BSIT-F20-014

03-135172-018 SYED SAAD BIN AHMAD

03-135172-038 MUHAMMAD ABUBAKAR UMAR

Autonomous Suitcase with Smart IoT Solution

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Information Technology

Supervisor: Muhammad Zunnurain Hussain

Department of Computer Sciences

Bahria University, Lahore Campus

July 2021

C e r t i f i c a t e

We accept the work contained in the report titled

"**Autonomous Suitcase with Smart IoT Solution**"

Written by

SYED SAAD BIN AHMAD

MUHAMMAD ABUBAKAR UMAR

As a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Information Technology.

Approved by:

Supervisor: Muhammad Zunnurain Hussain

____(Signature)

July 2021

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations, which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name
03-135172-018	SYED SAAD BIN AHMAD
03-135172-038	MUHAMMAD ABUBAKAR UMAR

Date : 9 July 2021

Autonomous Suitcase with Smart IoT solution

ABSTRACT

We often see people dragging and pushing their heavy suitcase / bags on airport and railway stations. Sometime their bags are misplaced and they have trouble in finding them or are stole. Sometimes it difficult to handle such large bags while you are holding a child or your hands are occupy with other activities. Passengers with disabilities have problem in carrying their luggage. The above-mentioned problems are mainly happening with passengers.

AUTONOMOUS SUITCASE is the sole solution to the stated problems, as it comprises the solution for the majorly occurring and identified problems. For this, we will apply the SMART IoT solution on a suitcase by which our AUTONOMOUS SUITCASE will be easily trackable and it will move behind the user maintaining constant distance avoiding obstacles. Moreover, it will be trackable by an Application using GPS system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background

The luggage handling and carrying is always a challenge for passengers and hence providing ease with the emerging technology in this area is important. The luggage bags and suitcase can be lost or very difficult to carry for a specific range and kind of passengers.

The Autonomous Suitcase is the primary solution for this type of problems and these problems can be eradicate by providing a smart solution to the humanity for their benefits. Implementing Smart IoT solutions on a suitcase to make it Autonomous so that bag can run on wheels avoiding the obstacles and follow the passenger maintaining a constant distance with an adequate speed and GPS location of the suitcase on the map to identify the current location are the main constituents of this project.

1.2 Problem Statements

Passengers on railway stations and airport are usually find struggling in handling their heavy and big luggage bags with other belongings sometimes their bags are stolen or lost.

1.3 Aims and Objectives

Following are the objectives of our project:

1. Passenger ease in moving luggage from one place to another.
2. Locating the lost / stolen luggage.
3. Self-following Suitcase.
4. Facilitating the disabled passengers and the mothers with an infant baby.

1.4 Scope of Project

The AUTONOMOUS SUITCASE does not only get rid of traditional trolley and luggage suitcase and bags but also it also provides us the ease to move free hand. It also comprises the Utility to locate luggage when misplaced or lost.

The AUTONOMOUS SUITCASE will have an IoT solution to the problem as it will move where the user moved maintaining a constant distance and pace (suitcase speed cannot be determined yet as it depends upon motors, load, and sensor response), avoiding a range of obstacles while moving carrying fixed weight in it. This will result in ease for the verity of passengers. A GPS system in the SUITCASE will allow the passenger to locate it on the map using latitudes and longitudes as a result the Suitcase will not get lost while traveling and your precious goods will remain with you.

CHAPTER 2

LITERATURE REVIEW & SRS

2.1 Purpose

The Autonomous Suitcase with Smart IoT Solution Is a hardware based project, which contains sensors, drivers, wires and circuit boards. This suitcase has a purpose of following the passenger automatically. Enabling this functionality requires some linkage with the passenger and the hardware. For this purpose, we are using a Mobile Application as the command and control driver for the Autonomous Suitcase. This Software Requirement Specification covers the software's Aspects and illustrate its usage in brief.

2.2 Product Scope

The Software, which controls the aspects and features of the Autonomous Suitcase. It will consist of A GPS system in the SUITCASE, which will allow the passenger to locate it on the map using latitudes and longitudes as a result the Suitcase will not get lost while traveling and your precious goods will remain with you.

There are some other optional product scope like theft alarm when the Suitcase is disconnect with the Mobile Application, real time weight monitoring using the Application.

2.3 Product Functions

Following are the product Functions:

- Suitcase tracking
- Linking hardware with Software
- Theft alarm

2.3.1 User Classes and Characteristics

Using the Autonomous Suitcase is does not require any complex task. It requires the powered Suitcase and a Passenger who is Android Mobile Application will connect to the Suitcase Bluetooth. After the successful connection user can have following:

1. Passenger / User can connect/ disconnect the Autonomous device
2. Passenger / User can keep track of the location and locate the Suitcase in case of misplacement and theft
3. Passenger / User can have alert / alarm in case of disconnection
4. Passenger / User can have facility of Optional Tasks.

2.3.2 Operating Environment

Since the software aims to support the hardware device; therefore, the software will operate on following:

- Operating Hardware: Mobile Phone / Android Device
- Operating Software: Android 6 or above

As mentioned before, the system is a hardware system so it would mostly rely upon its embedded firmware for its functionality. However, the mobile application may require the mobile phone/handheld system to least be equipped with Android 6 (Marshmallow) or any later version.

In addition, the development of any desktop or web application is not on the agenda of this particular project, and it is limited to the development of the hardware unit and its companion mobile application.

2.3.3 Design and Implementation Constraints

The project may experience any or all of the following constraints that may potentially limit the development team's ability to design and implement the proposed system.

1. The memory and processing requirements to operate the software
2. The Standardized Design Principles to have a good Application interface

2.3.4 Assumptions and Dependencies

The ideal functionality of the system is associated with certain aspects of the operating environment.
It is assume that:

1. The sensors used in the project are highly accurate and flawless.
2. The sensory networks would be deployed to collect the data
3. The sensors would be compatible with the computing system onboard
4. The users of the system would be familiar with the functionality of the system.

Based on the previously mentioned assumptions, the dependencies of the system is highlight below.

1. The Autonomous Suitcase Application relies upon the working of sensors in the real environment / scenarios.
2. The system depends upon the sensory network to act accordingly as per stated / defined. The behavior of linked electric devices can be tailor accordingly.

2.4 Performance Requirements

Since the device comprises on one hardware and a software component that function in combination for optimized performance; therefore, the performance requirements would be as of following for both aspects of the system.

- Hardware:
  The proposed device heavily relies upon the communication between sensors and Mobile Application, which will derive the hardware, accordingly i.e. made for use on Railway Platforms and Airports; hence, the operating environment should be free from hurdles and too much obstacles.

- Software:
  To utilize the functionality of software application, the handheld computing device must be capable to transmit the data using Wi-Fi or Bluetooth communication protocols

2.5 Safety Requirements

Although safety remains the number one priority throughout the development process of the Software Application during the project; nevertheless, there may be certain aspects that are need to take care of. The generalized safety precautions for the safe utilization of the device are enlist below:

- The software should remember the presets set and their values of every functionality.
- While linking the software with sensors set adequate presets to save the hardware from failure or being damaged.
- Software is develop for human machine interface, ease of maintenance and modification or enhancement.

2.6 Security Requirements

As it is, describe that the primary use of the device and its software is used to facilitate and provide ease to the passengers traveling via Airport or Railway Station. As the primary function of the Autonomous Suitcase is to follow the passenger, where ever he goes on the mentioned place. The security requirements for the software developed related to the project are as follows: The software will only pair with the device and make connection with it only to avoid communication disturbance and execution of functionality as mentioned. The software will work on time in terms of giving alert to the passenger when Suitcase get disconnect while walking.

2.7 Software Quality Attributes

Following quality attributes would be consider while designing and developing the software.

- The software is built, on the Blynk platform, which contains the world largest IoT libraries and functionalities and provides ease to develop all kinds of IoT related software.
- The Software is easily alterable as per hardware and sensors alteration.
- The software interface is aesthetic to use which provide easy functionalities and its UI is right according to the universal HCI principles.

2.7.1 Other Requirements

The system (as it combines the software and hardware aspects) is fairly an open system which may be prone to legal and privacy issues; therefore, all the users would be required to properly read and agree to the privacy and legal policy presented to them before they use the device. Other than that, there are no specific or special requirements but the ones that have already mentioned.

2.8 Literature Review

2.8.1 Background

As we have said in our previous documents, luggage has always been a challenge for passengers and therefore it is important to facilitate the technology that is emerging in this area. Luggage and suitcases can be lost or very difficult to carry over certain distances and passengers with special needs. An autonomous suitcase is the ultimate solution to this type of problem and these problems can be eradicate by providing a wise way out to humanity in its favour. Applying the Smart IoT solution to the suitcase to make it autonomous so that the bag can move on wheels, avoiding obstacles and following passengers keeping the distance fast enough, and the GPS location of the suitcase on the map to identify the current location is the main compiler of this project.

There are many different carrying bags for many companies, but they do not contain accessories that make travel easier, especially for passengers with heavy luggage. Luggage does not necessarily weigh, although such features are highly desirable [1]. Tourists are often ask to transport goods from a distance, for example in the long corridors of terminal 15 of the airport. Carrying luggage can be difficult and requires an unwanted level of physical training. Even luggage that can be roll by wheels needs to be pulled or pushed, which can stretch the arms and back of the person carrying the goods. Some developers have incorporated electronics and motors into their products to provide self-transport of luggage. However, luggage design during self-driving follows from behind and behind the eyes of users and is limited in their ability to move to avoid obstacles, especially when used in crowded places, such as airports, hotels or sidewalks [2].

The manifestation of exposure includes a system of smart luggage that drives independently and has one or more motorized wheel units. The smart luggage system is configure, to automatically track all types of items, such as users moving in a certain direction. Although the embodiment of a smart suitcase system, related to a suitcase, is explain and illustrated here, the embodiment can be use with other types of mobile equipment, such as shopping carts.
In one aspect, the smart baggage system is configure to follow along the user side in a particular direction. In one aspect, the smart baggage system is configure to track along the user side within a predetermined area that the system has detected. In one embodiment, the smart baggage system is configure to track alongside the user, switch to the tracking position behind the user to avoid obstacles, and then switch to the tracking position next to the user. In one aspect, the smart baggage system may move in a different direction from the direction of the baggage head [2].

2.9 Tools and Techniques

2.9.1 Blynk:

Blynk is a platform where we can develop applications for Android or iOS OS. Provides users with a GUI. Parking information is provide to users through the Blynk application [4].

We have designed, developed and tested complete IoT solution blocks, so companies working at Blynk do not have to.

Blynk combines cloud platforms with applications that place things, people and data as the heart of business operations.

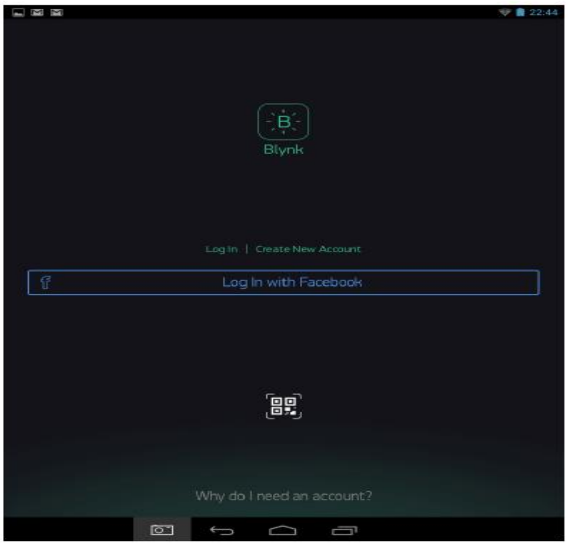

Figure 2.1: BLYNK login interface

Using the Blynk app is to control the Arduino board. Blynk allows us to create applications and then use them to operate Arduino boards that are connected anywhere in the world with a smartphone. This is one of the most exciting actions! Connections are make via Bluetooth between smartphones and Arduino boards [6].

Ultrasonic sensors are used to complete the most complex tasks involving the detection of objects.

2.9.2 Arduino UNO

Arduino is the first and most important company for open source computer hardware and software. The Arduino community refers to the project community and users who design and use a microcontroller-based expansion board. This expansion board is known as the Arduino module, which is a platform for open-source prototypes. Simplified microcontroller boards are available in a variety of expansion board packages.

The most common approach to programming is using the Arduino IDE, which uses the C programming language. This gives you access to a large, growing Arduino library thanks to the open source community [1].

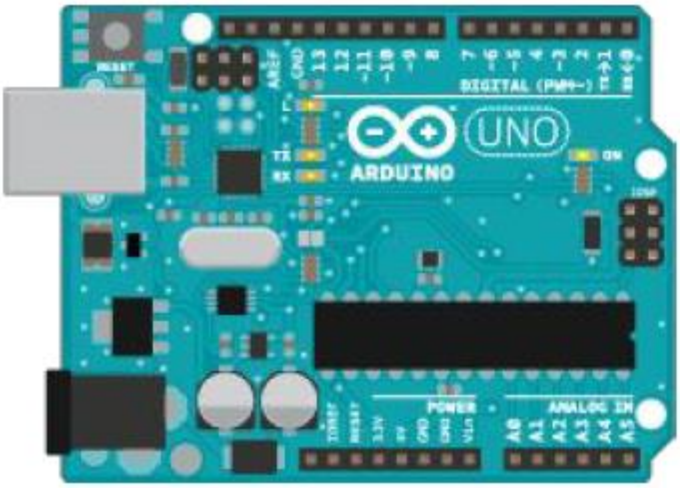

Figure 2.2: Arduino UNO Board

The Arduino board offers one critical advantage: an open source philosophy (both hardware and software), which benefits a large unprofessional community that thrives around the Arduino concept. A very rough estimate of community size can be obtain from a Google search that reports more than 12 million clicks for "Arduino". In fact, a large customer base and a growing market are showing increasing interest in the Arduino concept [2].

2.9.3 Ultrasonic Sensor

In industrial applications, reliability and exceptional measurements or levels characterize ultrasonic sensors with millimetre accuracy, because their measurement methods are reliable in almost all conditions. Infrared sensors are also use in many everyday products. Their low power requirements, simple circuitry and mobile features make it desirable.

In our project we made a small prototype car for adaptive cruise control where the prototype car, which

The heart will be Raspberry Pi, a microcontroller, which will lock the object / car in front of it when placed. We have to choose for this corresponding sensor. Therefore, we first conducted a study on the following ultrasonic sensors. [3]

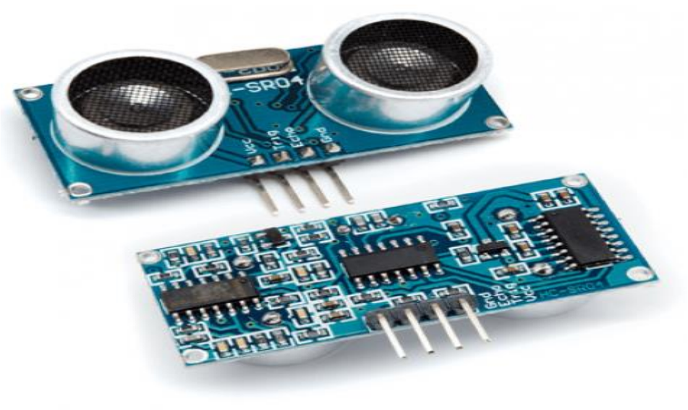

Figure 2.3: Ultrasonic Sensor - SR04

2.9.3.1 Ultrasonic Sensor features

There is no other method of measurement, which can be apply successfully in a wide variety and in many different applications.

- This device is very powerful, which makes it suitable for even the most difficult situations.
- The surface of the sensor is clean with vibration, making the sensor insensitive to dirt.
- The physical principle - the dissemination of voice - works, with a few exceptions, in almost all environments.
- Ultrasonic sensors have proven their reliability and durability in almost all industrial sectors [3].

2.9.4 Infrared Sensor

An infrared sensor is an electronic device that emits to detect some aspects around you. The IR sensor can measure the heat of an object as well as detect its movement. If the IR sensor does not receive infrared light, it means that there are no objects. [4]

Figure 2.4: Infrared (IR) Sensor

Passive infrared sensor systems include a first or second sensor unit, including a first or second light receiving element and a first or second optical system. The first sensor unit has an "oriented" direction directed to the upper part of the human body so that it can clearly determine the surface area of the first hour from the ground surface. The second sensor unit has a similar direction of "view" which is oriented downwards towards a point on the ground surface, which is distant from the clock determined by the place where it is placed so that it can determine the area of the second clock determined below the surface of the first clock. This passive type infrared sensor system also includes first and second level detection circuits. That can operate, to output a detection signal, only when the level of the output electrical signal from the first or second associated sensor unit exceeds a predetermined reference level. A human detection circuit for emitting, when detection signals are remove from the first and second level detection circuits, human detection signals indicating the entry of the human body into the surveillance zone, including the first and second surveillance areas [5].

2.10 Miscellaneous

2.10.1 GSM Module:

It is generally envisage that these systems will be equipped for four-wheelers, but for countries like Pakistan where most people use two-wheelers, the following is the cheapest source of anti-theft tracking systems. Luggage tracking systems are typically use by fleet operators for fleet management functions such as routing, delivery, information, and security on board. Other apps include tracking driving behaviour, such as employee employers or parents with teen drivers. The baggage tracking system is also popular in consumer luggage as a means of prevention and theft. Passengers can only follow the signals issued by the tracking system and locate stolen vehicles [7].

2.10.2 Breadboard:

These findings relate to new and improved boards for experimental circuits in printed circuit configurations, which provide relatively inexpensive and very practical methods for designing and testing different electronic circuits [8].

2.10.3 Motor and Batteries:

The battery charging system in electric motor vehicles includes stationary units and vehicle-mounted units. The stationary unit is equipped with a first dielectric housing and a high-frequency generator connected to the main power supply and at least one inductor placed near the front wall of the first dielectric housing. The unit mounted on the vehicle includes a second dielectric jacket located along the front wall of the first dielectric jacket. The second dielectric box is equipped with at least one sensor connected to a rectifier, which in turn is connect to the battery in the vehicle [9].

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Methodology

The software model, which we will follow in this project, is the “Waterfall” model. This model is flexible in hardware related projects. The Waterfall Model was the first principal Process Model to be implement. It is easy to comprehend and utilize. In a Waterfall model, each stage must be finished before the following stage can start and there is no covering in the stages.

In the Waterfall approach, the entire procedure of programming advancement is isolated into independent stages. The result of one stage goes about as a contribution to the following stage successively. This implies any stage in the advancement procedure starts just if the last stage is finished.

AUTONOMOUS SUITCASE will follow the Waterfall approach and by following the IoT principles using Arduino UNO. The sole working will be implement by variety of sensors, which will be attach and link by various means with Arduino. The sensors will direct the SUITCASE motion as per Arduino Programming. The application will made on Blynk, will be integrated with hardware and its functionality will be operated and monitored by the application made on Blynk platform.

3.2 Resource Requirement

We will need variety of hardware comprising Arduino UNO and variety of sensors integrated with application Software that will be compatible for the project implementation. The detailed resources and our requirements are yet to be finalize.

3.2.1 Expertise of the Team

Both team members pre-equipped with the level of knowledge needed for the successful completion of this project. Both of us have appropriated knowledge of IoT and related to the equipment required for project. The courses we studied in relevance with project are C++ Programming, Basic Electronics and Digital Logic Design (DLD). Other than that, we have gained adequate knowledge from internet and E-Courses online, which has increased our knowledge and interest in IoT.

3.2.2 Tools / Technology

We will be using App Builder Interactive tools for Software Development and Arduino UNO as a main Circuit Hardware and variety of sensors for project operation e.g. Ultrasonic Sensors (sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm), Infrared Sensors, GPS Sensor. Some driver tools are also required like Jumper Wires, Motors and Motor Drivers etc., some integration Tools like: Scolding Gun and wire, glue, Cardboard and Bread boards etc. moreover we would require rechargeable Power Source, Motors and Step Motors etc. A hard case SUITCASE will be use in which all the hardware will be place.

The hardware and software requirements could be flexible according to their availability but mostly they are available in market. The bellow mentioned table illustrates the main hardware requirements for Autonomous Suitcase.

Table 3.1: Resources required for developing Autonomous Suitcase

#	Name
1	laptops
2	smartphones
3	motor driver module
4	ultrasonic sensor
5	Motors
6	Arduino
7	rechargeable batteries
8	tires
9	IR sensor module
10	breadboard
11	jumper wires
12	GPS module
13	suitcase
14	GSM Sim module
15	Bluetooth module
16	Miscellaneous

3.3 Top level Architecture

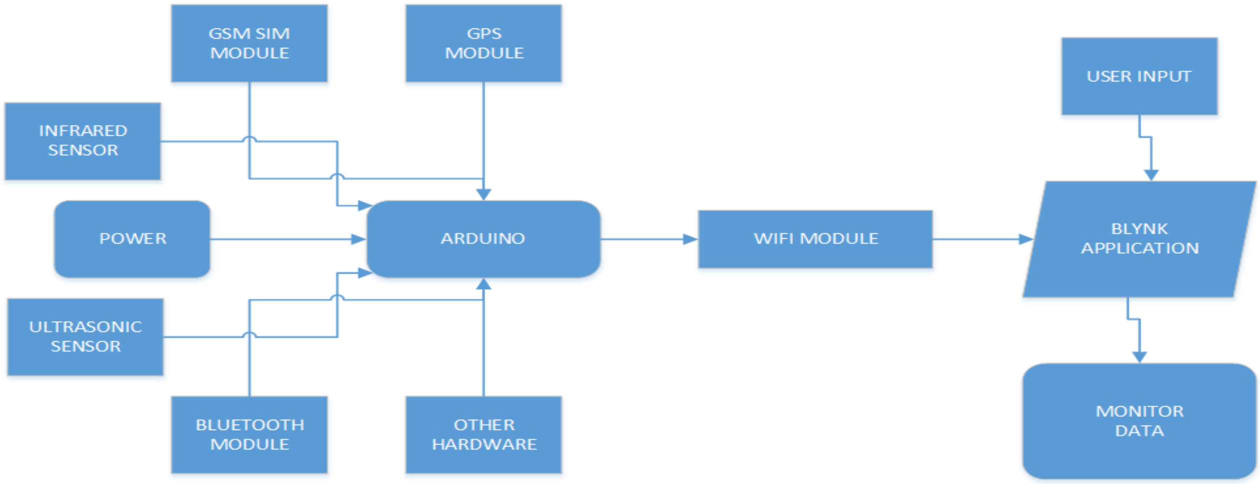

Figure 3.1: Block diagram representing entire system of Autonomous Suitcase

In the above figure, the block diagram illustrates the all components are attach to the main circuits and how the application interacts with the Arduino.

3.4 System Use Cases

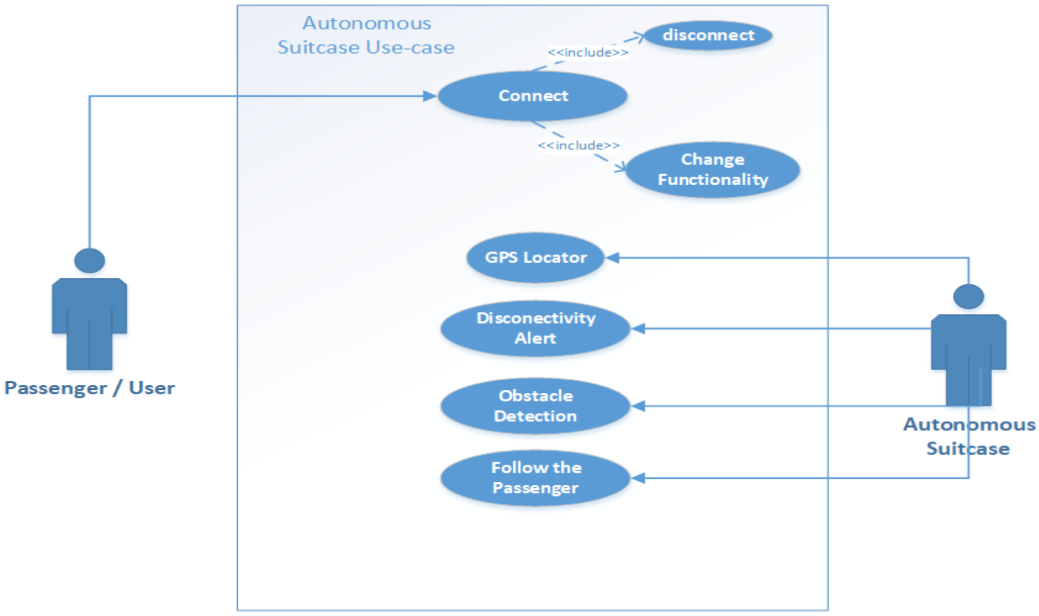

Figure 3.2: A general Use case diagram of Autonomous Suitcase

The above figure is illustrating the complete use case for Autonomous System. How the system work and the functionality is executing?

3.4.1 Description for Connection

Table 3.2: Use-Case details of Connecting with Application

Unique ID	Usecase-1
Objective	Connection with Autonomous Suitcase
Priority	High
Source	Mobile Application
Primary Actors	Passenger / User
Flow of Events	1. Select the option of connectivity (i.e. Bluetooth / Wi-Fi 2. Select the SSID of the device to be connected 3. Enter pass code
includes	Disconnect, Change Functionality
Pre-Condition	Application setup
Post-Condition	Connectivity

3.4.2 Description for GPS Locator

Table 3.3: Use-Case 2 details for device tracking

Unique ID	Usecase-2
Objective	Track Autonomous Suitcase
Priority	High
Source	Mobile Application
Primary Actors	Passenger / User
Flow of Events	1. Select the option to locate the suitcase 2. Click on the location 3. Find it on the map

includes	N/A
Pre-Condition	Connectivity
Post-Condition	Device Found

3.4.3 Description for Alert

Table 3.4: Use-Case 3 details for Alert

Unique ID	Usecase-3
Objective	Autonomous Suitcase disconnection Alert
Priority	Medium
Source	Mobile Application
Primary Actors	Passenger / User
Flow of Events	1. device crosses the limit of distance from its user 2. the connection between app and suitcase breaks 3. Alert will notify you about this event
includes	N/A
Pre-Condition	Connectivity
Post-Condition	Notification of device disconnection

3.4.4 Obstacle Detection

Table 3.5: Use-case of AS obstacle Detection

Unique ID	Usecase-4
Objective	Autonomous Suitcase Obstacle Detection
Priority	High
Source	Infrared Sensor
Primary Actors	Autonomous Suitcase
Flow of Events	1. device detects an obstacle 2. the sensor move the suitcase to alternate direction to avoid it

includes	N/A
Pre-Condition	Device if following the Passenger
Post-Condition	Obstacle has been avoided

3.4.5 Follow the Passenger

Table 3.6: Use-case for AS following the Passenger

Unique ID	Usecase-5
Objective	Autonomous Suitcase Following the Passenger
Priority	High
Source	Ultrasonic Sensor
Primary Actors	Autonomous Suitcase
Flow of Events	1. device detects the passenger 2. device maintain an adequate distance 3. the sensor move the suitcase towards passenger
includes	N/A
Pre-Condition	Stationary Suitcase
Post-Condition	Suitcase is following Passenger where ever he goes

3.4.6 Data Flow Diagram

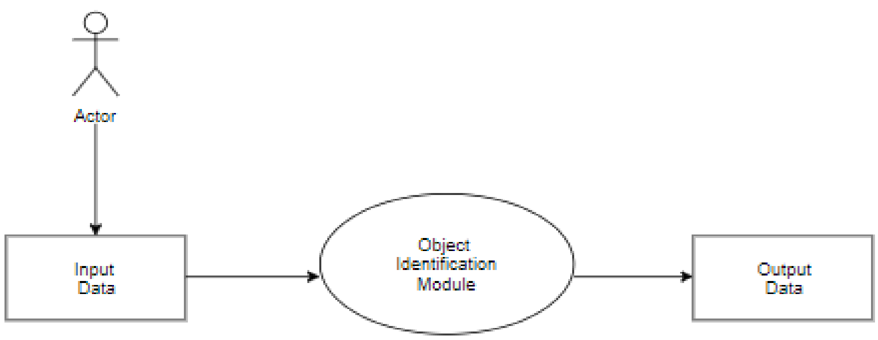

Figure 3.3: DFD for illustration of data flow between Passenger and Autonomous Suitcase

The data flow diagram DFD illustrates the simplest flow and sequence of execution of the operation, generically. In the above diagram the actor acts as an input for the device sensors and the action of device is output.

3.5 Sequence Diagram

A group object diagram is a sequence diagram because it describes how and in what order a group of objects work together. Developers and business professionals to understand the requirements of a new system or to document an existing process use these diagrams.

3.5.1 Connecting passenger with mobile phone.

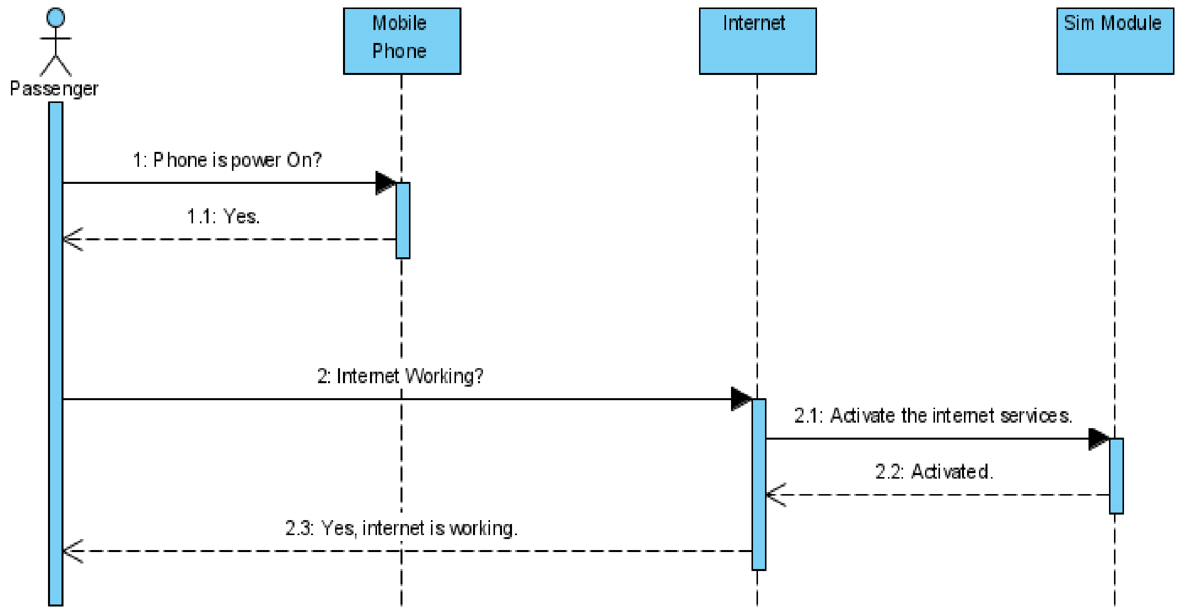

Figure 3.4: Sequence Diagram representing Mobile Connectivity sequence

The above mention set of sequence in the figure illustrates the sequence of data flow of passengers with the mobile phone. The passenger acts on the mobile phone and the network access provides access to the passenger to interact with the system.

3.5.2 Connecting mobile phone with suitcase.

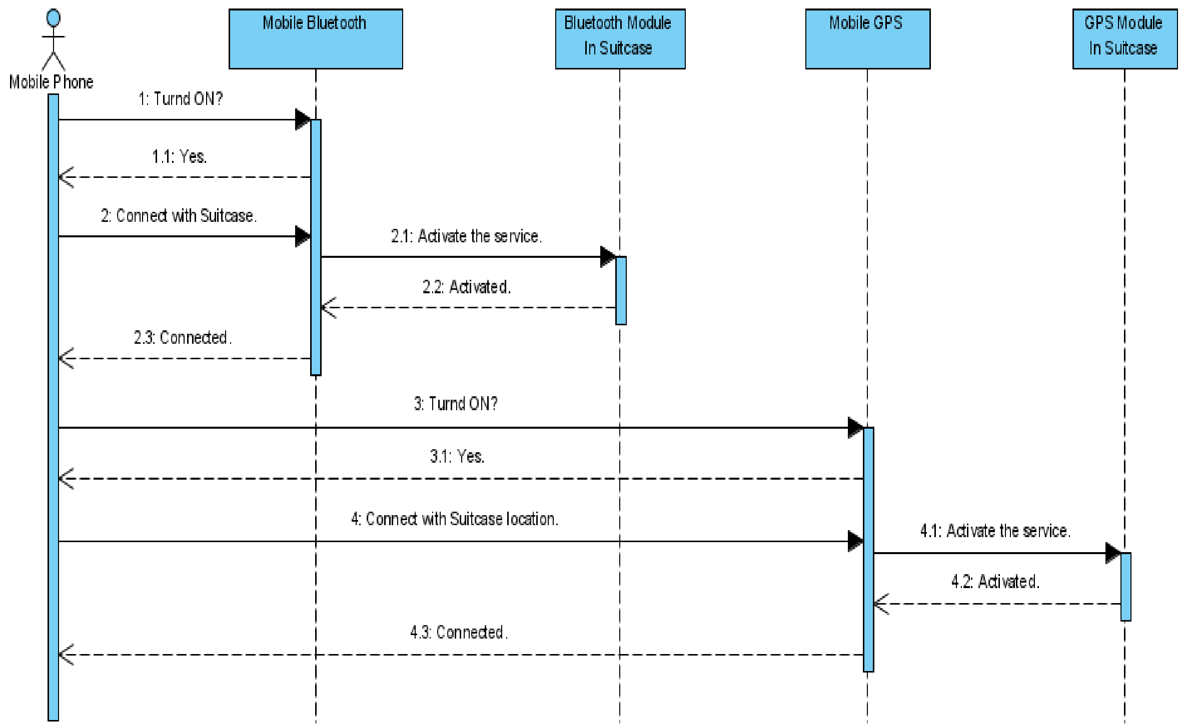

Figure 3.5: Sequence Diagram of Mobile Device interacting with AS

The above mention set of sequence in the figure illustrates the sequence of data flow of passenger's mobile phone with the Autonomous Suitcase. The passenger acts on the mobile phone and the other functionality is carry out with the suitcase with in the collaboration of both mobile and suitcase functions.

3.5.3 Connecting suitcase with passenger.

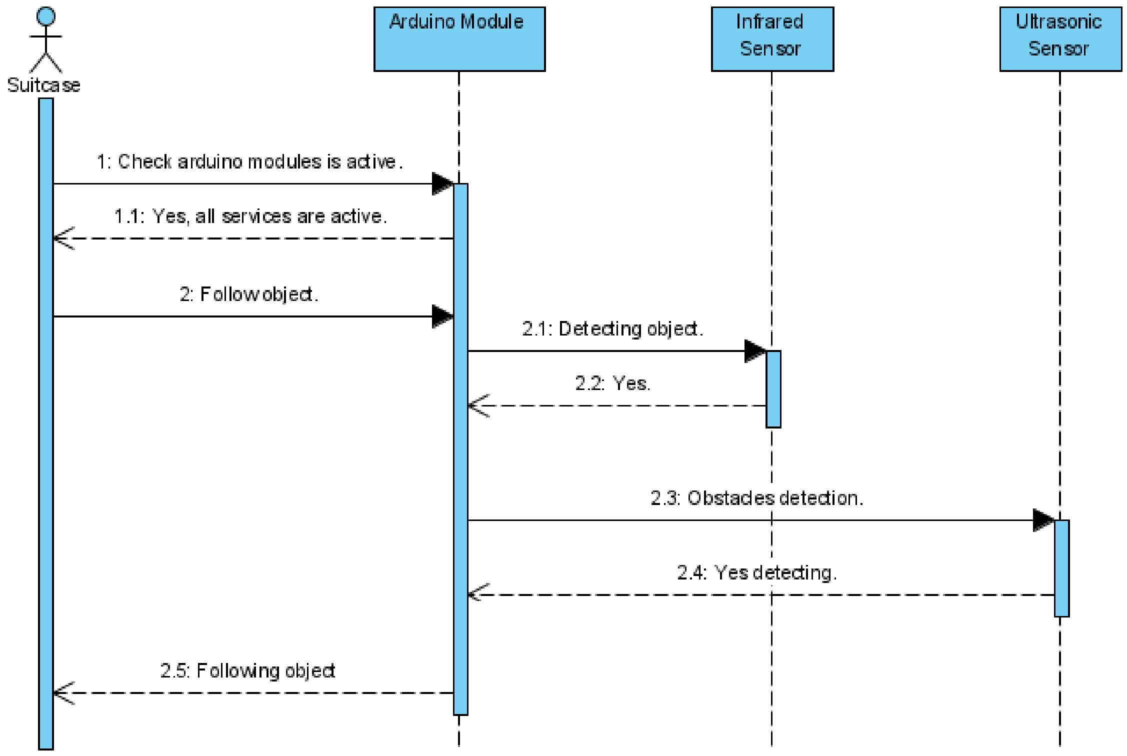

Figure 3.6: Sequence Diagram illustrating interaction of Passenger with Suitcase

The above mention set of sequence in the figure illustrates the sequence of data flow of Autonomous Suitcase with Passengers. The passenger's movement is analysed by the suitcase with the defined hardware and software functionality and it acts accordingly with variable scenarios

3.6 Communication or Collaboration Diagram

Communication diagram is an extension of an object diagram that shows objects moving with messages.

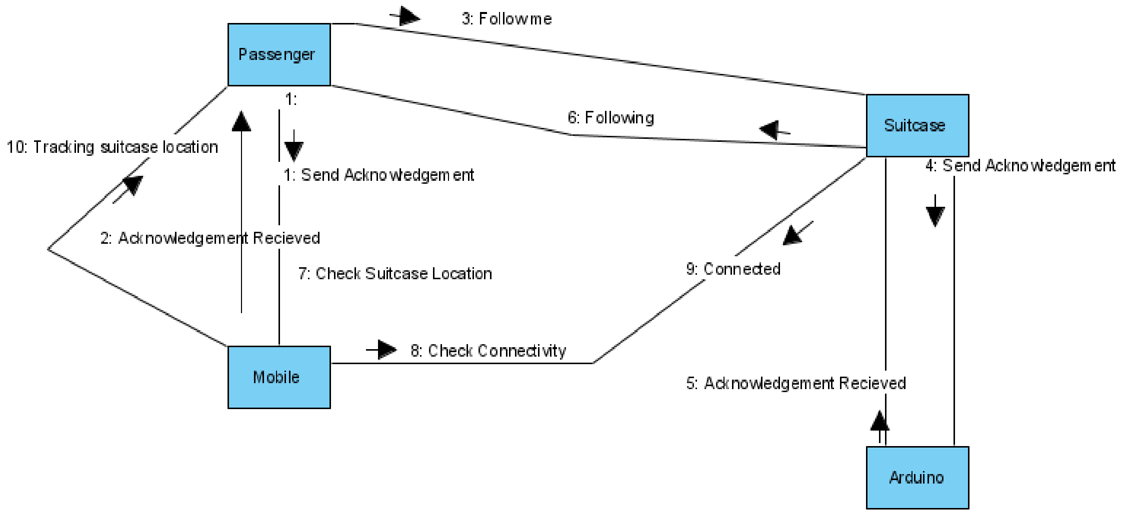

Figure 3.7: Illustration with a diagram Systems collaborating with one another

In the above communication diagram, it illustrates the complete communication of the all three main components of the entire system i.e. Mobile phone, Autonomous Suitcase and the passenger itself.

CHAPTER 4

IMPLEMENTATION

4.1 Hardware implementation

In this project the main components which are used are Arduino Nano, Lead Battery (5 Ampere), GPS GSM Sim module (808 EVB v3.2) GPS + GPRS, and Ultrasonic Sensor, Micro Servo 9g (sg90). DC Gear Motor (9V), Motor Driver Module (L298N), Bluetooth Module (Hc05) and Suitcase. All the components are integrating or interlink to get the actual output. Components are attach on the Vero board and configurations are make on Arduino Nano. Pulses coming from ultrasonic sensor are passing to the motor driver module through Arduino. GSM Sim module is use to track the location of the suitcase. Blynk App is configuring with Arduino and GSM Sim module the monitor the proper working of the suitcase.

4.2 Arduino Nano

Specifications are:

- Microcontroller: Microchip ATmega328P.
- Operating Voltage: 5 Volts.
- Input Voltage: 6 to 20 Volts.
- Digital I/O Pins: 14 (plus 6 can PWM output pins)
- Analog Input Pins: 8.
- DC Current per I/O Pin: 40 mA.
- DC Current for 3.3V Pin: 50 mA.

- Flash Memory: 32 KB of which 0.5 KB used by bootloader.

Open source hardware and software are Arduino, used for designing and for making digital and interactive objects, which will be sensing and controlling the objects. An IDE is use to write and load the code on physical board.

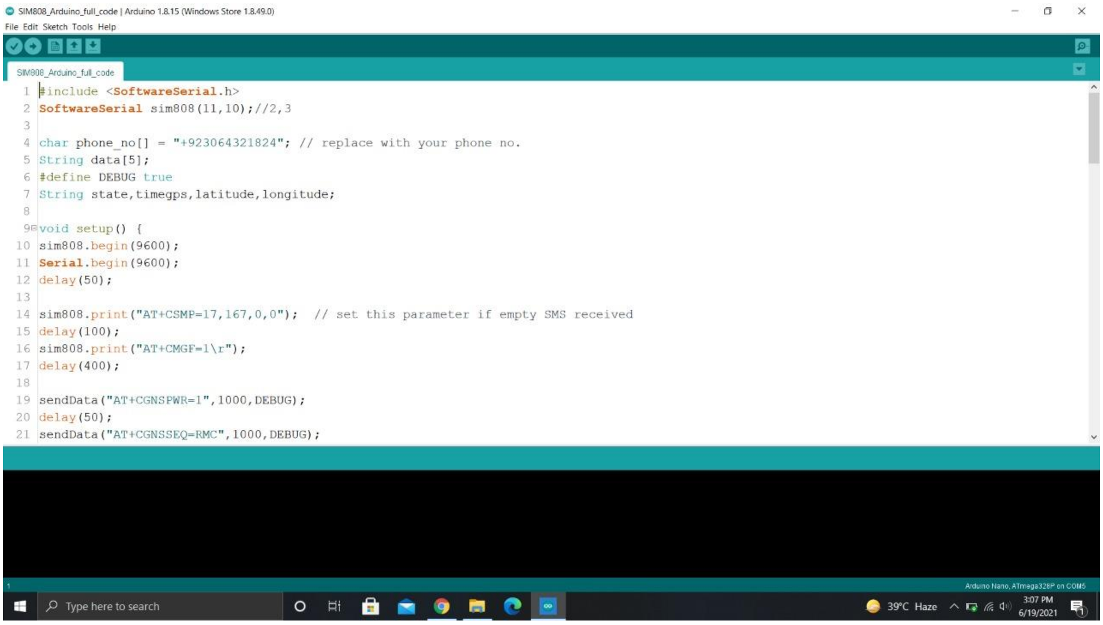

Figure 4. 1 Arduino IDE representing the code and compiling platform

4.3 808 GSM Sim Module (EVB v3.2) GPS + GPRS

Specifications are:

- Quad-band 850/900/1800/1900MHz.
- GPRS multi-slot class 12/10.
- GPRS mobile station class B.
- Dimensions: 24*24*2.6mm.
- Control via AT commands (3GPP TS 27.007,27.005 and SIMCOM enhanced AT Commands)
- Supply voltage range 3.4 ~ 4.4V.
- Low power consumption.
- Operation temperature: -40°C ~85°C

Global System for Cellular Communications (GSM) module is design to monitor wireless radiation via short message service (SMS). This module is capable of receiving serial data through which we can track the location of the suitcase.

4.4 Ultrasonic Sensor (hc-Sr04)

Specifications are:

- Provides 2-400 cm non-contact measurement function.
- Ranging accuracy can reach to 3mm and effectual angle is < 15°.
- Powered from 5V power supply.

It is use for detecting the distance to an object using sonar. Generate pulses, which move to motor driver module to effect the movement of the suitcase.

4.5 Motor Driver Module (L298N)

Specifications are:

- Double H Bridge Drive Chip: L298N
- Logical Voltage: 5V
- Drive Voltage: 5V-35V
- Logical Current: 0-36 mA
- Drive current: 2A (MAX single bridge)
- Max Power: 25 W
- Dimensions: 43 x 43 x 26 mm

The L298N is a dual H-Bridge motor driver, which allows speed and direction control of two DC motors at the same time on the instructions coming from Arduino.

4.6 Bluetooth Module (Hc-05)

Specifications are:

- Bluetooth protocol: Bluetooth Specification v2.
- Frequency: 2.4GHz ISM band.
- Modulation: GFSK (Gaussian Frequency Shift Keying).
- Emission power: ≤4dBm, Class 2.
- Sensitivity: ≤-84dBm at 0.1% BER.
- Speed: Asynchronous: 2.1Mbps (Max) / 160 kbps, Synchronous: 1Mbps/1Mbps.
- Security: Authentication and encryption.

Its use to provide the theft alters when the suitcase will be far away from the passenger more than 10 meters.

4.7 DC Gear Motor (9V)

Specifications are:

- Voltage: 9V.
- Speed at no load: 190 RPM.
- Current at no load: 0.05 A.
- Speed max. efficiency: 147 RPM.
- Current max. efficiency: 0.125 A.
- Torque max. efficiency: 0.0122 NM.
- Current at stall: 0.35 A.
- Torque at stall: 0.0784 NM.

DC motors have higher starting torque, quick starting and stopping, reversing, variable speeds with voltage input.

4.8 Micro Servo 9g (sg90)

Specifications are:

- Modulation: Analog
- Torque: 4.8V: 25.0 oz-in (1.80 kg-cm)
- Speed: 4.8V: 0.10 sec/60°
- Weight: 0.32 oz (9.0 g)
- Dimensions:
  - Length: 0.91 in (23.1 mm)
  - Width: 0.48 in (12.2 mm)
  - Height:1.14 in (29.0 mm)

Servomotors have a gear output shaft, which can be control by electric current for one (1) degree rotation at a time. This leads to a complete change of luggage correct rotations after the owner (passenger).

4.9 Integration of components

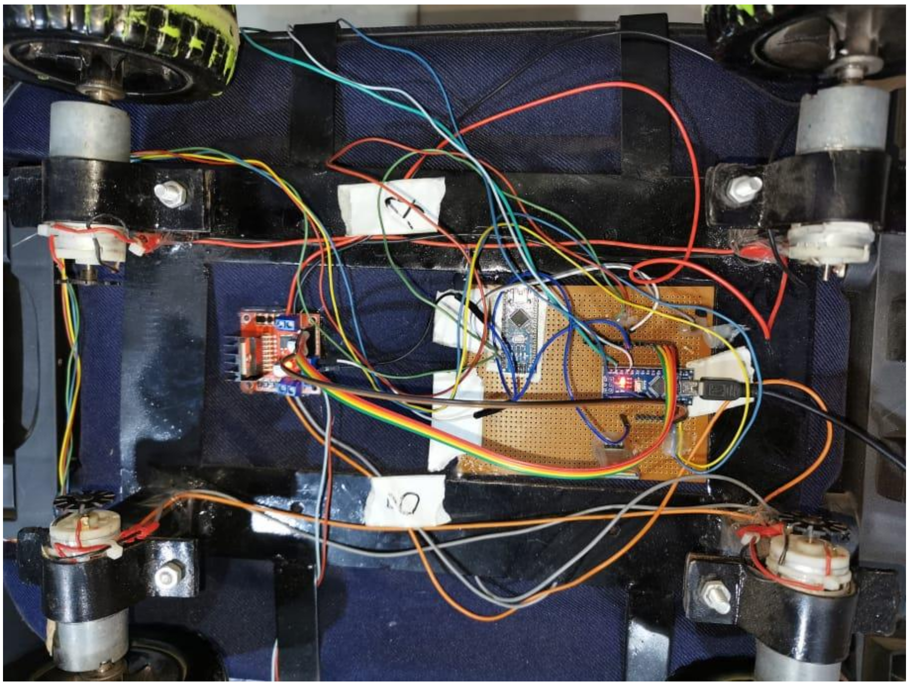

Figure 4. 2 Component / Hardware integration on AS

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The Autonomous Suitcase, which comprises IoT solution to the hassle problem of lifting the heavy bags, and the suitcases while traveling. Passengers will get accurate information about location using GPS technology by Text message using their GSM connection via SIM inserted in the GPS GSM module. The location will help them to track the Luggage in case of theft or losing the suitcase in any case. The other sensory operations will allow the suitcase to follow the passenger wherever he / she might go avoiding the obstacles. More over the Mobile application will allow the passenger to connect to the suitcase via Bluetooth and using some limited functionalities. The AS can be useful in many ways hence it has a large room for research and development, which hence in the form of product could provide benefit to the passengers.

5.2 Future Recommendations

In Autonomous Suitcase there is large room for advancement hence improving its functionalities and efficiency the usage of better sensors like “LIDAR” for following the Passengers / owners, efficient and accurate GPS Location sensing technologies, less power and resource consuming hardware, light and portable batteries and the mobile Application containing the aesthetic controls and other essential features flawlessly. The mentioned future work and extensive research and development will create more and more ease for the traveling passengers, which can travel while having their hands free.

CHAPTER 6

DATA SHEET

6.1 Arduino NANO

Arduino Nano is a microcontroller board designed by Arduino.cc. The microcontroller used in the Arduino Nano is Atmega328, the same one as used in Arduino UNO. It has a wide range of applications and is a major microcontroller board because of its small size and flexibility [17].

6.1.1 Summary

Table 6. 1 The below table illustrates the Arduino summary

#	Pin No.	Name	Type	Description
1	1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
2	3, 28	RESET	Input	Reset (active low)
3	4, 29	GND	PWR	Supply ground
4	17	3V3	Output	+3.3V output (from FTDI)
5	18	AREF	Input	ADC reference

6	19-26	A7-A0	Input	Analog input channel 0 to 7
7	27	+5V	Output or Input	+5V output (from on-board regulator) or +5V (input from external power supply)
8	30	VIN	PWR	Supply voltage

6.1.2 Features

Here are few of its basic features, which you must know if you are thinking to work on this great microcontroller board:

- It has 22 input/output pins in total.
- Fourteen of these pins are digital pins.
- Arduino Nano has eight analogue pins.
- It has six PWM pins among the digital pins.
- It has a crystal oscillator of 16MHz.
- Its operating voltage varies from 5V to 12V.
- It also supports different ways of communication, which are:
    - Serial Protocol.
    - I2C Protocol.
    - SPI Protocol.
- It also has a mini USB Pin that is use to upload code.
- It also has a Reset button on it.

6.1.3 Arduino Nano Bill of Material

Item Number	Qty.	Ref. Dest.	Description	Mfg. P/N	MFG	Vendor P/N	Vendor
1	5	C1,C3,C4,C7,C9	Capacitor, 0.1uF 50V 10% Ceramic X7R 0805	C0805C104K5RACTU	Kemet	80-C0805C104K5R	Mouser
2	3	C2,C8,C10	Capacitor, 4.7uF 10V 10% Tantalum Case A	T491A475K010AT	Kemet	80-T491A475K010	Mouser
3	2	C5,C6	Capacitor, 18pF 50V 5% Ceramic NOP/COG 0805	C0805C180J5GACTU	Kemet	80-C0805C180J5G	Mouser
4	1	D1	Diode, Schottky 0.5A 20V	MBR0520LT1G	ONSemi	863-MBR0520LT1G	Mouser
5	1	J1,J2	Headers, 36PS 1 Row	68000-136HLF	FCI	649-68000-136HLF	Mouser
6	1	J4	Connector, Mini-B Recept Rt. Angle	67503-1020	Molex	538-67503-1020	Mouser
7	1	J5	Headers, 72PS 2 Rows	67996-272HLF	FCI	649-67996-272HLF	Mouser
8	1	LD1	LED, Super Bright RED 100mcd 640nm 120degree 0805	APT2012SRCPRV	Kingbright	604-APT2012SRCPRV	Mouser
9	1	LD2	LED, Super Bright GREEN 50mcd 570nm 110degree 0805	APHCM2012CGCK-F01	Kingbright	604-APHCM2012CGCK	Mouser
10	1	LD3	LED, Super Bright ORANGE 160mcd 601nm 110degree 0805	APHCM2012SECK-F01	Kingbright	04-APHCM2012SECK	Mouser
11	1	LD4	LED, Super Bright BLUE 80mcd 470nm 110degree 0805	LTST-C170TBKT	Lite-On Inc	160-1579-1-ND	Digikey
12	1	R1	Resistor Pack, 1K +/-5% 62.5mW 4RES SMD	YC164-JR-071KL	Yageo	YC164J-1.0KCT-ND	Digikey
13	1	R2	Resistor Pack, 680 +/-5% 62.5mW 4RES SMD	YC164-JR-07680RL	Yageo	YC164J-680CT-ND	Digikey
14	1	SW1	Switch, Momentary Tact SPST 150gf 3.0x2.5mm	B3U-1000P	Omron	SW1020CT-ND	Digikey
15	1	U1	IC, Microcontroller RISC 16kB Flash, 0.5kB EEPROM, 23 I/O Pins	ATmega168-20AU	Atmel	556-ATMEGA168-20AU	Mouser
16	1	U2	IC, USB to SERIAL UART 28 Pins SSOP	FT232RL	FTDI	895-FT232RL	Mouser
17	1	U3	IC, Voltage regulator 5V, 500mA SOT-223	UA78M05CDCYRG3	TI	595-UA78M05CDCYRG3	Mouser
18	1	Y1	Cystal, 16MHz +/-20ppm HC-49/US Low Profile	ABL-16.000MHZ-B2	Abracon	815-ABL-16-B2	Mouser

Figure 6. 1 The above figure of table illustrates the BoM of Arduino NANO

6.1.4 Layout

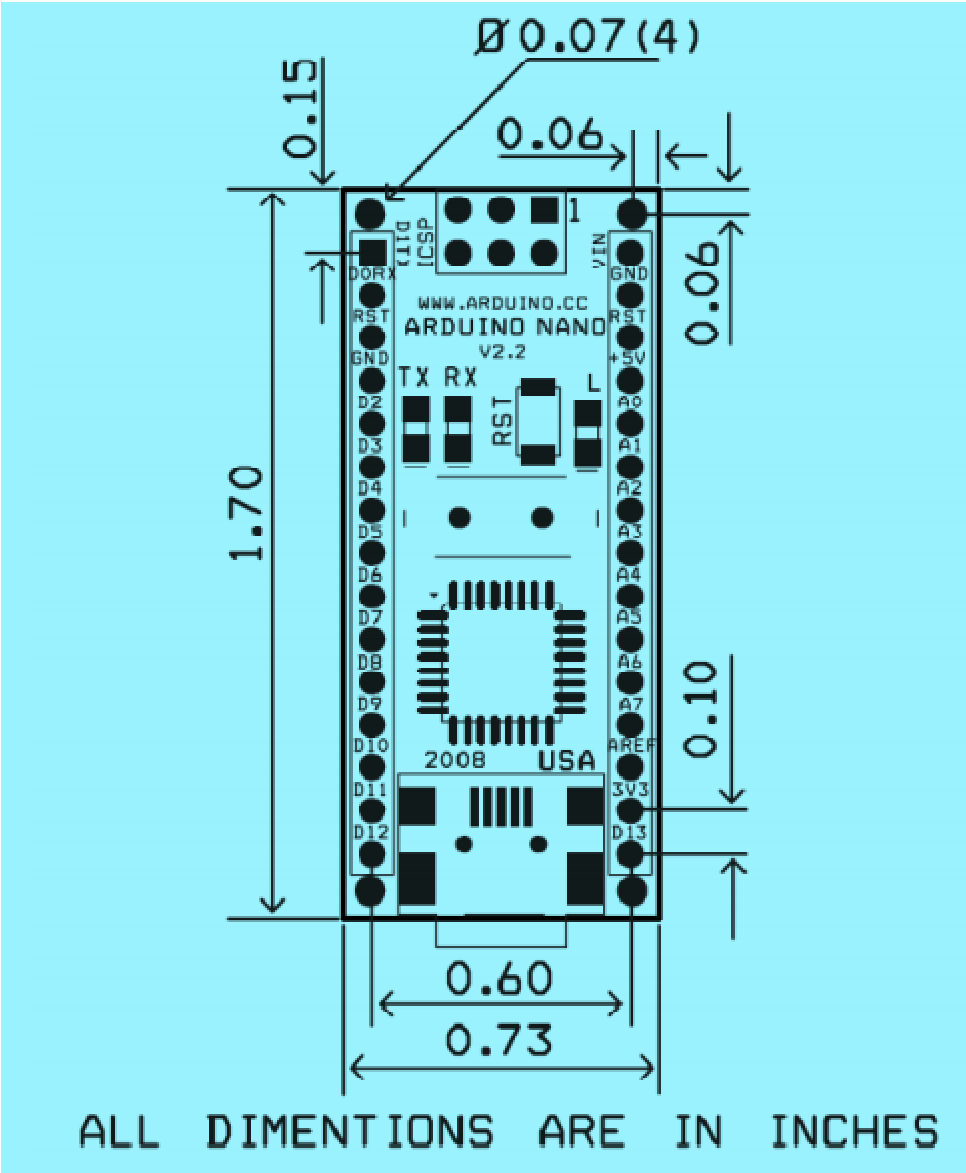

Figure 6. 2 Arduino Layout

6.2 GPS GSM Sim module (808 EVB v3.2) GPS + GPRS

This GPRS GSM GPS Shield based on the SIM808 GSM GPRS GPS all-in-one cellular phone module. You can add location-tracking, voice, text, SMS and data to your project. This shield fits right over your Arduino or Maduino, it is easy to use [18].

This shield fits right over your Arduino or compatible. At the heart is a powerful GSM cellular module (we use the latest SIM808) with integrated GPS.

6.2.1 Summary

Table 6. 2 The below table provides the summary for the SIM 808 Module

#	Name	Description
1	Connectors	
	Headset jack	This is a 'standard' TRRS 3.5mm phone headset jack with stereo earphone and mono microphone. Any 'iPhone' or 'Android' compatible (but not iPhone original) should work.
	uFL GSM connector	This is the GSM Antenna connector. An antenna is required to use the SIM808 shield!
	uFL GPS connector	This is the GPS Antenna connector, an antenna is required if you want to get GPS readings!
	uFL BT connector	This is the Bluetooth Antenna connector, in this version; there no have the Bluetooth function.
2	SIM Connector	A 2G Mini SIM card is required to use the module. You can buy it in any cell phone shop. It must be a 2G GSM card.
3	VRTC Battery holder	Power supply for RTC.
4	LEDs	
	PWR	Power light
	1PPS	This is the 'pulse per second' output of the GPS.
	Status	Power on status. Lit when the Cell module is booted and running.
	NetLight	Network status. You can use this for checking the current state without sending an AT command.
	64ms on, 800ms off	The module is running but has not made connection to the cellular network yet.

	64ms on, 3 seconds off	The module has made contact with the cellular network and can send/receive voice and SMS.
	64ms on, 300ms off	The GPRS data connection you requested is active By watching the blinks you can get a visual feedback on what is going on.
5	POWER_KEY	Power on or power off the SIM808.
6	Pins usage on Arduino	
	D0	Used if you select hardware serial port to communicate with SIM808.
	D1	Used if you select hardware serial port to communicate with SIM808.
	D7	Used if you select software serial port to communicate with SIM808.
	D8	Used if you select software serial port to communicate with SIM808.
	D9	Used for software control the Power ON or Power OFF of the SIM808.

6.2.2 Features

Main features are:

- Quad-band 850/900/1800/1900MHz - connect onto any global GSM network with any 2G SIM (in the USA, T-Mobile is suggested)
- Fully-integrated GPS
- Make and receive voice calls using a headset or an external 32Ω speaker + electret microphone
- Send and receive SMS message
- AT command interface with "auto baud" detection
- Send and receive GPRS data (TCP/IP, HTTP, etc.)
- GPS L1 C/A code

- 22 tracking /66 acquisition channels
- Tracking: -165 dBm
- Cold starts : -148 dBm
- Time-To-First-Fix：Cold starts-32s (typ.)，Hot starts-1s (typ.)，Warm starts-5s (typ.)
- Accuracy: approx. 2.5 meters
- Dimensions: 68mm*53mm*23mm

6.2.3 Layout

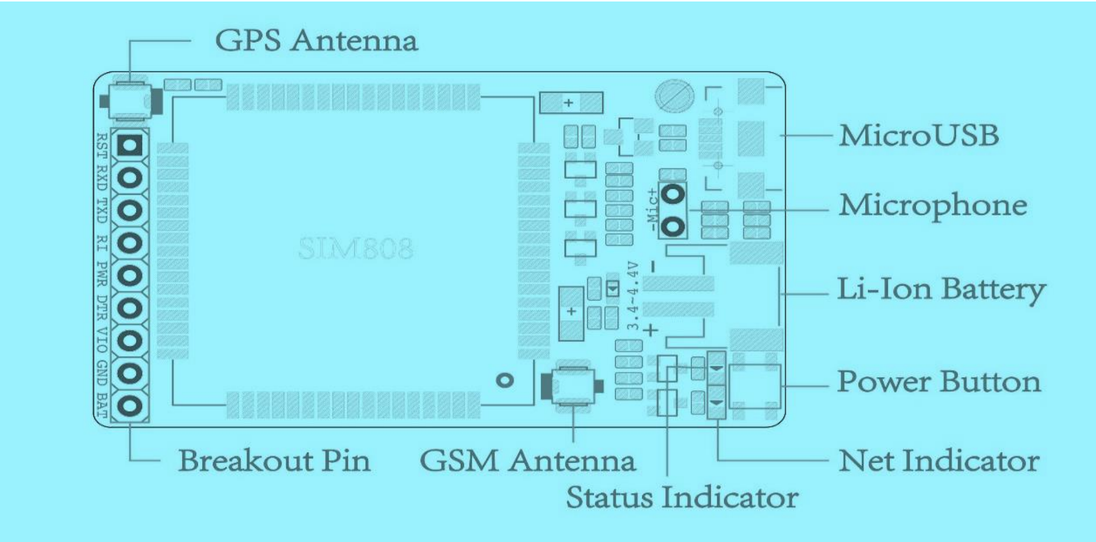

Figure 6. 3 The image illustrates the layout of SIM 808 Module

6.3 Motor Driver Module (L298N)

The L298N is an integrated monolithic circuit in a 15- lead Multi watt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver de-signed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provide to enable or disable the device independently of the input signals.

The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for [19].

6.3.1 Summary

Table 6. 3 This table contains the summary of L298N motor driver module

#	MW.15	Power S.	Name	Function
1	1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2	2;3	4;5	Out 1; Out 2	Outputs of the Bridge A the current that flows through the load connected between these two pins is monitored at pin 1.
4	4	6	Vs	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connect between this pin and ground.
5	5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6	6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
7	8	1,10,11,20	GND	Ground.
8	9	12	VSS	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground
9	10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B

10	13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15
11	-	3;18	N.C.	Not Connected

6.3.2 Features

The main features of the L298n Module are:

- High working voltage, can reach up to 46v.
- Large output current.
  - Instantaneous peak current can reach 3A.
  - Continuous working current can reach 2A.
- 25W Rated Power.
- High-Voltage and Current full-bridge driver with 2 H-bridges used to drive inductive loads like DC and Stepper Motors.
- Controlled with standard logic level signals.
- Two enable control terminals to enable or device without inputting signals.
- Able to drive a two-phase stepper motor, four-phase stepper motor or two DC motors.
- Has a high-capacity filter capacitor and freewheeling diode to protect devices from the reverse current of an inductive load.
- Built-in stabilivolt tube 78M05 can use to obtain 5v from the power supply.

6.3.3 Layout

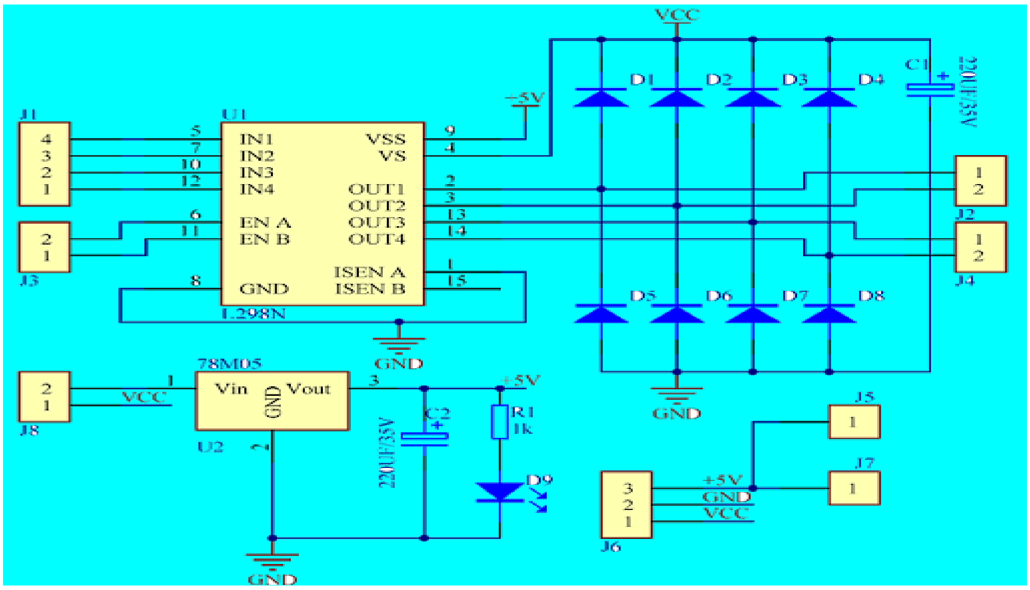

Figure 6. 4 This is the layout of L298N motor driver module

6.4 Ultrasonic Sensor Sr04

The transmitter emits eight bursts of a directional 40 KHz ultrasonic wave when triggered and starts a timer. Ultrasonic pulses travel outward until they encounter object. The object causes the wave to reflect back towards the unit.

The ultrasonic receiver would detect the reflected wave and stop the stop timer.

The velocity of the ultrasonic burst is 340m/sec. in air [20].

6.4.1 Summary

6.4.1.1 Module Pin Assignments

Table 6. 4 This illustrates the Configuration of Ultrasonic Sensor

#	Pin Symbol	Pin Description
1	VCC	5V power supply
2	Trig	Trigger Input pin
3	Echo	Receiver Output pin
4	GND	Power ground

6.4.1.2 Electrical Specifications

Table 6. 5 Following table illustrates the electrical parameters of Ultrasonic Sensor

#	Electrical Parameters	HC-SR04 Ultrasonic Module
1	Operating Voltage	5V
3	Operating Voltage	15mA
4	Operating Frequency	40KMz
5	Nearest Range	2cm
6	Measuring Angle	15 Degrees
7	Input Trigger Signal	10us min. TTL pulse
8	Output Echo Signal	TTL level signal, proportional to distance
9	Board Dimensions	1-13/16” X 13/16” X 5/8”
10	Board Connections	4 X 0.1” Pitch right angle header pins

6.4.2 Features

The main features are:

- Input Voltage: 5V

- Current Draw: 20mA (Max)
- Digital Output: 5V
- Digital Output: 0V (Low)
- Working Temperature: -15°C to 70°C
- Sensing Angle: 30° Cone
- Angle of Effect: 15° Cone
- Ultrasonic Frequency: 40kHz
- Range: 2cm - 400cm
- Dimensions
- Length: 43mm
    - Width: 20mm
    - Height (with transmitters): 15mm
    - Centre screw hole distance: 40mm x 15mm
    - Screw hole diameter: 1mm (M1)
    - Transmitter diameter: 8mm

6.4.3 Layout

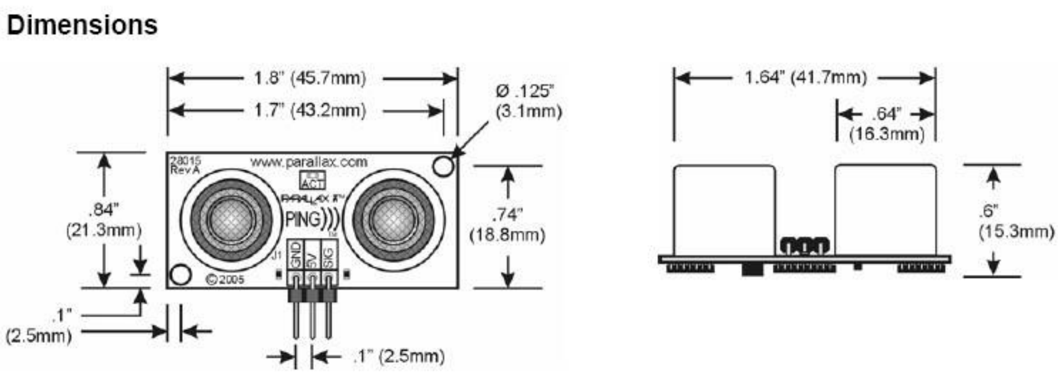

Figure 6. 5 Figure illustrates the Ultrasonic Sensor's layout

6.5 Bluetooth Module (Hc-05)

HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller or PC [21].

6.5.1 Summary

Table 6. 6 This is summary of Bluetooth module

#	Pin Name	Pin Description
1.	KEY/En	This pin is used to bring the Bluetooth module in AT commands mode. By default, this pin operates in data mode. The Key/EN pin should be high to operate Bluetooth in command mode. In HC-05, the default baud speed in command mode is 38400bps and 9600 in data mode.
2.	VCC	Used to power the Bluetooth module. Give 5V / 3.3 V to this Pin.
3.	GND	The ground pin of the module
4.	TXD	Connect this pin with the RXD pin of the Microcontroller. This pin transmits Serial data (wireless signals received by the Bluetooth module are converted by module and transmitted out serially on this pin)
5.	RXD	Connect this pin to the TXD pin of the Microcontroller. The HC-05 Bluetooth module receives the data from this pin and then transmits it wirelessly.

6.	STATE	It is use to check if the module is connected or not. It acts as a status indicator.

6.5.2 Features

Main features are enlist below:

- Bluetooth protocol: Bluetooth Specification v2.0+EDR (Enhanced Data Rate)
- Frequency: 2.4GHz ISM band
- Modulation: GFSK (Gaussian Frequency Shift Keying)
- Emission power: ≤4dBm, Class 2
- Sensitivity: ≤-84dBm at 0.1% BER
- Speed: Asynchronous communication: 2.1Mbps (Max) / 160 kbps, Synchronous communication: 1Mbps/1Mbps
- Security: Authentication and encryption
- Profiles: Bluetooth serial port
- Supply Voltage: +3.3V to 6.0 V
- Supply Current: 30mA
- Working temperature: -20 ~ +75Centigrade
- Dimension: 26.9mm x 13mm x 2.2 mm
- HC-05 Bluetooth module follows the IEEE 802.15.1 standardized protocol, through which one can build a wireless Personal Area Network (PAN). It uses frequency-hopping spread spectrum (FHSS) radio technology to send data over the air.

6.5.3 Layout

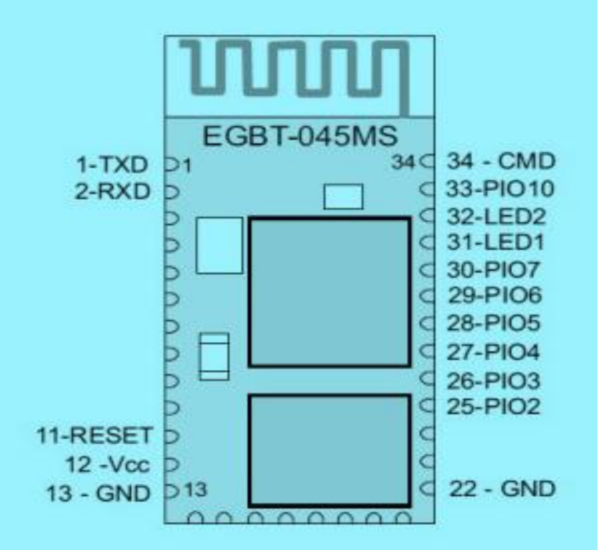

Figure 6. 6 The above-mentioned image contains layout of Bluetooth Sensor

6.6 Micro Servo 9g (sg90)

Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos [22].

6.6.1 Summary

Table 6. 7 The table contains the Servo summary

#	Wire Colour	Description
1	Brown	Ground wire connected to the ground of system
2	Red	Powers the motor typically +5V is used
3	Orange	PWM signal is given in through this wire to drive the motor

6.6.2 Features

The main Servo features are:

- Weight: 9 g.
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm.
- Operating speed: 0.1 s/60 degree.
- Operating voltage: 4.8 V (~5V)
- Dead bandwidth: 10 µs.
- Temperature range: zero ºC – 55 ºC.

6.6.3 Layout

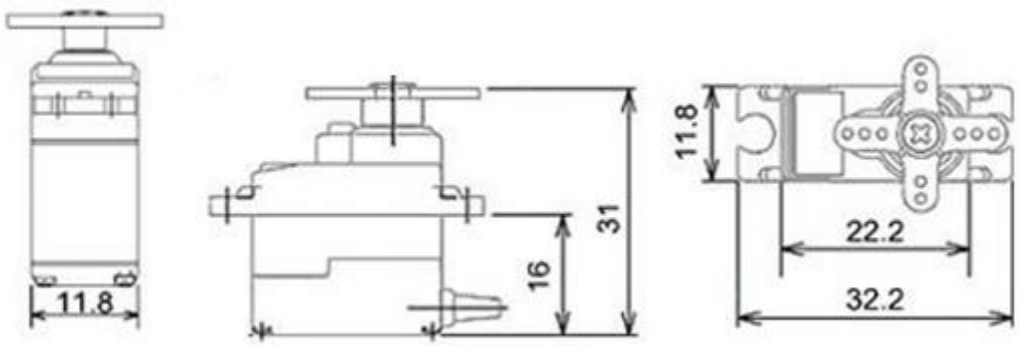

Figure 6. 7 Following is the layout of the servo Motor

6.7 DC Gear Motor (9V)

DC motor is the most commonly used motor on Make block platform. It is driven by direct current and is the ideal option to get things moving in all kinds of projects [23].

6.7.1 Summary

Table 6. 8 This is the Summary of DC Gear Motor

#	Name	Description
1	Voltage	9.0V
2	Ambient Temperature	0 ~+ 50 degree centigrade
4	Ambient Relative	60%±5%
5	Gross Weight	110g (3.88oz)
6	Package Content	1 x PH2.0-2P to Stripped-End Cable - 35cm 1 x DC Motor-25 9V

Comparison Chart

Model	SKU	Gear Ratio	Stall Torque	Stall Current	No Load		Load		
					Speed	Current	Torque	Speed	Current
700RPM	80032	1:10	1.5kg.cm	≤2.7A	700±10% rpm	≤150mA	0.3kg.cm	580±10rpm	≤600mA
325RPM	80035	1:20	4.5±kg.cm	≤2.7A	325±10% rpm	≤150mA	1.0kg.cm	280±10rpm	≤600mA
185RPM	80038	1:45	5.8 min	/	185±10% rpm	≤220mA	1.5kg.cm	150±10rpm	≤1.1A
86RPM	80041	1:75	11kg.cm	≤2.7A	86±10% rpm	≤150mA	3.5kg.cm	75±10rpm	≤550mA
30RPM	80044	1:217	/	≤2.7A	30±10% rpm	≤150mA	8.0kg.cm	26±10rpm	≤500mA
16RPM	80047	1:217	14.0 kg.cm	≤0.7A	16±10% rpm	≤100mA	14.0 kg.cm	12±10rpm	≤250mA

Figure 6. 8 The above image compares the aspects of DC motor

6.7.2 Features

The main features of the Gear Motor are:

- Voltage: 9V.
- Speed at no load: 190 RPM.
- Current at no load: 0.05 A.
- Speed max. efficiency: 147 RPM.
- Current max. efficiency: 0.125 A.
- Torque max. efficiency: 0.0122 NM.
- Current at stall: 0.35 A.
- Torque at stall: 0.0784 NM.

6.7.3 Layout

Following is the layout plan for dc gear motor

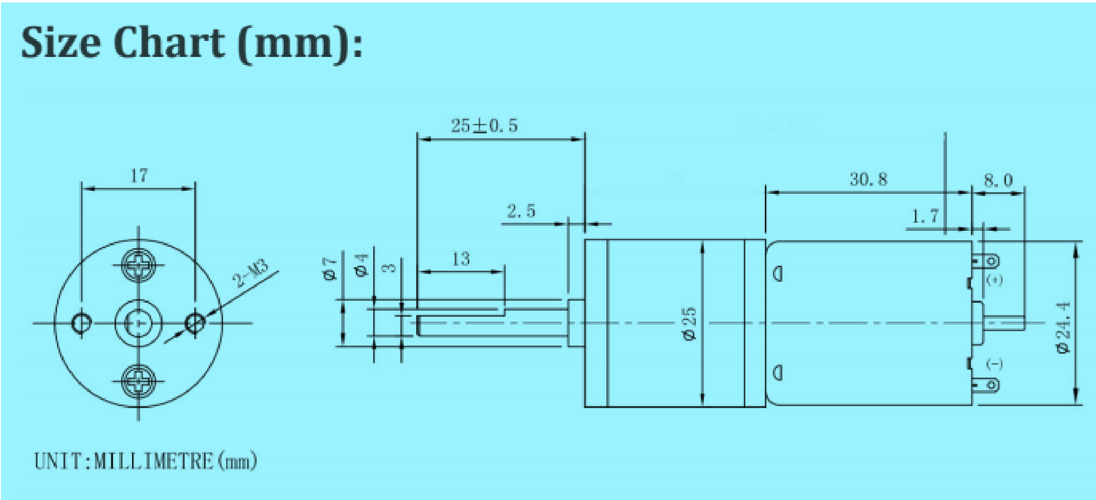

Figure 6. 9 The above image illustrates the motors layout

REFERENCES

[1] S. M. H. Khorassani, M. T. Maghsoodlou, N. Hazeri, M. Nassiri, G. Marandi, and A. G. Shahzadeh, "A facile synthesis of stable phosphorus ylides derived from harmin, harman, and carbazole," *Phosphorus, Sulfur Silicon Relat. Elem.*, vol. 181, no. 3, pp. 567–572, 2006, doi: 10.1080/10426500500269190.

[2] A. D'Ausilio, "Arduino: A low-cost multipurpose lab equipment," *Behav. Res. Methods*, vol. 44, no. 2, pp. 305–313, 2012, doi: 10.3758/s13428-011-0163-z.

[3] D. S. Vidhya Delicia Perlin Rebelo, M. D. Cecilia Jane, S. Linford William Fernandes, and M. Clarissa Joella Costa, "Obstacle Detection using Ultrasonic Sensors," *IJIRST-International J. Innov. Res. Sci. Technol.*, vol. 2, no. 11, pp. 316–320, 2016, [Online]. Available: www.ijirst.org.

[4] M. Geetha, "IoT Based Car Parking using Arduino and Blynk Application," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 8, no. 6, pp. 737–739, 2020, doi: 10.22214/ijraset.2020.6119.

[5] T. Sugimoto and H. Tomooka, "Passive-type infrared sensor system for detecting human body," no. 19, 1997, [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/0375650585900112.

[6] M. Todica, "Controlling Arduino Board with Smartphone and Blynk via Internet," *Tech. Doc.*, no. November, 2016, doi: 10.13140/RG.2.2.23956.30080.

[7] K. Maurya, M. Singh, and N. Jain, "Real Time Vehicle Tracking System using GSM and GPS Technology-An Anti-theft Tracking System," *Int. J. Electron. Comput. Sci. Eng.*, vol. 2, no. 3, pp. 1103–1107, 2012, [Online]. Available: http://www.estdl.org/wp-content/uploads/2012/08/Volume-1Number-3PP-1103-1107.pdf.

[8] M. L. Heilig, "United States Patent Office," *ACM SIGGRAPH Comput. Graph.*, vol. 28, no. 2, pp. 131–134, 1994, doi: 10.1145/178951.178972.

[9] P. E. S. Wong, "United States Patent (19)," no. 19, 1998.

[10] H. S. Doshi, M. S. Shah, and U. S. A. Shaikh, "INTERNET of THINGS (IoT): INTEGRATION of BLYNK for DOMESTIC USABILITY," *Vishwakarma J. Eng. Res.*,

2017.

[11] B. Bohara, S. Maharjan, and B. R. Shrestha, "IoT based smart home using blynk framework," *arXiv*. 2020.

[12] H. Durani, M. Sheth, M. Vaghasia, and S. Kotech, "Smart Automated Home Application using IoT with Blynk App," in *Proceedings of the International Conference on Inventive Communication and Computational Technologies, ICICCT 2018*, 2018, doi: 10.1109/ICICCT.2018.8473224.

[13] E. Media's, . S., and M. Rif'an, "Internet of Things (IoT): BLYNK Framework for Smart Home," *KnE Soc. Sci.*, 2019, doi: 10.18502/kss.v3i12.4128.

[14] P. Serikul, N. Nakpong, and N. Nakjuatong, "Smart Farm Monitoring via the Blynk IoT Platform : Case Study: Humidity Monitoring and Data Recording," in *International Conference on ICT and Knowledge Engineering*, 2019, doi: 10.1109/ICTKE.2018.8612441.

[15] P. Serikul, N. Nakpong, and N. Nakjuatong, "Smart Farm Monitoring via the Blynk IoT Platform," *2018 Sixt. Int. Conf. ICT Knowl. Eng.*, 2018.

[16] J. M. S. Waworundeng, N. C. Suseno, and R. R. Y. Manaha, "Automatic Watering System for Plants with IoT Monitoring and Notification," *CogITo Smart J.*, 2019, doi: 10.31154/cogito.v4i2.138.316-326.

[17] U. Manual, "Arduino Nano V2.3 User Manual," *Arduino*, pp. 1–5, 2008.

[18] "SIM808_Hardware Design_V1.00", 2013.

[19] A. Bell, "Datashet Module L298," *Rep. Ser. / Geol. Surv. Irel. ; RS 92/2*, p. 13, 2012.

[20] M. Principles, "HC-SR04 User Guide," pp. 8–11, 2016.

[21] Robotic Solutions, "HC-05 Bluetooth Module User 's Manual V1 . 0," *User Man.*, 2015.

[22] TowerPro, "TowerPro SG90 Micro Servo," *Cytron Technol.*, pp. 3–5, 2020.

[23] T. Tseng and M. Resnick, "Building examples," p. 176, 2012, doi: 10.1145/2307096.2307119.

APPENDICES

APPENDIX A: Arduino Nano

This represents the global variables and the serial pins for Arduino NANO of Bluetooth. It also contains the authentication token I for the connection of Blynk app.

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(2,3); // RX, TX
int speed_=120;
const int trigPin = 12;
const int echoPin = 11;
#define BLYNK_PRINT Serial
#include <BlynkSimpleSerialBLE.h>
int button = 6;
int led=13;
int ledState = HIGH;
// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
//char auth[] = "CIUKbCgkKyMc2aE0kfD5A4Ba9efIv7Uw";
char auth[] = "GAXZwLLDDIfiUZcudnS-RQV6SLgdAlIJ";//
```

The below image of code represents the pin configuration so motor driver module in coincidence with motors

```
int enA = 9;
int enB = 10;
int leftRev = A1;
int leftFwd = A2;
int rightRev = A3;
int rightFwd = A4;
//const int trigPin2 = A6;
//const int echoPin2 = A7;
// defines variables
long duration;
int distance;
//long duration2;
//int distance2;

float sensorValue1 = 0;
float sensorValue2 = 0;
float voltageValue = 0;
float currentValue = 0;


char val;
```

Follwing image represents the button switch and allert parametre for the bluetooth moduel disconnectivity.

```
Blynk.notify("aleart");

}
BLYNK_WRITE(V2)
{
  int pinValue = param.asInt(); // assigning incoming value from pin V1 to a variable
 if( pinValue == 1 ){
  ledState = !ledState;
  digitalWrite(led,ledState);
}
  // You can also use:
  // String i = param.asStr();
  // double d = param.asDouble();
  Serial.print("V2 Slider value is: ");
  Serial.println(pinValue);

}
```

The following is the main setup function for the overall project

```
void setup()
{
pinMode(led, INPUT);
  pinMode(button,INPUT_PULLUP);
  timer.setInterval(1000L, notifyonfire);
  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);

//  pinMode(trigPin2, OUTPUT);
//  pinMode(echoPin2, INPUT);

 pinMode(leftRev,OUTPUT);
 pinMode(leftFwd,OUTPUT);
 pinMode(rightRev,OUTPUT);
 pinMode(rightFwd,OUTPUT);

  Serial.begin(115200);

  Serial.println("START");
  mySerial.begin(9600);
//  mySerial.println("Hello");
  SerialBLE.begin(9600);
  Blynk.begin(SerialBLE, auth);

//  Serial.println("Waiting for connections...");

  // This will print Blynk Software version to the Terminal Widget when
  // your hardware gets connected to Blynk Server
  terminal.println(F("Blynk v" BLYNK_VERSION));
  terminal.println(F("-------------"));
//  terminal.println(F("Type 'Marco' and get a reply, or type"));
//  terminal.println(F("anything else and get it printed back."));
```

This piece of code contains loop for alert function and the logic function for the Autonomous Suitcase following the passenger

```
void loop()
{
Blynk.run();
if(!SerialBLE){
Blynk.notify("Alert");
  }

digitalWrite(trigPin, LOW);
delayMicroseconds(2);
// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echoPin, HIGH);
// Calculating the distance
distance= duration*0.034/2;
// Prints the distance on the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" ");


if(distance >= 50){

Farward();
}
if(distance >= 25 && distance <= 45 ){
Stop();
```

APPENDIX B: 808 Sim Module (Code)

The following piece of code is for the other microcontroller, which is reserved for providing location information using GPS SIM module

```
#include <SoftwareSerial.h>
SoftwareSerial sim808(11,10);//2,3

char phone_no[] = "+923064321824";
// replace with your phone no.
String data[5];
#define DEBUG true
String state,timegps,latitude,longitude;

void setup() {
sim808.begin(9600);
Serial.begin(9600);
delay(50);

sim808.print("AT+CSMP=17,167,0,0");
// set this parameter if empty SMS received
delay(100);
sim808.print("AT+CMGF=1\r");
delay(400);



void loop() {
sendTabData("AT+CGNSINF",1000,DEBUG);
if (state !=0) {
Serial.println("State  :"+state);
Serial.println("Time  :"+timegps);
Serial.println("Latitude  :"+latitude);
Serial.println("Longitude  :"+longitude);

sim808.print("AT+CMGS=\"");
sim808.print(phone_no);
sim808.println("\"");
```

```
void sendTabData(String command , const int timeout , boolean debug){

sim808.println(command);
long int time = millis();
int i = 0;

while((time+timeout) > millis()){
while(sim808.available()){
char c = sim808.read();
if (c != ',') {
data[i] +=c;
delay(100);
} else {
i++;
}
```