 5V power and ground. Arduino pin 8 is dedicated to receiving the output stream from the sensor. And has been defined as an input pin in the code.

If the sensor provides an output of HIGH, it means it detects no obstacles and if it sends an output of LOW it means obstacles are detected. A separate loop is a setup to analyze the input provided by the IR sensor. It has an if statement with a dual condition which states that the digital pin input should be HIGH and at the same time the command received from the Bluetooth sensor should be 78 and only then the motors may move. If the digital pin receives LOW input the motor stops irrespective of the Bluetooth command. These two loops run in a pseudo-parallel environment using the Simple Timer library by calling each loop every second to provide the effect that they run in parallel. This causes a maximum delay of one second but this latency is tolerable for our system given the range of the IR sensor.



Interface Design: Motor Setup



Interface Design: GPS-GSM Setup

VI. TESTING AND RESULTS

A. *Test Plan*

Features of the system that need to be tested:

1) *Motion Detection:* Detect if the motor driver is connected to the Arduino and with the external battery properly. Also need to check whether the motors are connected to the motor driver or not. The connection of the motor driver and Arduino should be correct in proper order with their suitable pin number. After doing so we need to check when the motor driver receives the input voltage and external voltage and the motor should work accordingly.
2) *Tracking the Location:* GPS and GSM are connected to the Arduino and when they receive the input voltage the LED should be turned on. GPS should double-blink after every 6 seconds. GSM's nwk signal should blink after every 3 seconds. When the message is triggered the GPS should give the exact location of the bag.
3) *Obstacle Avoidance:* IR sensor is used to detect whether or not the obstacle is there. It should detect the obstacle and stop the bag from moving. When it receives the signal of some obstacle nearby, it should react to our system.
4) *User Control:* Through the mobile app we can start and stop the motor. The Arduino is connected to a Bluetooth module which provides connectivity to the smartphone. For this, it is necessary that the user who is using this product must have a Bluetooth-enabled smartphone. So, when the user clicks on the stop button, the bag should stop moving and when we click on start the bag should start moving.

B. *Testing Methods*

1) *Unit Testing:* As per software engineering unit testing is the process that examines the smallest components of an application, known as unit, individual, or independent for a specific operation. This product contains various individual components, so it was necessary to research each and every component. After research, each component was implemented one by one to see whether it was working independently or not. In this manner, each and every sensor and module was tested independently.

2) *Integration Testing:* After completing unit testing, the next stage is integration testing. The components were integrated using the bottom-up approach. Taking one component, integrating with another, and then integrating another module, and so on. For instance, the Bluetooth module was integrated with the IR sensor which was further integrated with the motor, and so on.
3) *System Testing:* Once the integration of the modules was successful, the system as a whole was tested to ascertain whether the functionalities and outputs were accurate or not. In case errors were observed, changes were made accordingly to the product. In this manner, the system has been tested comprehensively.

C. Test Cases

Unit Testing				
Sr No.	Test Case	Expected o/p	Actual o/p	Remark
1	Arduino with USB	Sensors should activate and code should get uploaded	Sensors activated and code uploaded "ON" LED stable TX pin blinking. Once the code is uploaded TX is stable.	Pass
2	Bluetooth, powered on but no connected	Commands from the mobile app not sent	Commands from the mobile app not sent. Continuous blinking of LED (red)	Pass
3	Bluetooth, powered on and connected	Send commands from mobile app	Send commands from mobile app Double blink every 6 seconds	Pass
4	IR sensor power on	Detecting obstacles	Detecting obstacles VCC LED stable OUT LED: on when obstacle detected OUT LED: off when no obstacle	Pass
5	GSM power on, with sim: network detection	Sending and receiving SMS messages	Sending and receiving SMS messages. DC-PWR led: stable NWK: blink every 3 seconds when network detected	Pass
6	GSM power on without SIM	Should not send and receive SMS messages	Should not send and receive SMS messages DC-PWR led: stable NWK: blinks continuously	Pass
7	GPS in confined space	Detection of satellite	Satellite not detected Blue LED does not blink/blinks occasionally	Fail
8	GPS in open skies	Detection of satellite	Satellite Detected Blue LED blinks continuously at regular intervals	Pass
9	Motor without driver with 9V battery	Spinning of Motors	Motors Spin Motor wheels move very fast at a fixed speed and can be stopped and started only by removing / connecting battery	Pass
10	Motor with driver	Spinning of Motors	Motors Spin Motor works with speed control. Can be started and stopped using code.	Pass

Test Cases

Location Tracking				
Sr No.	Test Cases	Expected o/p	Actual o/p	Remark
1.	Getting a response in a confined space	Receive location link	No response	Fail
2.	Getting a response in open skies	Receive location link	Receive location link	Pass
3.	Receiving location link with GET_GPS trigger	Receive location link	Receive location link	Pass

Test case of Location Tracking

Automated Motion				
Sr No.	Test Cases	Expected o/p	Actual o/p	Remark
1.	Press start button on app	Forward motion of motor	Forward motion of motor	Pass
2.	Press start button on app	Motor stops moving	Motor stops moving	Pass
3.	Pressing ON button while obstacle present	No movement of motor	No movement of motor	Pass

Test cases of Automated Motion

VII. CONCLUSION AND SCOPE

As a result of this project, an innovative carry bag was developed which makes life easier and smoother. The main aim of the project is to facilitate the transport of heavy bags. So through this project, the problem of dragging luggage was tackled and additionally, some security features were provided. This innovation is low-cost and user-friendly. The location of the system can be obtained by the user on their smartphone and they can also stop and start the bag. In this project, the proposed system is highlighted by the GPS (Global Positioning System) tracker and manual drive-through app.

VIII. ACKNOWLEDGEMENT

We are extremely grateful for being given the opportunity to make an IoT-based project in our IT department which is mostly software based. We are thankful to our guide, Prof. Arjun Jaiswal for providing us with guidance in this domain. We are also grateful to Prof. Priyanca Gonsalves for helping us throughout the project. We would also like to express gratitude for all the resources (components, labs) that were provided to us by our department during the project. We learned a completely new domain through this project and learned different tools/technologies as well. The exposure to IoT helped us understand the hardware side of different projects.

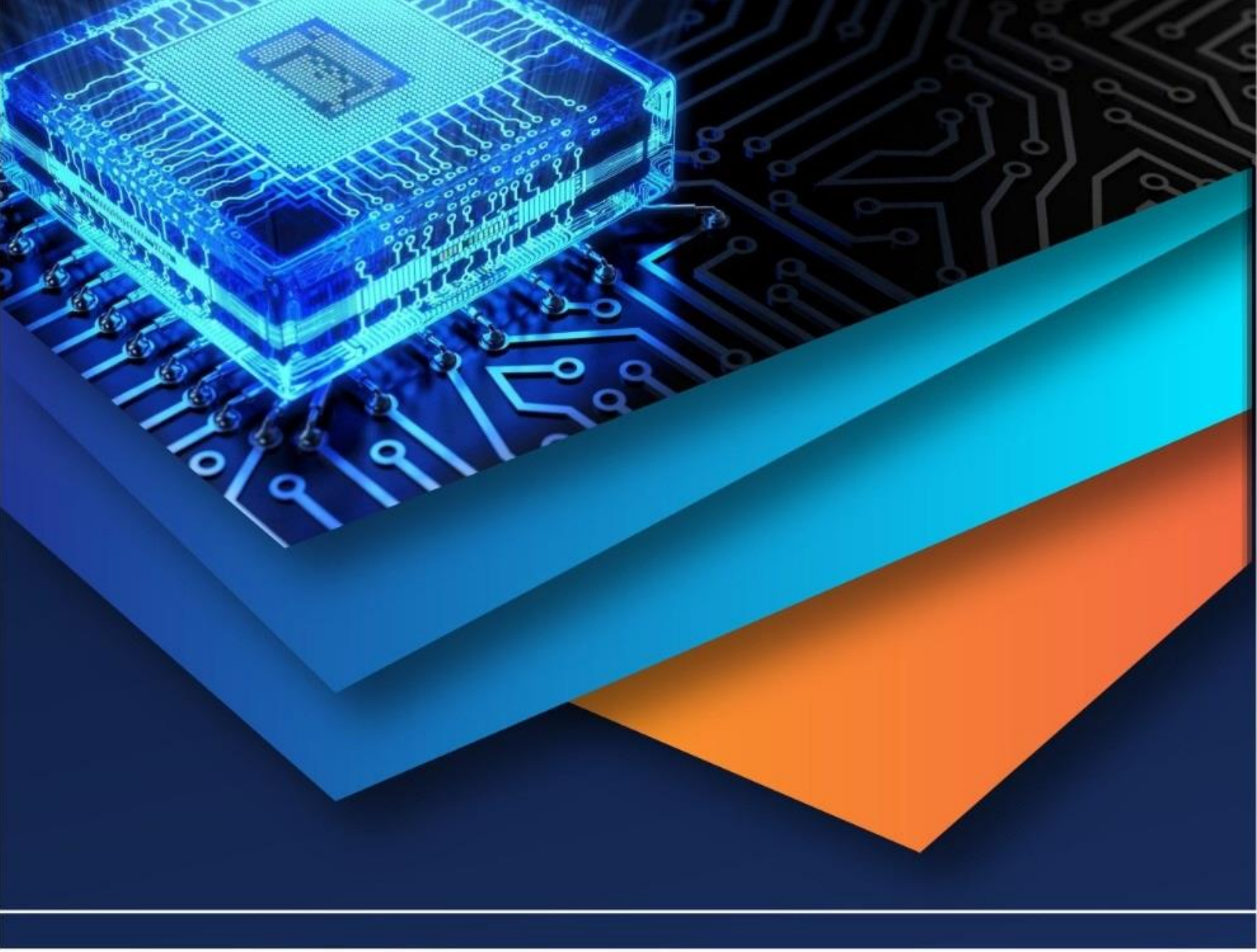
REFERENCES

[1] K. S. Nair, “Smart Luggage Tracker,” International Journal of Research in Engineering, Science and Management, vol. 2, no. 6, p. 3, June 2019.
[2] B. R. C. R. M. J. Sudha Senthilkumar, “LUGGAGE TRACKING SYSTEM USING IOT,” International Journal of Pure and Applied Mathematics, vol. 117, no. Special, p. 9, 2017.
[3] S.Karthick, “SMART LUGGAGE TRACKING AND ALERT SYSTEM USING ARDUINO,” International Research Journal of Modernization in Engineering Technology and Science, vol. 2, no. 5, p. 6, 2020.
[4] K. H. K. J. D. J. K. E. Maniyar Rut Vipulbhai, “Smart Luggage Carrier,” International Journal Of Creative Research Thoughts (IJCRT), vol. 8, no. 5, p. 4, 2020.
[5] Arduino, “SoftwareSerial Library,” Arduino.cc, 15 11 2022. [Online]. Available: https://docs.arduino.cc/learn/built-in-libraries/software-serial. [Accessed 2022].
[6] M. Hart, “Arduiniana,” Arduiniana, [Online]. Available: http://arduiniana.org/libraries/tinygpsplus/. [Accessed 2022].
[7] “Arduino UNO R3,” Arduino.cc, [Online]. Available: https://docs.arduino.cc/hardware/uno-rev3. [Accessed 2022].
[8] “Arduino Uno,” Wikipedia, [Online]. Available: https://en.m.wikipedia.org/wiki/Arduino_Uno. [Accessed 2022].
[9] “Arduino UNO,” JavaTPoint, [Online]. Available: https://www.javatpoint.com/arduino-uno. [Accessed 2022].
[10] “IR Sensor Working,” robocraze, [Online]. Available: https://robocraze.com/blogs/post/ir-sensor-working. [Accessed 2022].
[11] “WHAT IS AN INFRARED SENSOR?,” YoungWonks Coding Classes for Kids, [Online]. Available: https://www.youngwonks.com/blog/What-is-an-infrared-sensor. [Accessed 2022].

©IJRASET: All Rights are Reserved

[12] "GPS Receiver Module," ElectronicWings, [Online]. Available: https://www.electronicwings.com/sensors-modules/gps-receiver-module. [Accessed 2022].
[13] "GPS Modules Selection Guide," seeedstudio, [Online]. Available: https://wiki.seeedstudio.com/GPS-Modules-Selection-Guide/. [Accessed 2022].
[14] "All You Wanted to Know About GSM Module and GPRS Module," Electronics For You, [Online]. Available: https://www.electronicsforu.com/resources/gsm-module. [Accessed 2022].
[15] "L298N Motor Driver Module," Components 101, [Online]. Available: https://components101.com/modules/l293n-motor-driver-module. [Accessed 2022].
[16] "NEO-6MV2 GPS Module," Components 101, [Online]. Available: https://components101.com/modules/neo-6mv2-gps-module. [Accessed 2022].
[17] "SIM800L GSM Module," Components 101, [Online]. Available: https://components101.com/wireless/sim800l-gsm-module-pinout-datasheet-equivalent-circuit-specs. [Accessed 2022].
[18] "Arduino Uno," Components 101, [Online]. Available: https://components101.com/microcontrollers/arduino-uno. [Accessed 2022].

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)