**SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103**
**(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)**

**Mini Project Report**
**on**

# “Optimisation of Luggage Tracking System in Airport”

submitted in partial fulfillment of the requirement for the completion of VI semester of

**BACHELOR OF ENGINEERING**
in
**ELECTRONICS AND COMMUNICATION ENGINEERING**

**submitted by**

| | |
|---|---|
| Brunda U S | 1SI16EC016 |
| Harshita | 1SI16EC025 |
| Hrijusha Arbind | 1SI16EC029 |
| K Sandhya | 1SI16EC031 |

under the guidance of

**Mr. T R SATEESH KUMAR**
**Assistant Professor**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**
**2018-19**

**SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103**

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

# DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

## *CERTIFICATE*

Certified that the mini project work entitled **"Optimisation of Luggage Tracking System in Airport"** is a bonafide work carried out by **BRUNDA US (1SI16EC016), HARSHITA (1SI16EC025), HRIJUSHA ARBIND (1SI16EC029), K SANDHYA (1SI16EC031)** in partial fulfillment for the completion of VI Semester of Bachelor of Engineering in Electronics and Communication Engineering of Siddaganga Institute of Technology, an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year 2018-2019.

Mr. T R Sateesh Kumar
Assistant Professor,
Dept. of ECE,
SIT, Tumakuru-03

HOD
Dept. of ECE,
SIT, Tumakuru-03

**External Viva-Voce**

**Name of the Examiners:** **Signature of the Examiners:**

**1.**

**2.**

# ACKNOWLEDGEMENT

We offer our humble pranams at the lotus feet of His Holiness, **Dr. Sree Sree Sivakumara Swamigalu**, Founder President and His Holiness, **Sree Sree Siddalinga Swamigalu**, President, Sree Siddaganga Education Society, Sree Siddaganga Math for bestowing upon their blessings.

We deem it as a privilege to thank **Dr. M N Channabasappa**, Director, **Dr. Shivakumaraiah**, CEO and **Dr. K P Shivananda**, Principal, SIT, Tumakuru for fostering an excellent academic environment in this institution, which made this endeavour fruitful.

We would like to express our sincere gratitude to **Dr. K V Suresh**, Professor and Head, Department of ECE for his encouragement and valuable suggestions.

We thank our guide **Mr. T R Sateesh Kumar,** Assistant Professor, Department of Electronics & Communication Engineering, SIT, Tumakuru for the valuable guidance, advice and encouragement.

We would like to express thanks to supporting staff for their support in carrying out our project work successfully.

Above all, we would like to express thanks to our parents for their support all along.

| | |
|---|---|
| Brunda U S | 1SI16EC016 |
| Harshita | 1SI16EC025 |
| Hrijusha Arbind | 1SI16EC029 |
| K Sandhya | 1SI16EC031 |

| Course Articulation Matrix - Mini Project – 6ECMP | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **COs** | **Statement** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | Identify , formulate the problem and define the objectives | 3 | | | | | | | | | | | | **2** | |
| CO2 | Review the literature and provide efficient design solution with appropriate consideration for societal, health and safety issues | | **2** | | | | 1 | | | | | | | **2** | **1** |
| CO3 | Select the engineering tools/components and develop an experimental setup to validate the design | | | **2** | | **2** | | | | | | | **2** | **2** | **2** |
| CO4 | Test, analyse and interpret the results of the experiments in compliance with the defined objectives | | | | **2** | | | | | | | | | **2** | |
| CO5 | Document as per the standard, present effectively the work following professional ethics and interact with target group | | | | | | | | **2** | | **2** | | **2** | | |
| CO6 | Contribute to the team, lead the diverse team, demonstrating engineering and management principles | | | | | | | | | **2** | | **1** | | | **1** |
| **Average** | | **3** | **2** | **2** | **2** | **2** | **1** | | **2** | **2** | **2** | **1** | **2** | **2** | **2** |

# CONTENTS

# LIST OF FIGURES



# LIST OF TABLE

# Abstract

Complaints and concerns of air passengers about luggage mishandling show no signs of decline. Failure to load the bags onto the aircraft, tagging errors and technical issues were cited by sources in the airline industry as some of the key reasons for mishandled baggage.

The aim of the project is to solve this problem of luggage tracking in aviation industry by replacing the older tags with *Radio Frequency Identification (RFID)* tags. The designed system consists of two modules namely, the luggage tracking module and the RFID reader module. The RFID reader module comprises of NodeMCU interfaced with RFID reader, placed at the Cargo bay of the airplane and the luggage tracking module comprises of an RFID tag and NodeMCU interfaced with Neo-6M *Global Positioning System (GPS)* module, placed in the luggage of the passenger. The passenger will receive a notification when the luggage is boarded or deboarded onto the plane. Also, location of the luggage is sent to the cloud server. The location from the cloud can be retrieved on passengers' request through Android application developed on Android Studio using Java and *Extended markup language (Xml)* languages. Arduino *Integrated Development Environment (IDE)* is used for providing a programming environment to the NodeMCU on Windows operating system.

The user name and password for logging into the Android application will be given to the user during checking-in and the luggage tracking module will be placed in their luggage. It will be removed from the luggage at the destination. Thus, making it a one-time investment attending some major loopholes in the current system.

# CHAPTER 1

# INTRODUCTION

## 1.1 Motivation

The most common loopholes experienced in aviation industry for luggage handling are mislaid baggage, lost baggage and damage to the belongings. The current luggage tracking system depends entirely on barcodes and automatic 360-degree barcode reader mounted above conveyor belts. The issue with the current system is the readability of the barcode tag. Wrinkled tag is difficult to read and dirty barcode reader cannot consistently identify the labels, which leads to mishandling of luggage. Globally, 6.5 bags per thousand passengers got misplaced in 2015 according to an annual baggage report by Geneva-based technology service provider *Society International de Telecommunications Aéronautiques* (SITA) [1]. In order to overcome these problems, a system is designed which tracks the luggage boarding, deboarding and an Android application to know the exact GPS location of luggage on user demand [2].

The system designed ensures security for baggage since the owner can get the position of his/her luggage during the journey. The RFID tag is attached to the luggage of each passenger. RFID reader consists of the *Radio Frequency* (RF) module, control unit and coupling element to interrogate tag via RF communication. The tag is passive and operates in the *LHF* (up to 125KHz) range of frequency [3]. The tag powers itself from the radio waves received by the reader. Each tag has unique ID which is stored in database maintained by aviation authority along with the passenger details.

## 1.2 Objective of the project

The objective of the project is to design a system for tracking the luggage of the passenger throughout his/her travelling duration and to design an Android application through which the user can know about the location and status of the luggage.

## 1.3 Organization of the report

The report is divided into 7 chapters. Chapter 2 comprises of literature survey. Chapter 3 contains block diagram and the description of individual blocks. Chapter 4 comprises of hardware description, chapter 5 describes the software tools used. Chapter 6 includes the results. Finally, chapter 7 lists conclusions drawn from the project and future enhancements that can be made.

# CHAPTER 2

# LITERATURE REVIEW

The conventional luggage tracking system is centered around the low reliability rate of barcode tag. In this chapter, methods that are proposed in context to the luggage tracking system are discussed.

## 2.1 Baggage tracing and handling system using RFID and IoT for airports:

Ashwini Singh, Sakshi Meshram, Tanvi Gujar, P. R. Wankhede

This paper describes the design of baggage tracking and handling system using smart RFID tag and IoT which is based on cloud server. It proposes a prototype at two locations having both check-in and check-out processes. A more secured algorithm is used for generating tag that are attached to printed baggage label with the details of passenger and airline stored in it. RFID reader in the check-out areas facilitate step tracking of baggage which prevents baggage loss. The baggage's real time position is tracked and stored in a cloud using IoT and unique ID can be retrieved by the passengers wherever and whenever necessary. The same ID can be used while collecting bag at check-out counters. The system provided ensures less consumption of time, security for baggage and is economical hence provides customer satisfaction [4].

## 2.2 An interactive RFID-based bracelet for Airport luggage tracking system:

Aicha Slassi Sennou, Asmae Berrada, Yassine Salih-Alj, Nasser Assem

This paper investigates the use of an interactive bracelet that communicates with the RFID system by means of a database application. The database system interacts with the bracelet using messages that inform the passenger about his luggage status [5].

## 2.3 A data warehouse solution for analyzing RFID based baggage tracking data:

Ahmed, Tanvir, Torben Bach Pedersen, and Hua Lu

This paper presents a carefully designed *Data Warehouse (DW*), with a relational schema sitting underneath a multidimensional data cube, that can handle the many complexities in the data. The paper also discusses the *Extract-Transform-Load (ETL)* flow that loads the data warehouse with the appropriate tracking data from the data sources. The presented concepts are generalizable to other types of multi-site indoor tracking systems based on Bluetooth and RFID. The system has been tested with large amount of real-world RFID-based baggage tracking data from a major industry initiative [4].

It can be observed that in these existing projects, the location of the luggage is not known by the user. This is a disadvantage because the user will only know about the status of luggage whether it is boarded or deboarded. It does not ensure the safety of the luggage. This project aims at providing a secure and reliable system for luggage tracking at the Airport. It ensures that the passenger get information about the luggage location during their travel.

# CHAPTER 3

# SYSTEM BLOCK DIAGRAM

This chapter describes the block diagram of the designed system which is shown in Fig. 3.1.

## 3.1 Block diagram of the system

The system consists of two modules namely, *Luggage tracking module* placed in the luggage of the passenger to track the location of luggage and *RFID reader module* which is placed at Cargo bay to notify the user about the status of his/her luggage boarding and deboarding.

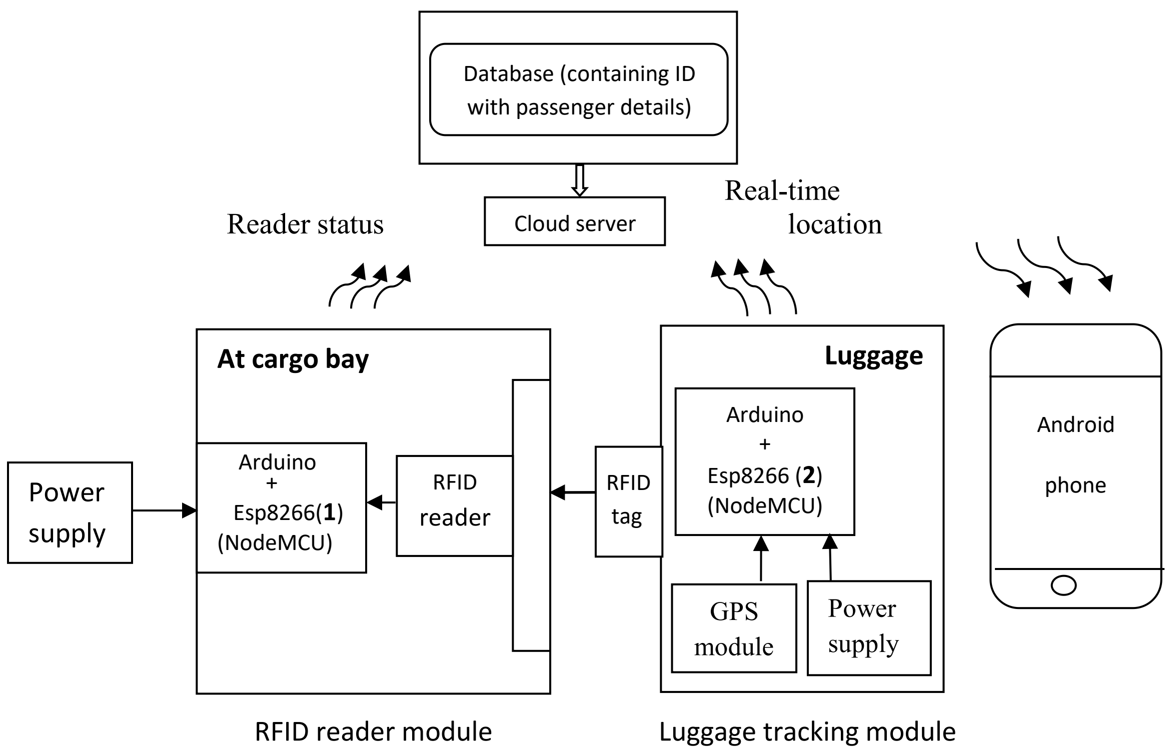



Fig. 3.1: Block diagram of the system

When the passenger gives their luggage for checking in, the luggage tracking module consisting of GPS module and NodeMCU (2) along with RFID tag is attached to the luggage. Also, the username and password is provided to user, through which they can login to the designed application.

The RFID reader interfaced with NodeMCU (1) is attached to the cargo bay. Whenever the luggage is scanned by the RFID reader, a notification is sent to user about his/her luggage boarding status. The location of the luggage is also sent to the user mobile. The status of boarding and location of luggage is sent to the firebase through inbuilt Wi-Fi.

# CHAPTER 4

## SYSTEM HARDWARE

This chapter explains the hardware implementation and the detailed description of the hardware components used in the system is shown in Fig. 4.1.

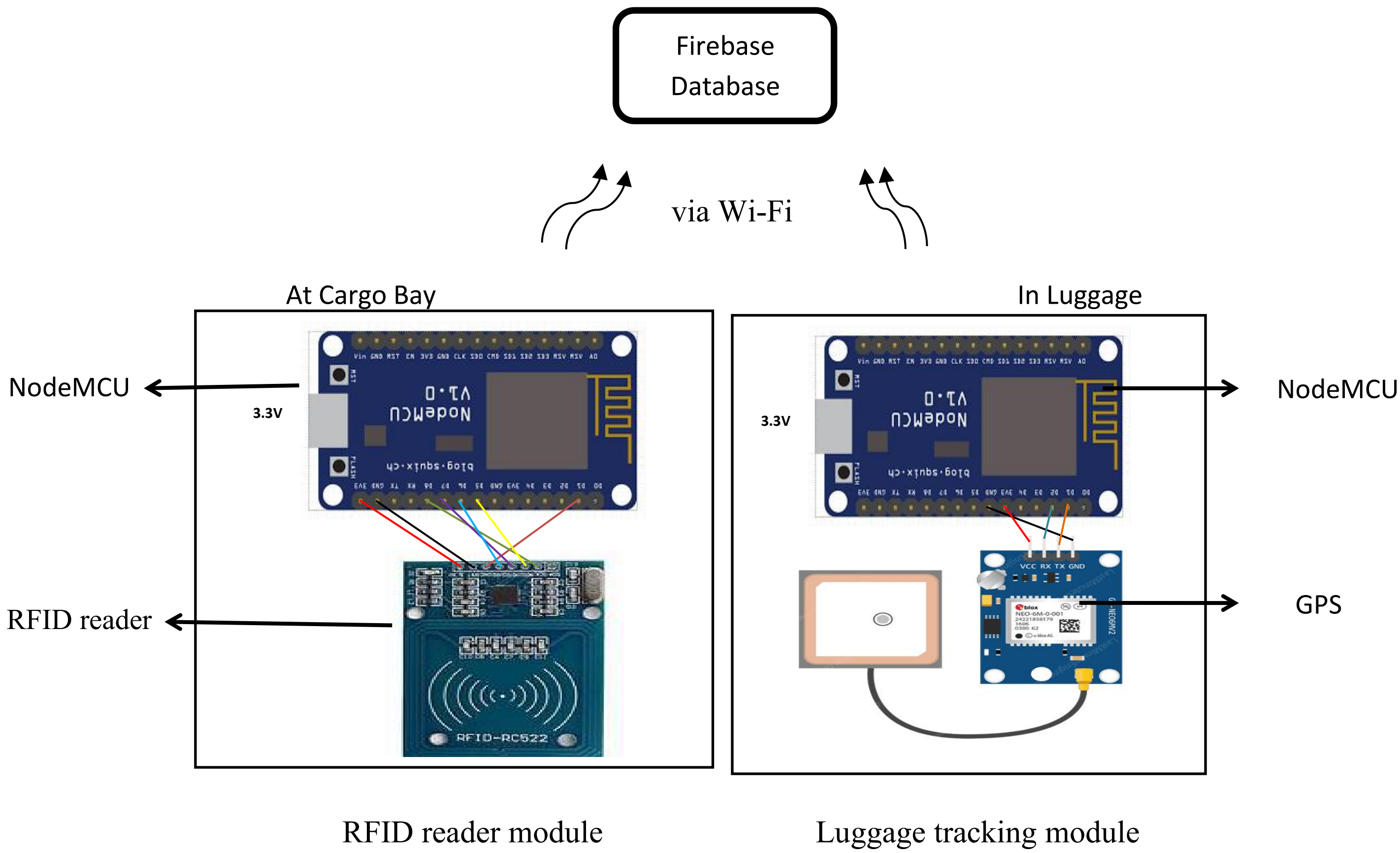



Fig. 4.1: Hardware implementation

### 4.1 NodeMCU

NodeMCU is a development board for ESP8266, which is Wi-Fi chip with 32bit microcontroller. It has inbuilt Wi-Fi such that it can be connected to internet without much effort compared to connecting Arduino UNO to internet. NodeMCU is Arduino compatible, hence it can be programmed in ‘C’ language using Arduino IDE directly. NodeMCU is an advanced application program interface for hardware IO, which can reduce the redundant work for configuring and manipulating hardware. The Development kit based on ESP8266, integrates GPIO, PWM, IIC, 1-Wire and ADC all in one board. The NodeMCU ESP-12E module is shown in Fig. 4.2.

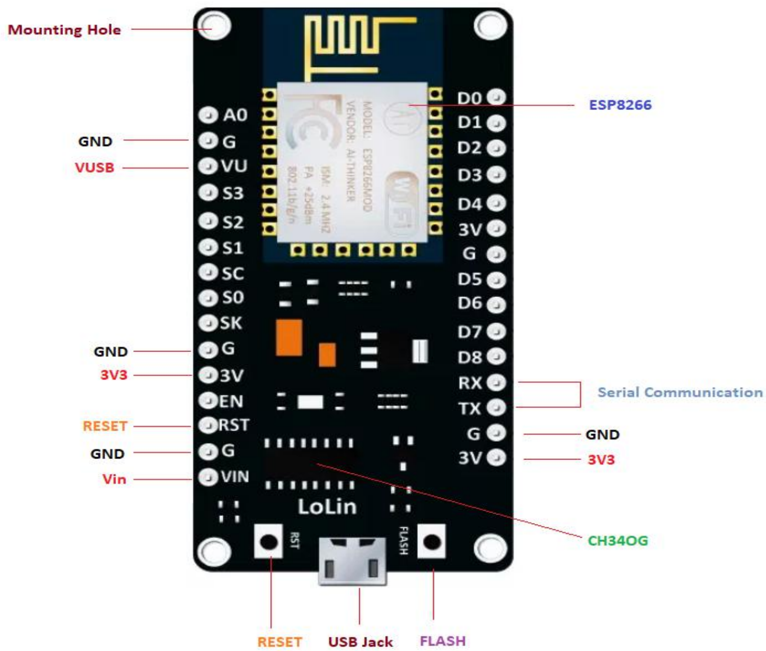
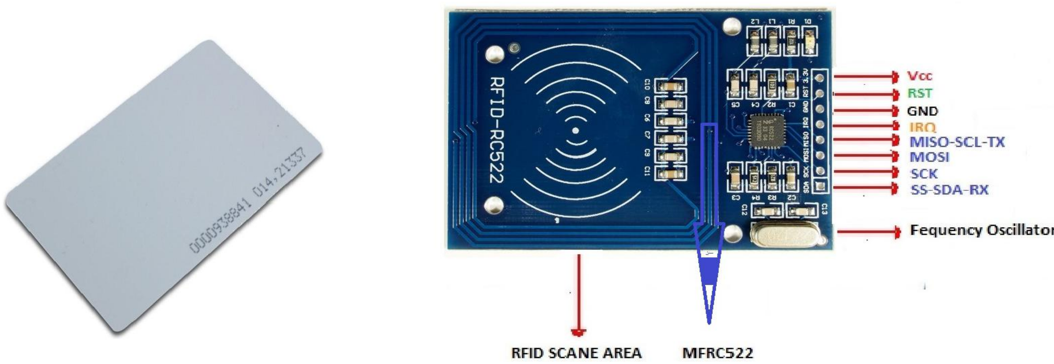



Fig. 4.2: NodeMCU ESP-12E module (courtesy: [7])

The RX and TX pin of the NodeMCU provides Serial Communication with a host PC from which complied object code is downloaded. NodeMCU is attached to the Luggage tracking module and RFID module.

## 4.2 RFID reader

The MFRC522 RFID reader shown in Fig.4.3 is a highly integrated reader/writer IC for contactless communication at 13.56 MHz. This low cost MFRC522 based RFID reader module is easy to use and can be used in a wide range of applications. RC522 supports ISO 14443A/MIFARE mode. The Reader module comes with an on-chip antenna and can be power up with a 3.3 V power supply. MOSI, MISO, CS and CLK are pins of SPI (*Serial Peripheral interface*) protocol. SPI uses these separate pins to communicate with the target device. If the tag is within the reading distance, it is read by the RFID reader and this information is send to the cloud via NodeMCU.




RFID Tag RFID reader

Fig. 4.3: RFID reader (courtesy: [8])

The interfacing of the RFID reader with NodeMCU is shown in Fig 4.4. The RST pin of the reader is connected to D1 pin of the NodeMCU. Reader communicates NodeMCU by connecting through SPI protocol which ensures good reading distance [10]. The VCC pin of the reader is connected to +3.3V and GND pin is given to ground pin of NodeMCU.

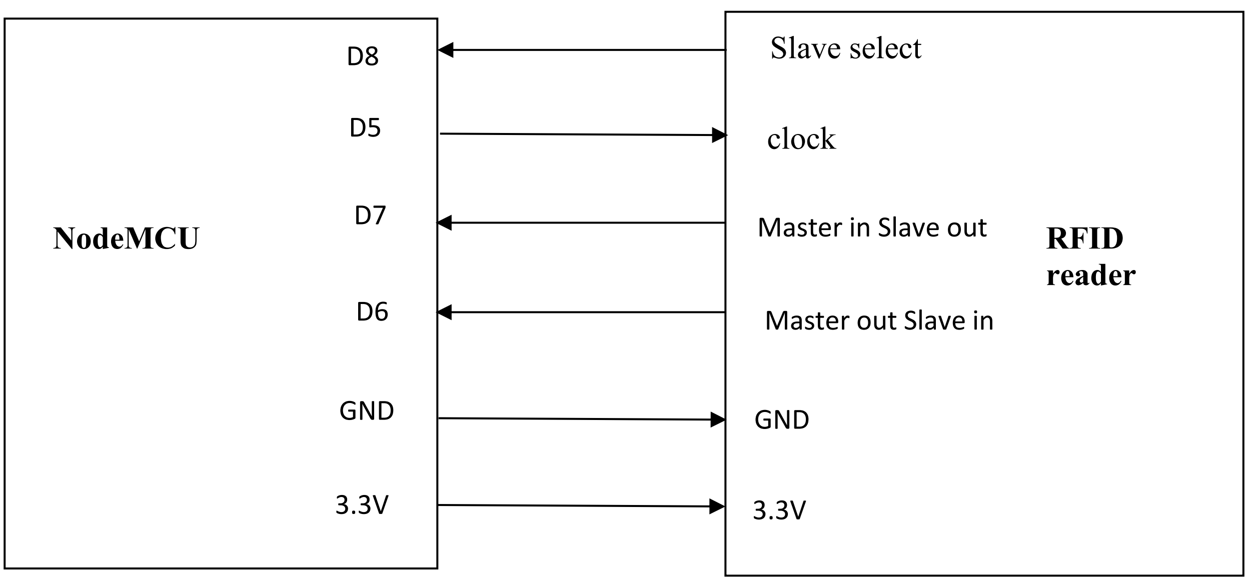



Fig. 4.4: Interfacing of reader with NodeMCU

Whenever the luggage is boarded to the Cargo bay, the RFID reader at the Cargo bay reads the tag attached to the luggage. The *Slave Select (SS)* pin of reader communicate with D8 pin of NodeMCU and send the information through *Master Out Slave in (MOSI)* pin of reader to D7 pin of NodeMCU. The information from NodeMCU is transmitted to database through inbuilt Wi-Fi.

## 4.3 GPS module

The NEO-6M GPS module is a well-performing complete GPS receiver with a built-in 25 X 25 X 4mm ceramic antenna, which provides a strong satellite search capability. The NEO-6M GPS module is shown in Fig. 4.5. This module interfaced with NodeMCU helps to get the current location of the luggage. It has a built-in EEPROM to save configuration parameter data, equipped with power and signal indicator lights and data backup battery.


Fig. 4.5: NEO-6M GPS module (courtesy: [9])

Two GPIO pins (D1 and D2) are fed to transmit and receive the signals from the GPS module. In the GPS module, VCC pin is connected to +3.3V and GND pin is connected to ground pin of the NodeMCU as shown in the Fig. 4.6.

Whenever GPS module receives the signal from satellite, it sends the received information from Transmitter pin of GPS Module to D1 pin of NodeMCU. The information from NodeMCU is then transmitted to Firebase through Wi-Fi.

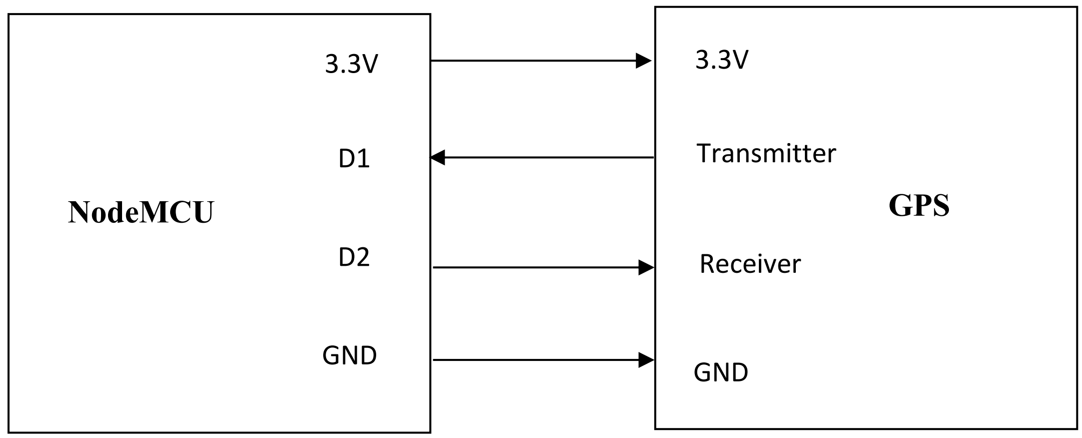



Fig. 4.6: Interfacing of GPS module with NodeMCU

# CHAPTER 5

# SOFTWARE DESCRIPTION

This chapter describes the software tools that are used in the project. It presents the cloud database storage and android app development involved.

## 5.1 Android Studio

Android Studio is an application development environment for android operating systems. The programming language used is Java and Xml. It has built-in support for google cloud platform enabling integration with google app engine and firebase cloud. Android app developing using Android Studio is shown in Fig. 5.1.

Android Studio is used to develop the Android application. The first page of the application is login page followed by a menu page having options to know the current location and status of boarding. The location information of the luggage and status of the reader will be stored in the Firebase database. The Android application retrieves the recent information from the database and provides the information to the user.

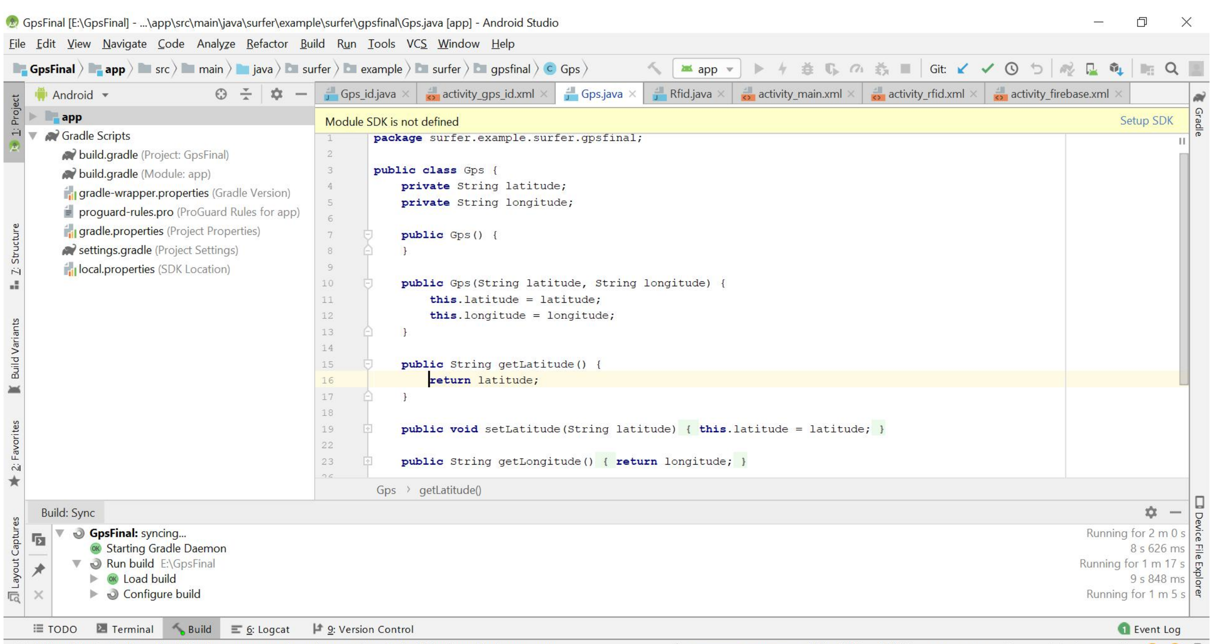


Fig. 5.1: Android Studio editor window

## 5.2 Arduino IDE

The Arduino Software IDE is shown in Fig. 5.2. It contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino to upload programs. Programs written using Arduino IDE are called Sketches and these sketches are saved with file extension (.info).

In this project Arduino IDE is used for programming the NodeMCU and Embedded C is for coding. Two sketches have been made: one is for the luggage tracking module that enables the NodeMCU to transmit the signal received from the GPS module to the Firebase through Wi-Fi, second is for the RFID reader module that enables the NodeMCU to transmit the RFID reader status to the Firebase through Wi-Fi.

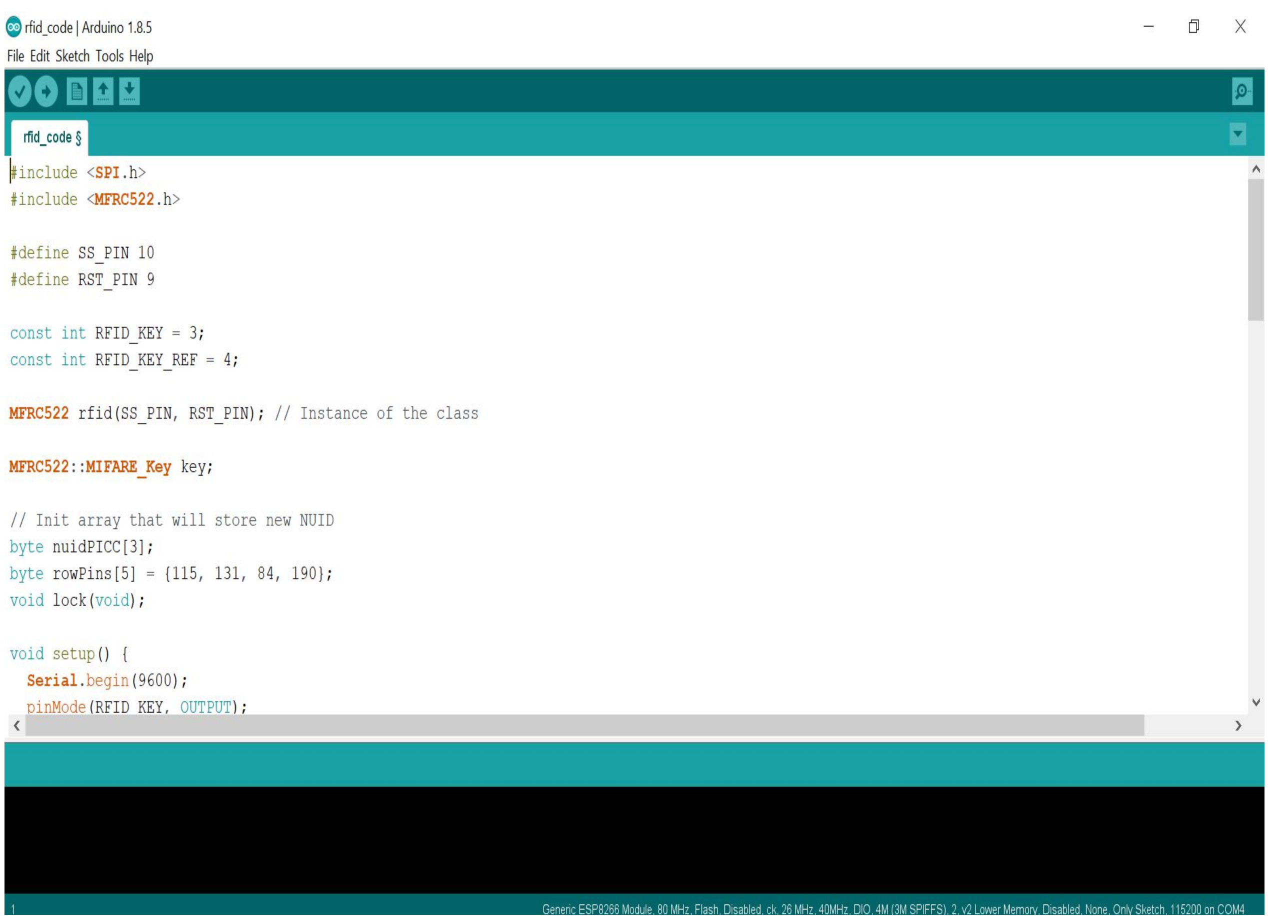



Fig. 5.2: Text editor window of Arduino IDE

## 5.3 Firebase database

The Firebase database is a cloud-hosted database. Data is stored as *Java Script Object Notation* (JSON) and synchronized to every connected client. The Firebase database is used to maintain database of passengers. The output of RFID reader and location of luggage is stored in Firebase, which is the backend of android application. Two ‘child’ of firebase database have been created, one for GPS location and other for luggage status.

When the RFID tag is scanned, the latitude and longitude of the luggage is stored in the Firebase database along with the status of boarding as shown in Fig. 5.3.

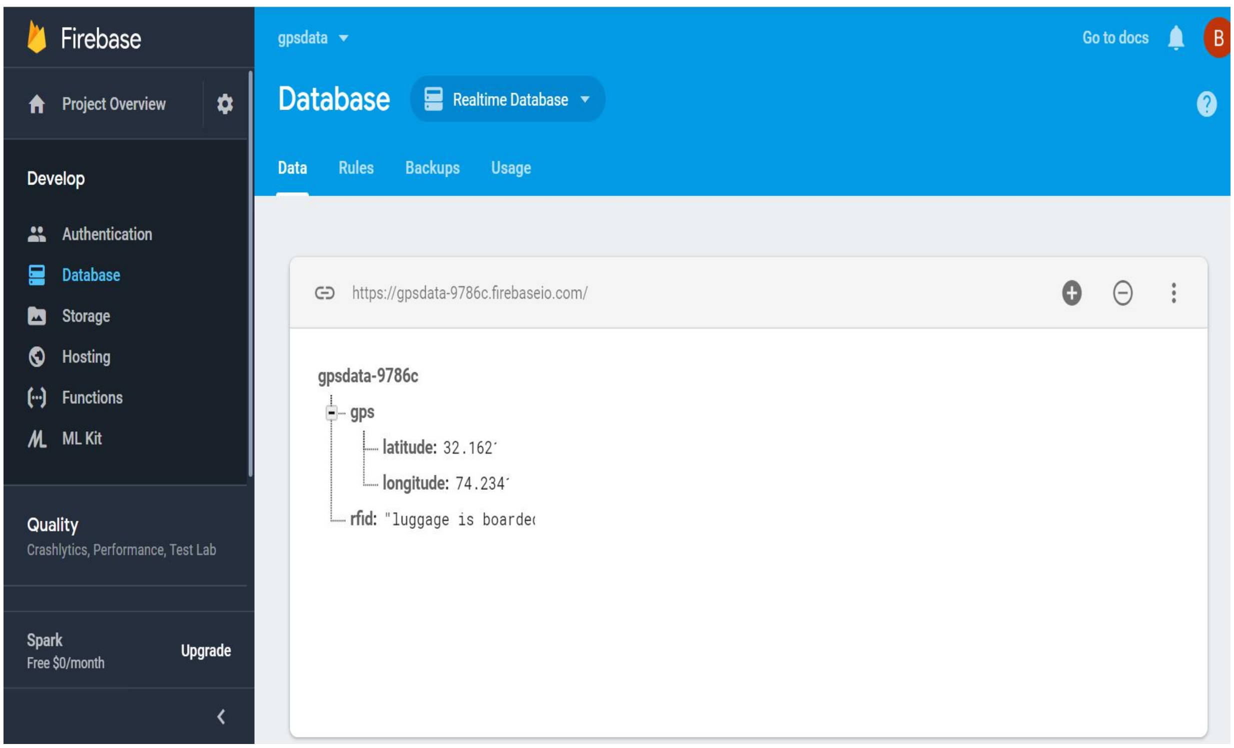



Fig. 5.3: Data stored in the Firebase console

## 5.4 Flow chart

Fig. 5.4 explains the algorithm when the RFID reader reads the RFID tag present in the luggage tracking module.

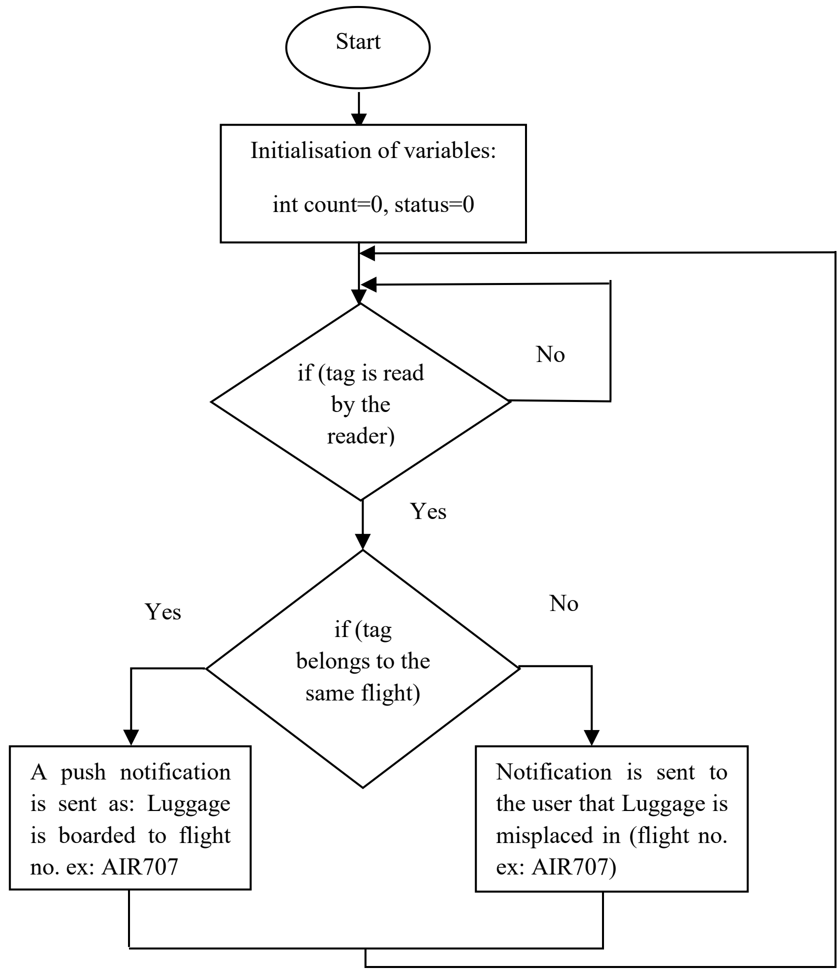



Fig. 5.4: Flowchart of RFID based luggage tracking system

Fig. 5.5 explains the algorithm for knowing the luggage location through an Android application.

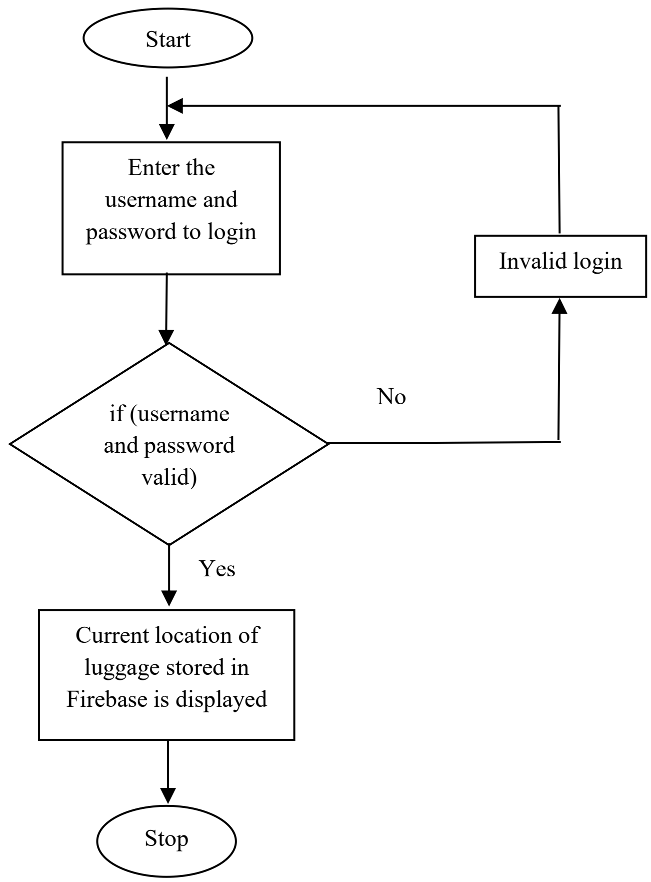



Fig. 5.5: Flowchart for displaying location of luggage in an Android application

# CHAPTER 6

# RESULTS

This chapter describes the results of the Luggage Tracking System in Airport. The Project provides a secure and reliable system for luggage tracking at the Airport. It ensures that the passenger get information about the luggage location throughout their travel. The designed system implementation is shown in Fig.6.1


Fig.6.1: Designed model

Fig.6.2 shows the snapshots from the developed android application of the login page and menu page of the tracking system. The login page is authenticated by the user identity with the password given by the authority at the time of boarding. Authority will maintain separate database of each user and passwords for ensuring the authentication. After Login, the menu page pops up, which shows two options to know the “Status of boarding” (result of RFID scanner) and “Tracking luggage location” (current location of the luggage).

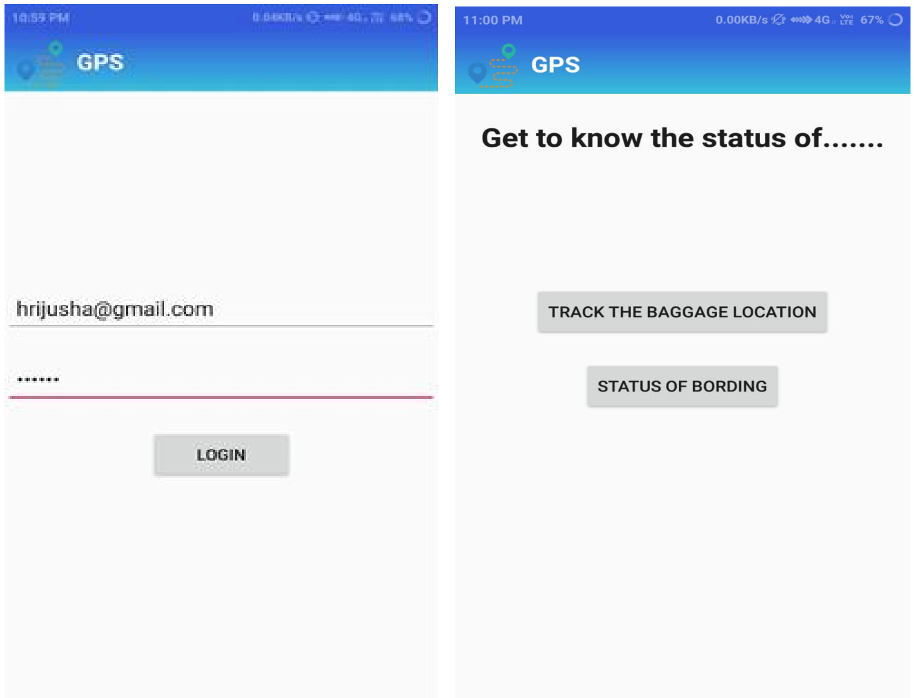



Fig. 6.2: Login and Menu page of Android application

"*STATUS OF BOARDING*" button in the application will enable the user to know whether the luggage is boarded to correct flight or not as shown in Fig. 6.3. If the luggage is placed in different flight, it will inform the user about the flight in which his/her luggage is boarded.

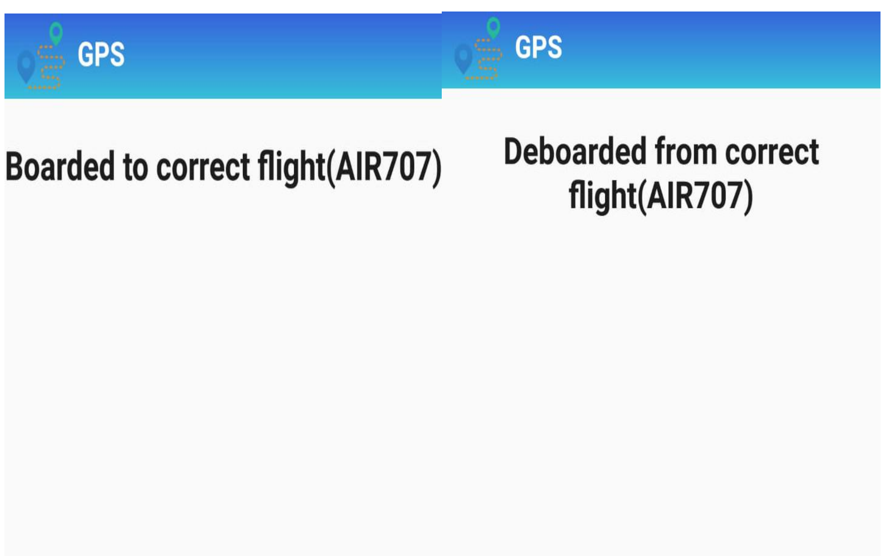



Fig. 6.3: Status of luggage at the user application

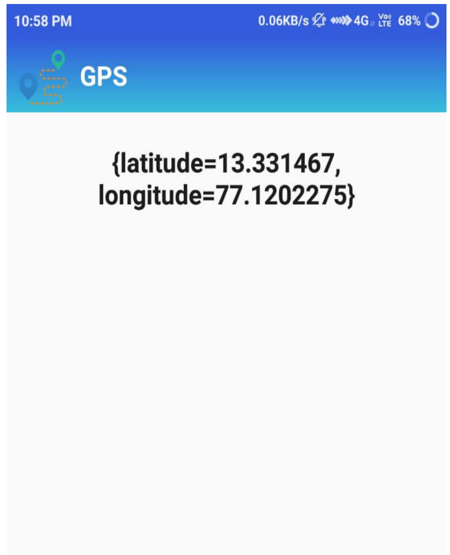



Fig. 6.4: Location of the luggage at the user mobile

"*TRACK THE BAGGAGE LOCATION*" button in the application will give the user the GPS location of his/her luggage by getting coordinates of latitude and longitude as shown in Fig. 6.4, thus making whole tracking procedure more effective.

# CHAPTER 7

# CONCLUSION AND FUTURE ENCHANCEMENT

This chapter includes conclusion and future improvements that can be incorporated in future to enhance the system performance.

## 7.1 Conclusion

Loss of luggage is one of the biggest areas of travel complaints. To save the passengers from this problem, a luggage tracking system with RFID tag and GPS module has been designed and tested. It ensures that the passenger knows about his/her luggage throughout the journey. The Android application makes the whole system more effective wherein the GPS location and the boarding status can be seen.

The system uses NodeMCU which is cost-effective, user compatible and has a built-in Wi-Fi which makes the connection to the internet easier. The low cost RFID reader module is highly integrated and good for contactless communication. Thereby, the system ensures safe handling of luggage.

## 7.2 Future Enhancement

The Project has scope for future enhancements in aviation industry which are mentioned as follows:

- ✓ The power requirement of GPS module can be reduced by activating the GPS module to send the signal information to database only on user request.
- ✓ Time of boarding and deboarding can be stored in database along with GPS location and flight number.

# ACTION PLAN

The Plan of Action is shown in Table 7.1.

| Activities | Oct 2018 | Nov 2018 | Dec 2018 | Jan 2019 | Feb 2019 | Mar 2019 | Apr 2019 |
|---|---|---|---|---|---|---|---|
| Problem Definition | √ | | | | | | |
| Literature review | √ | √ | | | | | |
| Design | | √ | | | | | |
| Documentation of phase 1 | | | √ | | | | |
| Testing | | | | √ | √ | | |
| Evaluation of results | | | | | | √ | |
| Documentation of phase 2 | | | | | | √ | |
| Report Writing | | | | | | √ | √ |

Table 7.1: Plan of Action

## REFERENCES

[1] From Hindustan Times, “Misplaced-luggage-tops-list-of-domestic-fliers-complaints/story-oVCem8rM3ChqCKB0WR5L8J.html”. 2 May, 2016

[2] Zhang, Ting, Yuanxin Ouyang, and Yang He. "Traceable air baggage handling system based on RFID tags in the airport." Journal of Theoretical and Applied Electronic Commerce Research 3.1 (2008).

[3] Sloan, Michael, et al. "Radio frequency identification (RFID) material tracking and apparatus." U.S. Patent No. 7,195,159. 27 Mar, 2007.

[4] Ashwini Singh, Sakshi Meshram, Tanvi Gujar, P.R. Wankhede, “Baggage tracing and handling system using RFID and IoT for Airports”, IEEE 16th International Conference on computing analytics and security trends, 2016.

[5] Aicha Slassi Sennou, Asmae Berrada, Yassine Salih-Alj, Nasser Assem “An interactive RFID-based bracelet for airport luggage tracking system” Intelligent Systems Modelling & Simulation (ISMS), 4th International Conference on IEEE, 2013.

[6] AeroAssist, “RFID in Aviation: airport luggage control,” AeroAssist.pt, Jun, 2008.

[7]https://in.images.search.yahoo.com/search/images;_ylt=AwrPg1DLprpcQwMAsQ27HAx.;_ylu=X3oDMTB0N2poMXRwBGNvbG8Dc2czBHBvcwMxBHZ0aWQDBHNlYwNwaXZz?p=nodemcu&fr2=piv-web&fr=mcafee

[8]https://in.images.search.yahoo.com/search/images;_ylt=AwrPhmmKqLpcHgQAFAy7HAx.;_ylu=X3oDMTB0N2poMXRwBGNvbG8Dc2czBHBvcwMxBHZ0aWQDBHNlYwNwaXZz?p=rfid+reader+module&fr2=piv-web&fr=mcafee

[9]https://in.images.search.yahoo.com/search/images;_ylt=AwrPiFX4qLpcjQUA3Ay7HAx.;_ylu=X3oDMTB0N2poMXRwBGNvbG8Dc2czBHBvcwMxBHZ0aWQDBHNlYwNwaXZz?p=neo6m+gps+module&fr2=piv-web&fr=mcafee#id=1&iurl=https%3A%2F%2Fwww.makerlab-electronics.com%2Fmy_uploads%2F2017%2F06%2FGPS-Module-01.jpg&action=close

[10]https://electronicsforum.com/electronics-projects/rfid-based-access-control-using-arduino

# APPENDIX

**CODE 1:**

The program which is dumped to Nodemcu and send the RFID interfaced data to firebase is as follows

```
#define SS_PIN 15  //D8

#define RST_PIN 5 //D1

#include <SPI.h>

#include <MFRC522.h>

#include <ESP8266WiFi.h>

#include <FirebaseArduino.h>

#define FIREBASE_HOST "final-2b6e1.firebaseio.com"

#define FIREBASE_AUTH "ICTM3JGeOwXmcjTtxd3p7jfRJ3D18Uuce1VN0ggZ"

#define WIFI_SSID "Lite"

#define WIFI_PASSWORD "123456789"

MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.

int statuss = 0;

int count=0;

int out = 0;

void setup()

{

  Serial.begin(9600);
```

```
  SPI.begin();
  mfrc522.PCD_Init();
  Serial.begin(115200);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED)
  {
    Serial.print(".");
    delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP( ));
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
void loop()
{
  if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    return;
  }
```

```
if ( ! mfrc522.PICC_ReadCardSerial())
{
  return;
}
Serial.println();
Serial.print("UIDtag:");
String content= "";
byte letter;
for (byte i = 0; i < mfrc522.uid.size; i++)
{
   Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
   Serial.print(mfrc522.uid.uidByte[i], HEX);
   content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
   content.concat(String(mfrc522.uid.uidByte[i], HEX));
}
content.toUpperCase();
Serial.println();
if (content.substring(1) == "06 A8 48 1B")
{count++;
if(count%2!=0){
    Serial.println(" Boarded to correct flight ");
```

```
    Firebase.setString("rfid","Boarded to correct flight(AIR707)");
    String str = Firebase.getString("rfid");
    Serial.println(str);
  }
  else {
    Serial.println(" Deboarded from correct flight ");
    Firebase.setString("rfid","Deboarded from correct flight(AIR707)");
    String str = Firebase.getString("rfid");
    Serial.println(str);
  }
  }

  else   {
   Serial.println(" Access Denied ");
   Firebase.setString("rfid","Luggage in wrong flight");
   String str = Firebase.getString("rfid");
   Serial.println(str);
  }
}
```

**CODE 2:**

The program which is dumped to Nodemcu and send the GPS interfaced data to firebase is as follows

```
#include <TinyGPS++.h>

#include <SoftwareSerial.h>

#include <FirebaseArduino.h>

#include <ESP8266WiFi.h>

static const int RXPin = 12, TXPin = 13;

static const uint32_t GPSBaud = 9600;

const char* ssid     = "Lite";

const char* password = "123456789";

#define FIREBASE_HOST "luggagetracking-6edd9.firebaseio.com"

#define FIREBASE_AUTH "R18Na5Y7mQOg4Q0Tjyiri50p2AAdn8Ln6fEjbfTY"

TinyGPSPlus gps;

SoftwareSerial ss(RXPin, TXPin);

void setup()

{

  Serial.begin(115200);

  ss.begin(GPSBaud);

  Serial.println(F("GPS module"));

  Serial.println(TinyGPSPlus::libraryVersion());
```

```
 Serial.println();
 Serial.print("Connecting to ");
 Serial.println(ssid);
 WiFi.begin(ssid, password);
 while (WiFi.status() != WL_CONNECTED) {
   delay(500);
   Serial.print(".");
 }
  Serial.println("");
 Serial.println("WiFi connected");
 Serial.println("IP address: ");
 Serial.println(WiFi.localIP());
 Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
void loop()
{
   while (ss.available() > 0)
   if (gps.encode(ss.read()))
     displayInfo();
 if (millis() > 5000 && gps.charsProcessed() < 10) {
   Serial.println(F("No GPS detected: check wiring."));
```

```
    while(true);
  }
}
void displayInfo()
{
  if (gps.location.isValid())
  {
    double latitude = (gps.location.lat());
    double longitude = (gps.location.lng());
    String latbuf;
    latbuf += (String(latitude, 6));
    Serial.println(latbuf);
    String lonbuf;
    lonbuf += (String(longitude, 6));
    Serial.println(lonbuf);
    Firebase.setString("gps/latitude",latbuf);
    Firebase.setString("gps/longitude",lonbuf);
    String str1 = Firebase.getString("gps/latitude");
    String str2 = Firebase.getString("gps/longitude");
    Serial.println(str1);
    Serial.println(str2);
```

```
    delay(100);
  }
  else
  {
    Serial.print(F("INVALID"));
    Serial.println();
    Firebase.setString("gps/latitude","NOT DETECTED");
    Firebase.setString("gps/longitude","NOT DETECTED");
    String str1 = Firebase.getString("gps/latitude");
    String str2 = Firebase.getString("gps/longitude");
    Serial.println(str1);
    Serial.println(str2);
    delay(100);
  }
  Serial.println();
}
```

Android application code is available on:

https://github.com/Brundashanthmurthy/Luggage_trackingsystem