ACKNOWLEDGEMENT

I would like to take this opportunity to thank a few people, without whose assistance this project would not have been successfully accomplished.

First of all, I want to thank the CEOs of VRCO Ltd and Raine Dev Ltd, Mr. Michael Smith and Mr. Soorya Jeyaraj for giving me the opportunity and access to the resources to carry out my placement year at their respective companies. They also provided me with valuable guidance which always helped me to improve my work skills and funded my educational trips through which I gained precious experience.

Finally, I thank my parents, friends, and everyone at the University for their support in everything.

I immensely thank everyone for buffing up my efforts in making such a triumphant placement year!

INDEX

I. ILLUSTRATIONS

II. TABLES

III. ABBREVIATIONS AND MEANINGS

B.Eng	Bachelor of Engineering
IP	Industrial Placement
IPO	Interdisciplinary Programs Office
UoS	University of Sheffield
UAV	Unmanned Aerial Vehicle
UI/UX	User Interface/ User Experience Design
PCB	Printed circuit Board
Wi-Fi	Family of Wireless Networks
MQTT	MQ Telemetry Transport
HTTP	Hypertext Transfer Protocol
IFTTT	If This Then That
USB	Universal Serial Bus
TTL	Transistor- Transistor Logic
LED	Light Emitting Diode
RX/TX	Receiving/ Transmitting
IDE	Integrated Development Environment
HDMI	High-Definition Multi-media interface
VNC	Virtual network Computing
MAVlink	Micro Air Vehicle Link
RESTCONF	HTTP based protocol to provide programmatic interface
gRPC	Remote Procedure Call
ADSB	Automatic dependence surveillance broadcast
Figma	Software for webdesign and designing in general.

1. INTRODUCTION

This report consists of the details related to the Industrial Placement Year which I went through as a requirement of the B.Eng. Aerospace Engineering Degree program that I am enrolled at IPO Department of the University of Sheffield. An introduction of the company which I worked for, categorized descriptions of the work I carried out during the work period, my own reflections about the work experience and the skills I improved during the time have been described throughout the report. Each project mentioned in the report has been broken down into categories to relate with the competencies and commitments with respect to the UK-SPEC. The 5 mail competencies and commitment are: A- knowledge & understanding; B- design, development, and solving engineering problems; C- responsibility, management, and leadership; D- communication, research, and interpersonal skills; E- professional commitment. Every project covers the aspects of A and B competencies. Competencies C, D, and E have been discussed towards the final part of the report.

During the IP, I had the wonderful opportunity of relating to as well as applying the knowledge and skills that I had learned so far and to work in the industry. I am confident to return to final year of study with a beneficial work experience which is supportive to the advanced academic studies and the individual project.

I have mostly explained the key points of important and interesting projects. Some of the projects have simple explanation as I played a minor role in the project and I was not entirely responsible for the project. For the projects that relied completely on me, I have provided a detailed description.

I was offered two roles for different period of time and as a result I worked for two different companies. The experience I received working at VRCO Ltd and Raine Dev Ltd as an Aerospace Engineer and Junior Design Intern for a period of 38 weeks is priceless to me. One was a startup and other was a large-scale company which molded me to be initiative driven, provided more opportunities to learn and grow, and best mentoring and support.

2. COMPANY PROFILE

VRCO Ltd is an Aerospace Engineering company based out of Derby. They focus on development and production of their major product XP4. Raine Dev Ltd is a software development company from Sheffield. They focus on selling various software services from mobile apps to backend data structure and maintenance.

2.1 Details of the Companies

Name of the Company	VRCO Ltd
Email Address	info@vrco.co.uk
Postal Address	ihub, Innovation Drive, Infinity Park Derby, Derbyshire, DE24 9FU
Person of Contact	Mr. Michael Smith (CEO, Co-Founder)
Email Address (POC)	mike@vrco.co.uk
Website	https://vrco.co.uk/

Table1: VRCO Ltd company details.

Name of the Company	Raine Dev Ltd
Email Address	support@eurusog.com
Postal Address	Raine Dev Ltd, EURUSOG, Regus the Balance, Pinfold Street, Sheffield, S1 2GU
Person of Contact	Mr. Soorya Jeyaraj (CEO, Founder)
Email Address (POC)	soorya@eurusog.com
Website	https://find-and-update.company-information.service.gov.uk/company/12714889

Table 2: Raine Dev company details.

2.2 Goals

2.2.1 VRCO Ltd

Defining Air Volution, redefining Aviation

“VRCO has developed a vision to change the face of luxury human transport.”

“Our vision over the next decade is that the shackles of commuting are falling away andit’s an age of Air Volution. These will be supercars of the skies. They will be luxury and high-performing.”

2.2.2 Raine Dev Ltd

Redefining workflows, optimizing production

“Buy, sell and trade through our marketplace. Our authentication team ensures ouritems are genuine.”

2.3 Introduction to the Companies

2.3.1 VRCO Ltd

VRCO has a vision for a flying car with a new form of personal aircraft as the initial phase. It is a showcase for British innovation with advanced lightweight composites enhanced with graphene nanomaterials.

The vertical take-off and landing craft will be capable of carrying four people. It features detection and avoidance technology with safety measures including a ballistic parachute and low altitude crash prevention systems.

Working with leading innovators and key industry partners at the forefront of the UK’s power storage systems, the craft will be fast-charging and have enhanced range.

Drawing on successes in the UAV space, Michael began the XP4 design in August 2016and has now moved from design and concept into the build phase.

Full-size build should be completed later this year with operational testing in 2022 and certification beginning in 2023/24.

VRCO’s founders have experience in unmanned aerial vehicles and have identified several novel component solutions to critical engineering challenges.

The concept’s feasibility and power optimization studies were undertaken by the University of Derby’s Institute for Innovation in Sustainable Engineering.

2.3.2 Raine Dev Ltd

This company was started with passionate projects from ideas of Mr. Soory along with his dear friend Mr. Dean. They registered and released their mobile app for android and IOS in 2020. Currently, they have clients for web development, web designing, data base handling. They have experience in computer science and control systems engineering.

They run the company together and aim to build a Multi-million dollar company within a period of 5 years in the field of computing, software tech, artificial interlligence, automation, and data.

2.4 Services

2.4.1 VRCO Ltd

a. XP4 development, designing, customization, manufacturing, pre-bookings and production.
b. Personalized Simulation development.
c. Investments.

2.4.2 Raine Dev Ltd

a. Mobile Apps (IOS, Android).
b. Static website.
c. Website development.
d. Website designing.
e. E-commerce website.
f. Back-end database management.
g. Front-end design.
h. Front-end development.

2.5 Organizational Structure

2.5.1 VRCO Ltd

They followed the structure of a large-scale company.

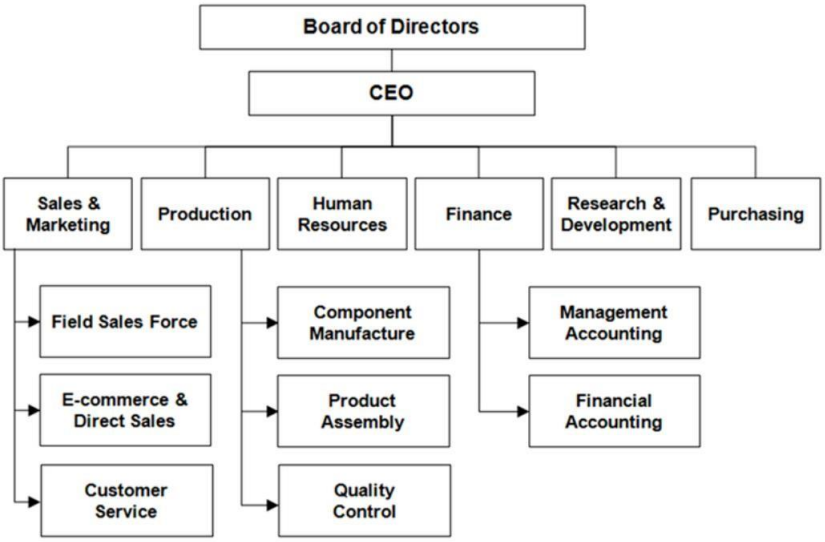


Figure 1: Organizational structure of VRCO as a large-scale company.

2.5.2 Raine Dev Ltd

They followed the systems of a SME or start-up companies.

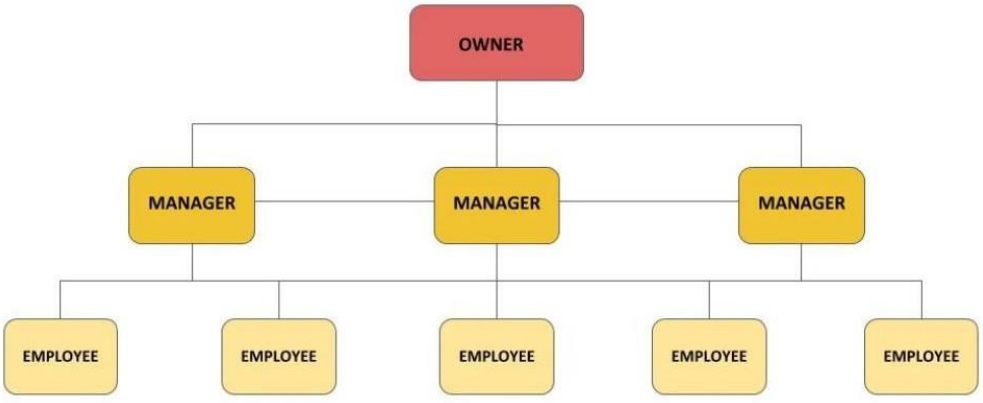


Figure 2: Organizational structure of Raine Dev as a start-up or SME.

3. PLACEMENT PROJECTS

Experiencing projects with both companies made me have varied experience. Projects with VRCO Ltd demanded acute accuracy, had tight deadlines, and was more inclined towards my course of study, whereas with Raine Dev Ltd, it was software based and they provided a good mentorship. I was able to learn new UI/UX designing skills.

Most projects with VRCO Ltd were self-taught and experimental as the company strongly believed in the skill of self- teaching. Due to medical conditions, some of my work with VRCO was carried out in a hybrid work style which also gave me equal insight into the project topic as the on-site projects. With respect to hybrid method, I was sent home the kit or working tools and 3D printed parts to work from home.

The standard of work in each working culture was outstanding. Although the time frame for all the projects combined was 4 to 5 months, I was able to complete it in a period of around 3 months with VRCO Ltd as they kept motivating me and pushing me out of my comfort zone to excel in the real world. On the other hand, the work culture of Raine Dev was flexible and was not extremely demanding as VRCO. They gave me the full freedom to choose my working times and all they required was the completion of work within the given deadline. I have explained the projects with VRCO and Raine Dev below.

3.1 VRCO Ltd

3.1.1 Level 1 Surface Mount Soldering

Mentors for this project: Mark de Vinck, LinkedIn Learning.

Task assigned and purpose: Surface Mount Soldering of avionics system from Tokyo model representation of the warehouse that the company was building.

Description: There were many terms and techniques for this project, a few of them have been shared below. It was important to follow safety procedure and was a must to:

a. Wear safety glasses
b. Turn on fume extractor
c. Have the magnification equipment near you

d. Wash hands after soldering
e. Lighting/ a tiny solder lamp
f. Not eat while soldering
g. Have timer by your side if needed

There are various types of soldering techniques such as paste/ using stencil, wire solder, flux core, more.

Flux is the magic tool that melted when the soldering tool was used to bind the solder and the wires. It maintains electrical and mechanical connections. Isopropyl alcohol must be used to clean the PCB before and after soldering. There are various types of flux such as:

a. Chemical
b. Liquid
c. Pen type
d. In build solder

The tip of the solder must be soldered before the start at all times. A 45-degree angle (of wire from PCB) or a little volcano shape after the solder indicates a good soldering technique and will stay intact. I used solder of thickness from 0.6 to 0.8 and a satellite grade material for most of the soldering.

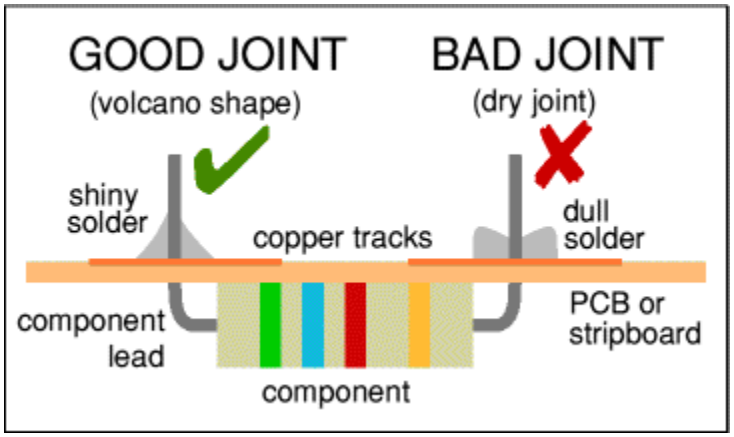


Figure 3: Little volcano shape/ cone shape indication of good soldering

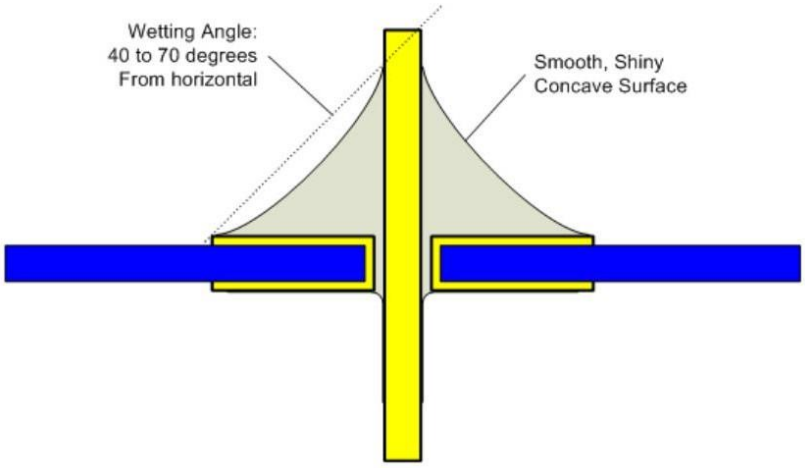


Figure 4: An ideal solder joint

Lineman's spice is a connection type to combine the wires for a better finished soldering.

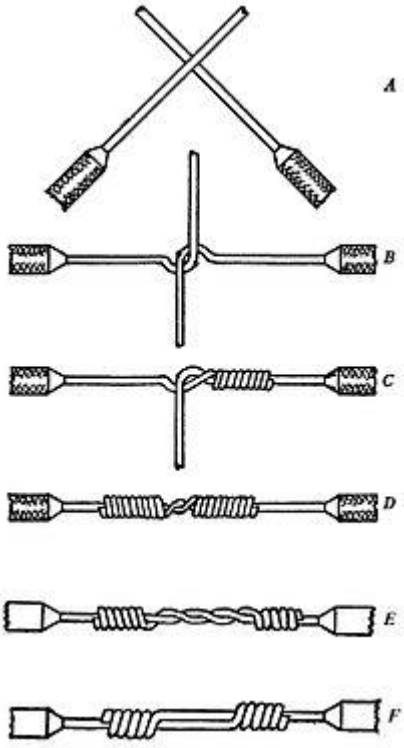


Figure 5: Lineman's spice

Usually, PCB cleaners consists of harmful chemicals. It is used in manufacturing and for ultrasonic cleaners and it has always been better to opt for isopropyl alcohol for soldering. Most of the PCB designing can be done on Eagle Altium or KiCAD.

There were a range of hardware and software tools used for this project. Some include:

Hardware

a. Solder iron
b. Hot air gun
c. Heat sink
d. Solder
e. Flux
f. PCB holder and PCB
g. Solder paste
h. Solder remover or gun or copper braids
i. Wires and wire cutters
j. Tapes
k. Helping hands
l. Practice PCB kit with resistors and capacitors imported from China

Software

a. AM CAP, OBS STUIO- for pictures and recordings of the soldering

Problems, solutions, and troubleshooting: Common mistakes that I learnt from were-

a. A cold solder junction- this develops when the solder does not completely melt. Cold joints, the result of insufficient heat, are frequently hard, harsh, and uneven in appearance. This soldering error results in an unreliable junction that is very prone to failure and cracking. Repairing cold joints is typically as simple as heating the connection again with a hot iron until the solder flows. Too much solder can also cause cold junctions, like the one in the image. Usually, the tip of the iron can be used to remove extra solder. Cold junctions can be avoided with a soldering iron that has been properly preheated and is powerful enough. Increasing the flux can also help with this.

Figure 6: Cold solder joint

b. A disturbed solder junction- this is due to the movement while the solder was hardening. The joint's surface might have a frosted, crystalline, or rough appearance. known as a "Cold Joint" a lot. The reason may be different, despite the fact that they may resemble a real cold joint. Reheating and letting it cool naturally will mend this joint. In order to prevent disturbed joints, take the proper precautions, such as immobilizing the joint and securing the work in a vice. To clean and solder correctly, remelt the solder and make use of copper braiding or other solder-removing equipment.

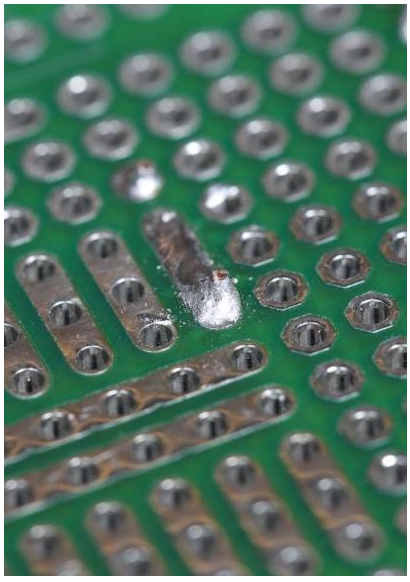


Figure 7: Disturbed solder joint

c. Overheated joint - The overheated joint represented the other extreme. Fixing this joint was challenging since the flux burned off and the solder did not flow well. After cleaning, an overheated joint was typically be repaired. The burnt flux was removed with a little isopropyl alcohol, a toothbrush, and careful scraping. Overheated joints were avoided with the use of a clean, hot soldering iron, appropriate setup, and thorough cleaning of the junction.

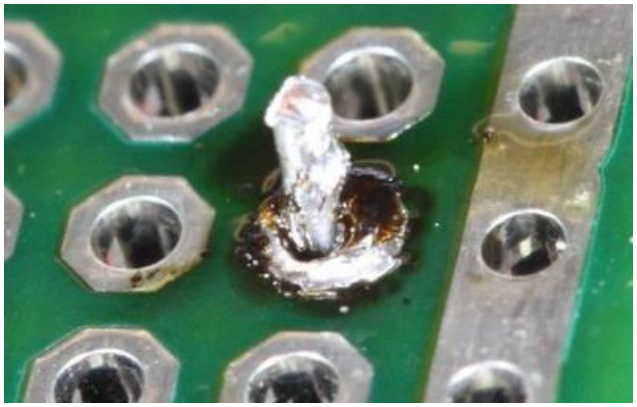


Figure 8: Overheated solder joint

d. Insufficient wetting (pad): The solder pad in each of these junctions appeared to be sufficiently moistened. The leads were nicely moistened by the solder, but the pad did not receive a strong bond. This may be the result of a wretched circuit board or a failure to heat both the pad and the pin. The majority of the time, this issue may be fixed by holding the tip of a hot iron at the base of the junction until solder flows to cover the pad. This issue was avoided by cleaning the board and evenly heating the pin and pad.

Figure 9: Insufficient wetting (pad)

e. Inadequate wetting (pin): The solder in this junction has only partially wetted the pad and was not wetted the pin at all. In that instance, the pin was not heated, and the solder did not have enough time to flow. Reheating and adding more solder allowed for the repair of this junction. The pin and the pad must be in contact with the hot iron's tip. This issue may be avoided by evenly heating the pin and the pad.

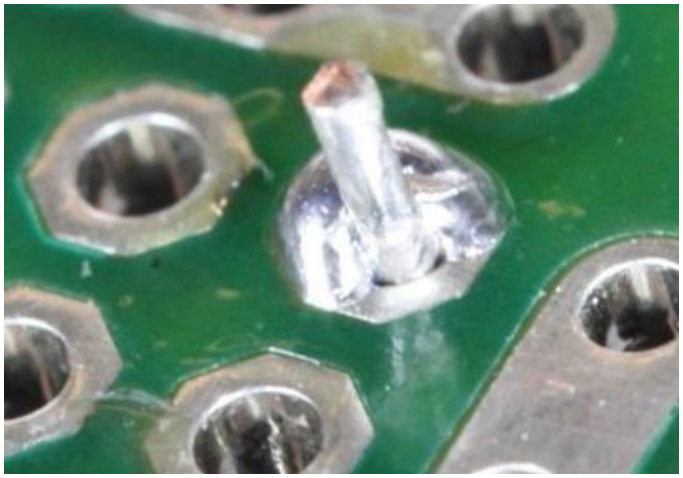


Figure 10: Inadequate wetting (pin)

f. Inadequate soldering (Surface Mount): There were three pins on this surface-mount component where the solder did not flow onto the solder pad due to insufficient wetting. This resulted from heating the pin rather than the pad. This

might be quickly fixed by adding solder to the heated solder pad using the tip of the iron until the solder flows and melts with the solder already on the pin. First, heat the pad.

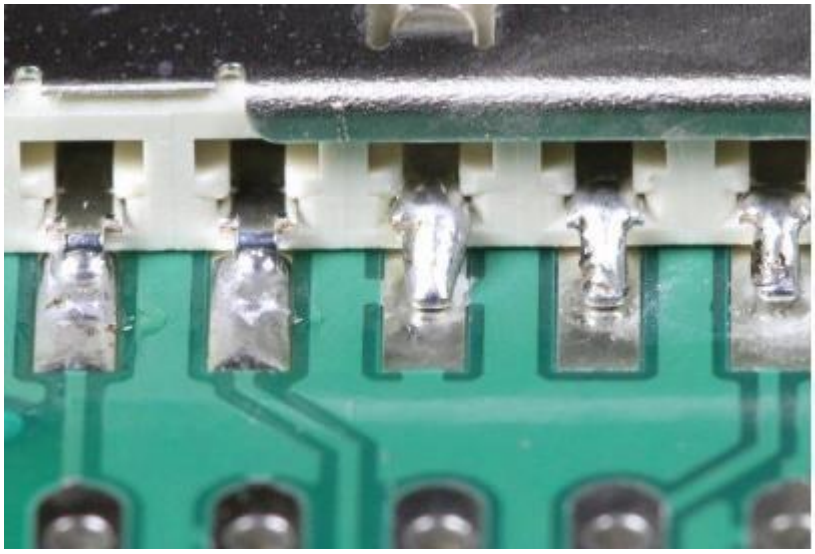


Figure 11: Inadequate soldering (Surface Mount)

g. Solder starving: A junction that lacks solder is said to be solder starved. It could produce excellent electrical contact, but it's difficult to tell by looking at it. It was a weak joint that eventually broke due to stress fractures. To create a solid junction,the joint was re-heated, and more solder was added.

Figure 12: Solder starving. Lack of solder around the area.

h. Extra solder: This solder glob did not moisten the pin or the pad, and it was not a strong electrical connection. A lovely concave surface, like the one on the joint on the far left, is the greatest indication of appropriate wetting (and excellent electrical contact). With a heated iron's tip, some of the extra solder was removed. A solder-sucker or some solder wick was useful in extreme circumstances.

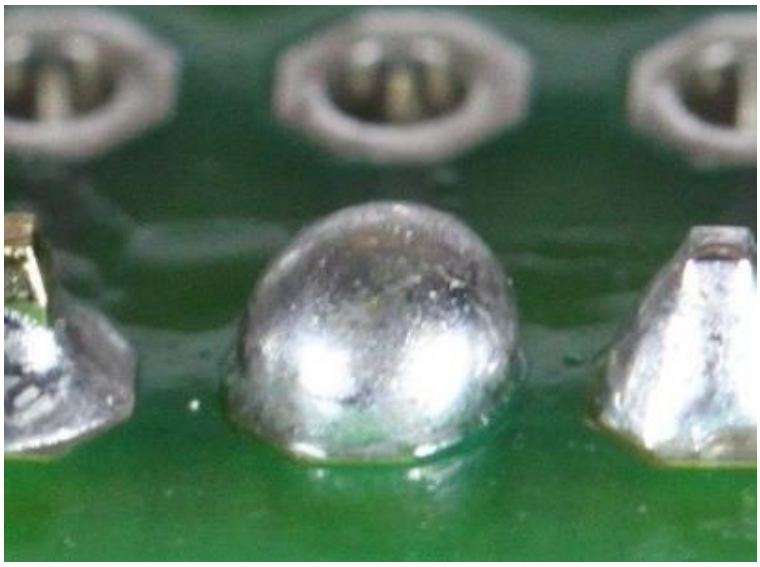


Figure 13: Excess solder and blob formation.

i. Untrimmed leads: short circuits could develop with excessively long leads. There was a clear risk of contact between the two joints on the left. The one on the right, though, was long enough to pose a threat as well. This lead was easily forced to bend such that it made another trace. Just at the top of the solder junction, all leadswere trimmed.

Figure 14: Leads not trimmed

j. Solder bridge: The two solder joints on the left were melted together and connected unintentionally. The tip of a hot iron would occasionally be used to pull the extra solder between the two solder junctions. A solder sucker or solder wick can assist remove extra solder if any. Solder bridges most frequently develop between junctions when there is already too much solder. Just enough solder was needed to create a solid bond.

Figure 15: Formation of bridge type of solder due to excess solder and simultaneous melting of two joints together.

k. Lifted pad: This most frequently happens while attempting to remove components from a circuit board. However, it might happen simply as a result of overworking the junction to the point where the copper and board's adhesive bond is broken.

Boards with thin copper layers and/or holes without through plating are more likely to have lifted pads. The easiest fix should be to flip the lead over to a copper trace that was once attached and solder it, as seen on the left. I had to gently scrape off enough solder mask from the board in order to reveal the bare copper.

Figure 16: Lifted pads

l. Stray solder spatters: Only the sticky flux residue is keeping these solder fragments on to the board. They were readily capable of causing a short circuit on the board when they were operated loosely. With the use of tweezers or the pointof a knife, they were simple to remove.

Figure 17: Stray solder pads spatters

m. Copper braids are life savers for soldering. Most of the times when there appears additional or excess solder, this helps remove it easily as it absorbs the solder.
n. For extreme mistakes use heavy duty cutters to cut solder out or the connections to the circuit board directly and use copper enamel coat.

Reflection: This contributed to the completion of Level 1 soldering and a certification for the same. I was able to assist the VRCO team with soldering the wiring system or the avionics system for the mockup or prototype model of the XP4 and their drone systems that they are yet to introduce. I used soldering skills for SURE project. (Note: Due to the confidentiality of this, pictures have not been shared).

3.1.2 IoT- Internet of Things

Mentors for this project: Mr. Kishan (senior student and an employee in an Indian company), YouTube, Google.

Task assigned and purpose: Understanding the working of basic IoT projects and using the similar methods for a greater goal.

Description: IoT stands for the "internet of things," which is the use of the internet to control objects. This refers to two or more linked devices that can perceive their environment and communicate with one another. Device categories:

a. General device - comprises wired or wirelessly linked household appliances.
b. Sensing equipment (sensors, actuators)—measures temperature, humidity, etc.— sends data to gateways, where it is processed and transferred to the cloud.

Cloud serves as a unit for storage and processing for subsequent phases where activities are carried out. Bluetooth, Wi-Fi, or ZigBee connections are made available; communication data is transferred via the MQTT or HTTP communication protocol. The Internet of Things is essentially the future—smart cars, smart homes, smart cities, smart watches, smart farms, and more.

By 2025, 80 billion devices are anticipated to be linked to the cloud. Two projects were examined, and one of them was used to execute the Tokyo model (using relayx4) and theother was meant to be incorporated in the avionics dashboard of XP4 model.

The projects are discussed in a step-by-step process below:
Most projects with VRCO Ltd were tested with Arduino which is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online.

Project 1: Adafruit and IFTTT
Adafruit IO is an IoT service to manage Arduino and view real-time data online while IFTTT derives its name from the programming conditional statement “if this, then that” and it allows you to do more with over 700 different apps and services, including Twitter, Dropbox, Evernote, Fitbit, Amazon Alexa, and Google Assistant.

Step 1: An account was created in Adafruit. New feed (named ‘on or off’ was added) and a new dashboard (named ‘light status’). IO was copied for later use.

Step 2: New account created on IFTTT, and a new applet was included.

Step 3: Now the freedom to choose the options were provided. I selected IF (google assistant- say a simple phrase) and THEN (adafruit- sent IO). This was the scenario and to make it clear the previous line was like coding. It was similar to using if statements. Here it refers that if google assistant picks up a simple phrase such as a command (example: turn on the lights), then it immediately linked with adafruit and executed the described condition on adafruit which was to turn on the lights.

Step 4: The circuits and codes were checked along with the port, library, board manager, output, and console (this must be kept tract to confirm the communication). Once run, the console displayed communication and it was tested using Google assistant. The commands were given as voices to Google assistant and the circuit turned the on and off lights.

The second project was a modification on the software part to the first project as described above. The hardware was similar.

Project 2: Blynk and IFTTT

Blynk is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device. After downloading the Blynk app, you can create a project dashboard and arrange buttons, sliders, graphs, and other widgets onto the screen. Node MCU is an open-source platform based on ESP8266 which can connect objects and let data transfer using the Wi-Fi protocol. In addition, by providing some of the most important features of microcontrollers such as GPIO, PWM, ADC, and etc. It can solve many of the project's needs alone.

Step 1: Account was created on Blynk app along with new project (named as required), a button, push to switch.

Step2: Set to Node MCU and Wi-Fi. This sent authentication code to the email.

Step 3: For IFTTT, the codes were I (google assistant- simple phrase), then (webhooks- url: http://ipaddress/auth_code/update/pin->put->application/json->set).

Step 4: Body was set to [“1”] or [“0”] depending on on or off respectively.

Step 5: Codes and circuits were checked. The codes were available on the Arduino examples as Blynk WiFi NodeMCU under the Blynk category.

The second project was used for the Tokyo model lights where every detail such as the soldering of series and parallel connections of resistors where connect to the lights. This was installed with Relayx4 (x4 indicates 4 units of relay) and Node MCU. The following circuit diagram was used.

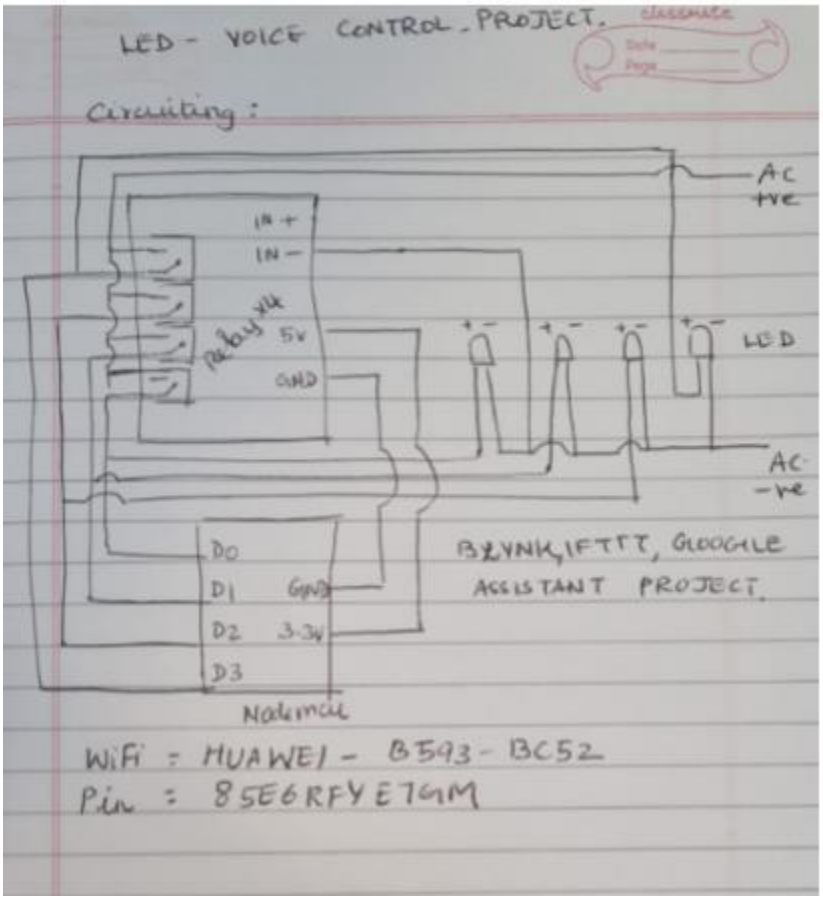


Figure 18: Rough representation of the circuiting.

There were a range of tools used for this project. They are listed below:

Hardware

a. Relay x4
b. Wi-Fi ESP Module
c. Jumper wires
d. Node MCU
e. Lights for the Tokyo Model
f. Micro USB Cable
g. 12V DC supply
h. DC supply (5V) to relay x4
i. Office mobile

Software

a. Blynk app

b. IFTTT
c. Google Assistant
d. Webhooks
e. Arduino IDE

Problems, solutions, and troubleshooting: Issues that I faced during this process and thecorrection procedures-

a. In IFTTT, avoid referring to the pin directly as D0 or D1; instead, examine thehardware and use code such as gpio16; the pin here is D16.
b. Before installing the board, check the port, board management URL, Blynk settings, library, and make sure the lights are operational by using a voltmeter or lights tested.
c. It is not a good practice to open the console while the code is being flashed on the board; instead, wait until the codes have fully uploaded before opening the console and pressing reset on the NodeMCU to display Blynk on the console.
d. Ensure that your IP address is accurate at all times. Pinging Blynk-Cloud.com from the PC's search bar and pressing Enter will reveal Blynk's IP address.
e. Use the 80 or 8080 ports in codes to prevent traffic or gateway.
f. Each light has its own voltage level; hence resistors are required to prevent burnouts.
g. Whenever possible, use resistors.
h. Verify that the Google account you use for the Google Assistant app is the same one you used for IFTTT.
i. IFTTT is a free application or website; nevertheless, for secure Internet of Things applications, utilize alternative cloud platforms.

Troubleshooting or loop tests on Arduino for this project:

a. Disconnect the board from your computer or battery.
b. Remove all shields, jumper cables, and other connections from the board.
c. Connect jumper cable from RESET to GND, Digital 0 (RX) to Digital 1 (TX).
d. Connect the board to your computer and open the Arduino IDE or the Arduino Cloud Web Editor.

e. Ensure correct port selection: For Arduino IDE follow Tools > Port > [Your Arduino Board] and for Web Editor make sure to select your board using the boards dropdown.
f. Open the Serial Monitor: For Arduino IDE follow Tools > Serial Monitor and for Web Editor, select monitor on the side bar.
g. Enter a message and click Send (the baud rate does not matter). Your message should immediately be echoed by the board and appear in the output field below.

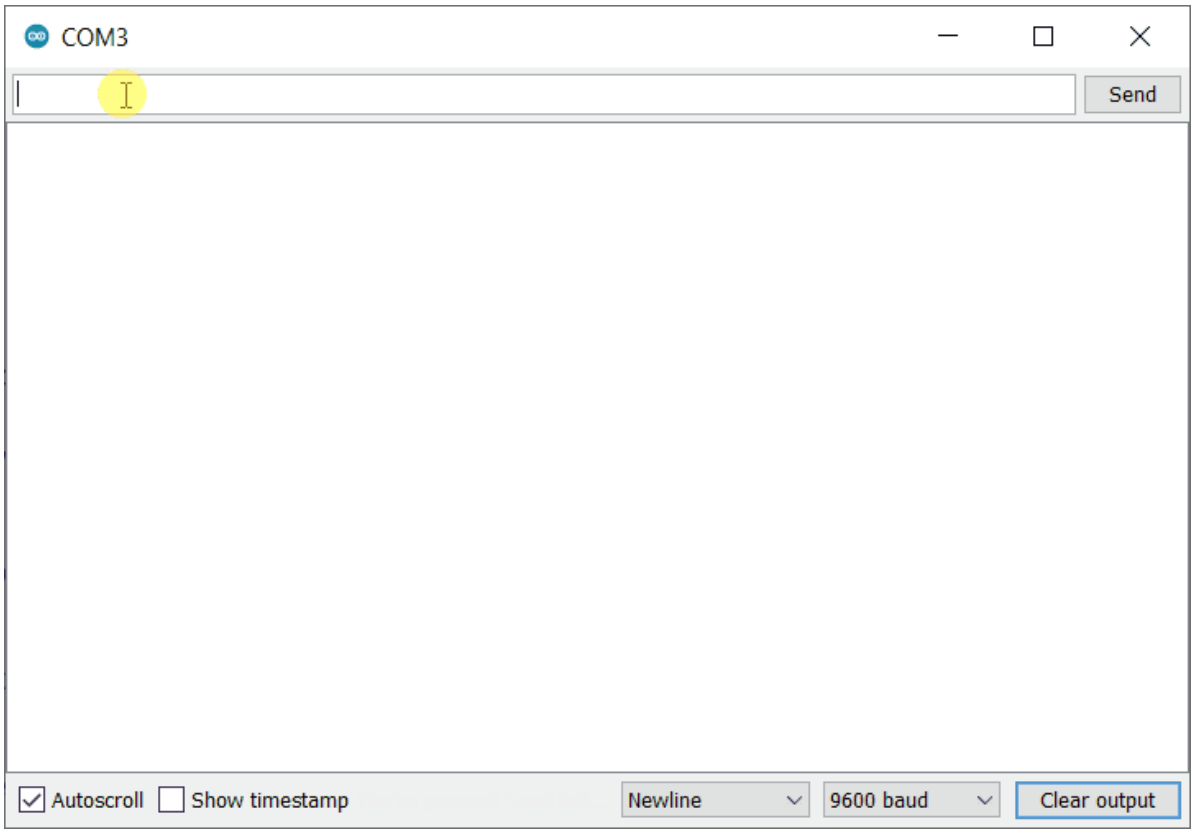


Figure 19: Console or Output field that must display text when tested or communication occurs.

Reflection: This helped my incorporate the projects into the Tokyo model system. It was demonstrated to the CEO and team. Additionally, a few aspects of these projects were also tested for the purpose of designing the avionics dashboard of the XP4 and future projects, the details of which cannot be disclosed.

3.1.3 Voice Recognition module

Mentors for this project: Jonathan (UoS, Electronics lab in-charge), Akash (University of Dhaka, Final year Robotics and Mechatronics Engineering student), MERT Arduino and tech- YouTube Channel, Mann (UoS, Final year Control Systems Engineering student).

Task assigned and purpose: To understand Voice recognition module and research it in comparison with IoT projects.

Description: Voice Recognition module is Arduino based and is simple to use if the data sheet is followed along with no glitches from Arduino and Access Port (Serial port monitoring tool). The voice commands must first be imported into the module using software. The voice importation process can only be started when the module is linked to the computer via a USB to TTL converter. Information on the voice module: http://www.geeetech.com/wiki/images/6/69/Voice_Recognize_manual.pdf (hex format basic commands). Notable commands for recording include:

a. AA11- recording of group 1
b. AA12- recording of group 2
c. AA00- waiting/ preparing
d. AA36- common mode (while recording the commands)
e. AA37 – compact mode (while using the commands in Arduino)

As you record voice instructions, data should appear on the access port's terminal screen. You may check your voice commands by pressing the play and stop buttons on the screen (if it works, you will be able to see the result on terminal screen). Data sheets for all products and modules must be provided. Be careful to read the data sheet, which contains all pertinent information. Social media is the ideal resource for approaching and learning since people there also discuss issues and solutions. Data sheet for the WiFi Relay module: http://www.icstation.com/esp8266-wifi-channel-relay-module-remote-control-switch-wireless-transmitter-smart-home-p-13421.html.

The figure 20 was the terminal screen, monitor screen, menu bar, commands, results, input, and the output for that project in Access Port.

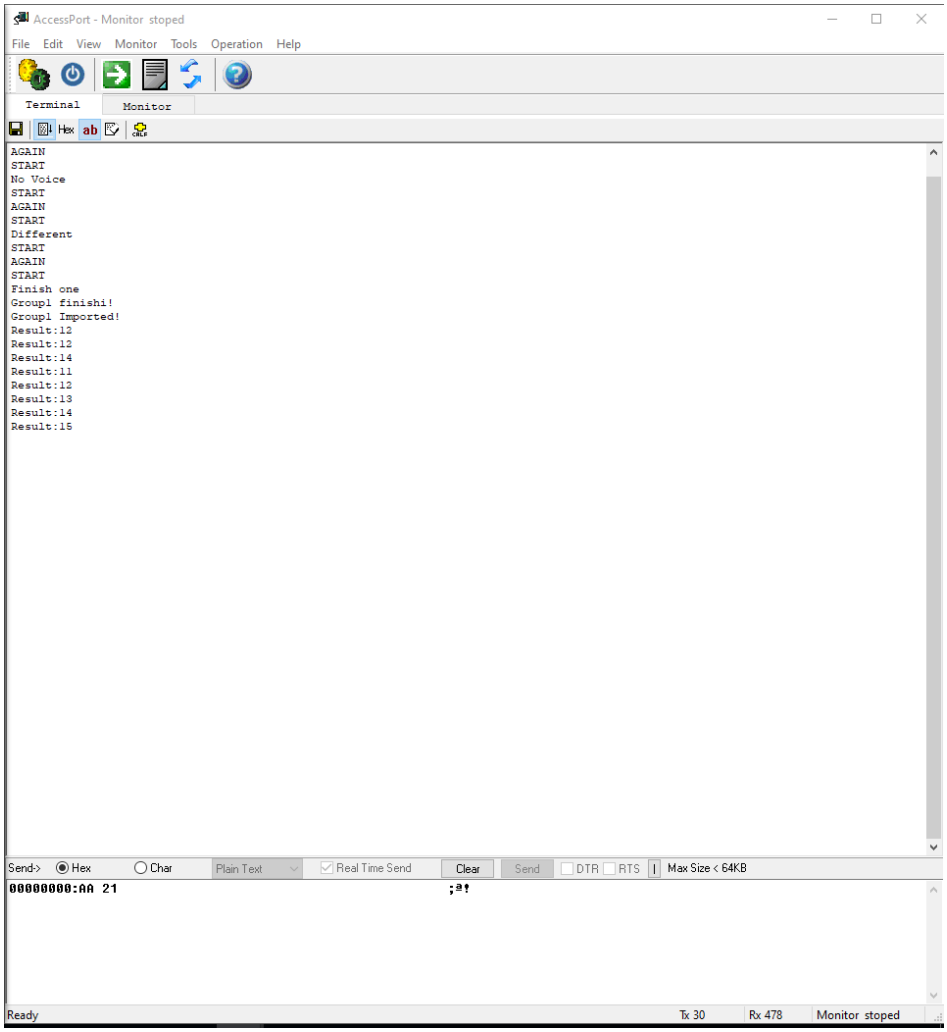


Figure 20: Access Port representation of commands, menu bar, and interface ingeneral.

Once the module was imported, it was disconnected from the USB to TTL converter, attached to the Arduino, and used the code-pen (social platform that stores the codes and shares them) codes for this project. Alternatives I also had the option to use the V3 Module codes on office computer account (info@vrco). Because we did not want to include a phone with the model, we utilized this project instead of project 2 of the IoT for the Tokyo model. Pushing the S button switched the relay from mode 1 to mode 2 and pressing the reset button reset the board. Removing the power supply and continued press of the reset button while supplying 12 volts switched the relay from mode 2 to mode 1.

Notice the codes in Figure 21 and 22, they represent the Hex format of data. The figure 23 shows the Relay testing and commands code.

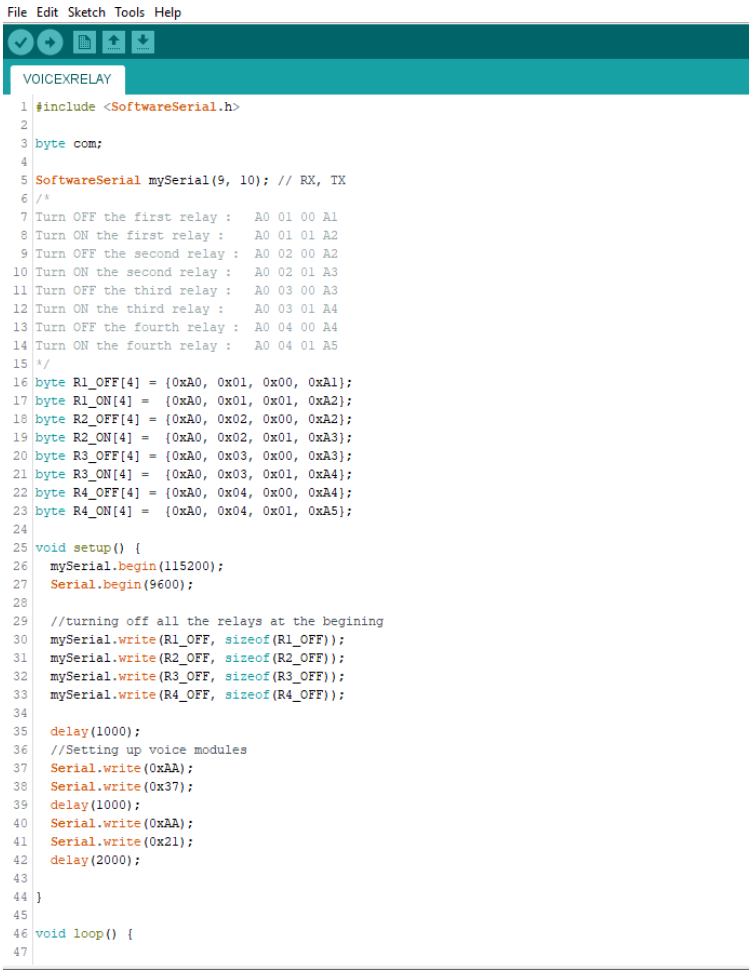


Figure 21: Codes for the voice module project and representation of Hex format.

VOICEXRELAY

```
  Serial.write(0xAA);
  Serial.write(0x37);
  delay(1000);
  Serial.write(0xAA);
  Serial.write(0x21);
  delay(2000);

}

void loop() {

  while(Serial.available()) {

    com = Serial.read();

    switch(com) {

        case 0x11:   //command 11 is for air side LED
          mySerial.write(R1_ON, sizeof(R1_ON)); // command switch on the relay 1
          break;

        case 0x12:  //command 12 is for road side LED
          mySerial.write(R2_ON, sizeof(R2_ON)); // command switch on the relay 2
          break;

        case 0x13:  //command 13 is for security LED
          mySerial.write(R3_ON, sizeof(R3_ON)); // command switch on the relay 3
          break;

        case 0x14:  //command 14 is for motion LED
          mySerial.write(R4_ON, sizeof(R4_ON)); // command switch on the relay 4
          break;

        case 0x15:  //command 15 is for switching off LED
          mySerial.write(R1_OFF, sizeof(R1_OFF)); // command switch off the relay 1
          delay(300);
          mySerial.write(R2_OFF, sizeof(R2_OFF)); // command switch off the relay 2
          delay(300);
          mySerial.write(R3_OFF, sizeof(R3_OFF)); // command switch off the relay 3
          delay(300);
          mySerial.write(R4_OFF, sizeof(R4_OFF)); // command switch off the relay 4
          delay(300);
          break;

    }
  }
}
```

Uploading to I/O Board...

```
        at jssc.SerialPort.closePort(SerialPort.java:1090)
```

Figure 22: Continuation of the voice module code and representation of switch cases used.

15>.Relay control command(Hexadecimal):
Turn OFF the first relay : A0 01 00 A1
Turn ON the first relay : A0 01 01 A2
Turn OFF the second relay : A0 02 00 A2
Turn ON the second relay : A0 02 01 A3
Turn OFF the third relay : A0 03 00 A3
Turn ON the third relay : A0 03 01 A4
Turn OFF the fourth relay : A0 04 00 A4
Turn ON the fourth relay : A0 04 01 A5

Fig 23: Hex format codes from the data sheet for the communication with voice module.

Rough schematics and circuiting were done on the electronic layout were made as depicted in figures 24 and 25.

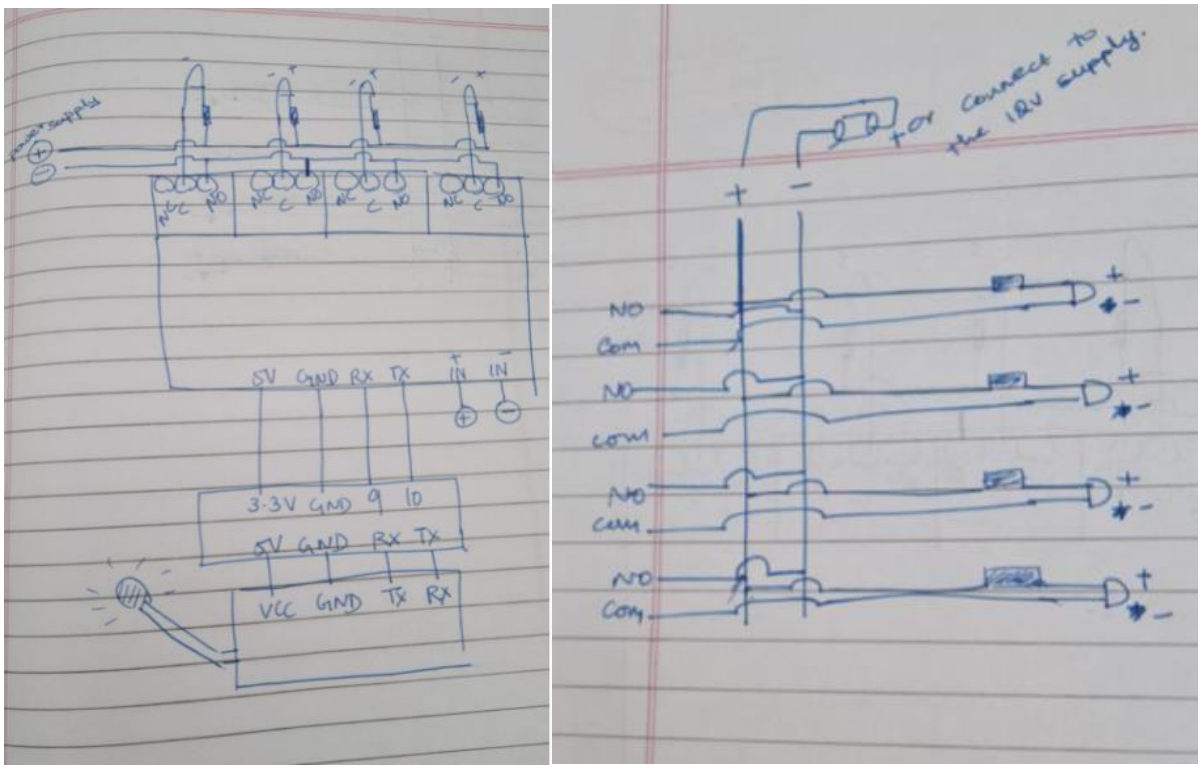


Figure 24: Rough circuit diagram of the voice module and relay units.

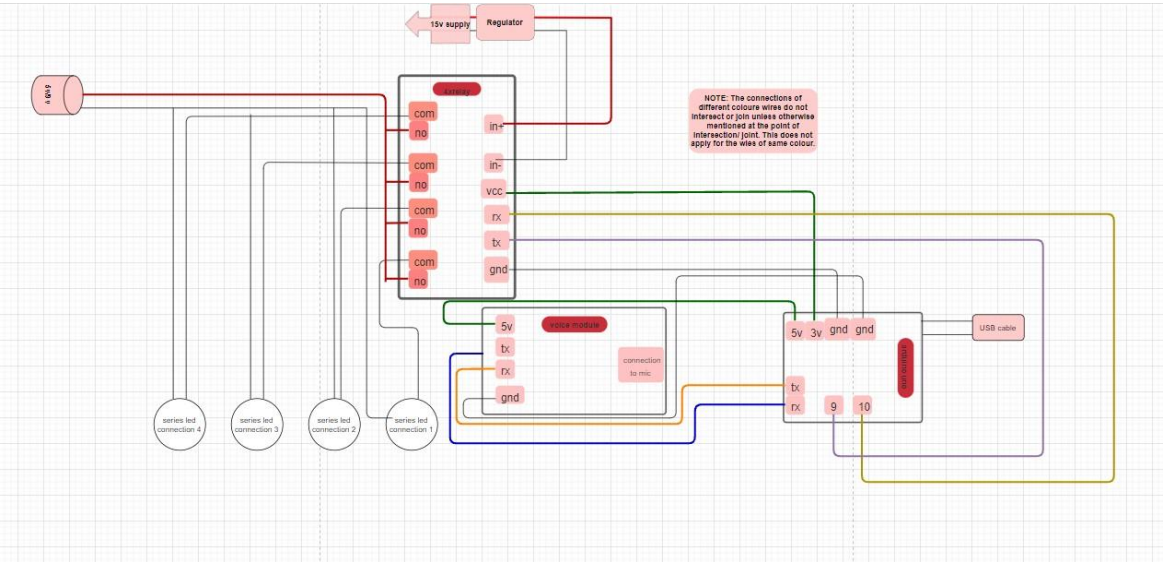


Figure 25: Electronic circuit schematic representation of Voice Module project.

Figures 26 and 27show SScom32 which was used to test the ports and connection.

Figure 26: SScom32 interface.

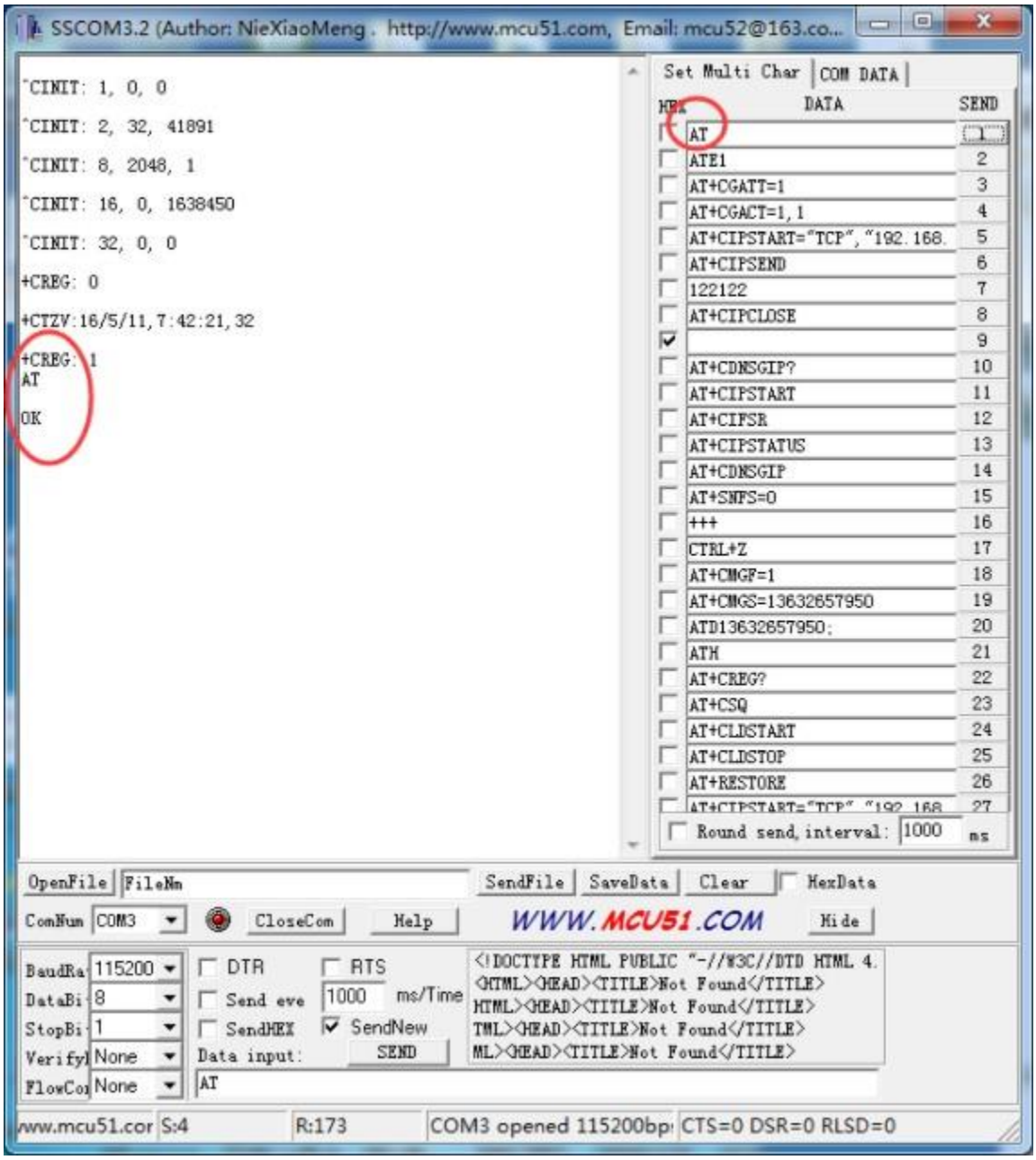


Figure 27: SScom32 interface and communication.

Multiple tools with respect to hardware and software were used as listed below:

Hardware

a. Geetech- voice module with mic
b. Arduino UNO R3
c. 4x relay with Wi-Fi module

d. 15-volt supply connected to transistor to provide only 12 volts
e. Jumper wires
f. Arduino to USB cable/ USB cable
g. Office phone for pictures and videos
h. Solder machine, solder

Software

a. Access port
b. Arduino
c. Hterm (Alternative for Access port)
d. SScom32
e. USR-TCP-232 test

Potential software for electronic layout:

a. Microsoft Visio
b. Every circuit
c. Tinker CAD
d. Proteus design suit
e. Ni Multisim
f. EasyEDA
g. KiCAD
h. Eagle- AutoCAD

Problems, solutions, and troubleshooting: Some mistakes and must avoids in the project include the following details-

a. Before testing the Arduino and access port software, make sure to troubleshoot and resolve everything.
b. Before using any software, make sure to select the correct serial port and verify this in device manager.

c. Avoid using extra files in the Arduino library if you have any extra files (such as examples or images on Arduino library then the codes would not be uploaded into the board, and this will show error).
d. Verify that the Arduino library file on the computer only contains library files or entire module files.
e. Verify that the LED function and the circuitry are sound.
f. Always connect RX to TX and TX to RX for this project (connections from voice module to USB to TTL convertor and voice module to Arduino)
g. Ensure that the jumpers connecting the relay module are wired RX to RX1 and TXto TX1.
h. The Arduino Uno's single serial port (RX and TX-UART communication protocol) cannot be used for both the voice module and the relay at the same time.
i. Use the Arduino's serial ports (RX, TX) to connect the voice module (TX, RX), and additional pins like 9, 10 to connect the RX, TX, and relay module, or the other wayaround.
j. You could also utilize a multi connector/multi communication port for RX-TX, serial, or UART connection.
k. Don't short circuit the electrical components; use the right resistor.

Troubleshooting Arduino and Access Port-

a. This is caused by a generic connection error between your computer and the Arduino and can result from many different specific problems.
b. Here are some easy things that can often fix this error:
    1. Disconnect and reconnect the USB cable.
    2. Press the reset button on the board.
    3. Restart the Arduino IDE.
    4. Make sure you select the right board in Tools > Board >, e.g. If you are using the Duemilanove 328, select that instead of Duemilanove 128. The board should say what version it is on the microchip.
    5. Make sure you selected the right port in Tools > Serial Port >. One way to figure out which port it is on is by following these steps:
        i. Disconnect the USB cable.

ii. Go to Tools > Serial Port > and see which ports are listed (e.g., COM4 COM5 COM14).

iii. Reconnect the USB cable.

iv. Go back to Tools > Serial Port >, and see which port appeared that wasn't there before.

6. Make sure digital pins 0 and 1 do not have any parts connected, including any shields.

c. If none of those work, you will want to try to isolate the issue by replacing things: try a different computer on the same Arduino, try a different Arduino on the same computer, and try using a different USB cable.

d. If the issue is with the computer:

1. Double-check all computer-related issues in the "easy fixes" list above.
2. Reinstall the IDE.
3. Reinstall the drivers.

e. If the issue is with the Arduino:

1. Double-check all board-related issues in the "easy fixes" list above.
2. Make sure the microcontroller is seated correctly.
3. You may need to burn the bootloader.
4. Replace the microcontroller if you have another one handy nearby.
5. You may have bricked your Arduino. Sorry :(

f. The Arduino IDE displays this warning when a folder which does not contain a validlibrary is found under one of the libraries folders.

g. This might be result from several different things:

1. Sometimes there is a glitch in the Arduino IDE's Library Manager which causes an empty folder to be left behind after you install or update alibrary.
2. You may have placed a folder containing a sketch or some other non-library content in one of the libraries folders.

iii. You may have attempted a manual installation of a library but added an extra subfolder. The library must be directly under the libraries folder.

h. Which of the above is the cause in your case, I can't say because I can't see your computer. The Arduino IDE provides a very helpful clue by telling you the path to the problematic folder, but you omitted that part of the warning message from yourpost so that's no help to me either.
i. The solution is to remove the problematic folder. If the folder contains nothing of value, then just delete it. If the folder contains something of value that isn't a library, then move it anywhere you like if it's not in a libraries folder. If the folder contains a library, then you'll need to install that library correctly.

Reflection: It was displayed, labelled, and packed ready for sale or use of company. This was used for the Tokyo warehouse prototype and the technology was also experimented for avionics display board system of the XP4 flying car. Valuable skills: IoT and voice recognition for aircrafts, better understanding and contribution to the avionics module forthe upcoming year as I am specializing in avionics.

3.1.4 Raspberry Pi and FlightAware Integration

Mentors for this project: Akash (University of Dhaka, Final year Robotics and Mechatronics student).

Task assigned and purpose: To monitor the flights from the office base using Raspberry Pi and FlightAware software.

Description: The Raspberry Pi is a micro-computer initially designed for education. It has all the components you would see on a normal desktop PC—a processor, RAM, HDMI port, audio output, and USB ports for adding peripherals like a keyboard and mouse.

I followed the steps as mentioned below for the for the successful display and completion of this project.

a. Pi manager was installed on the Raspberry Pi, and the required version was selected (desktop version was opted as it is easy to use).
b. Storage was selected and written.
c. Power connected to 5 volts and 2 or 3 amps of power supply; HDMI cable wasrequired to be displayed on the big screen.
d. SD card was inserted in the Raspberry Pi.
e. Set up of Raspberry Pi and Wi-Fi connection was established.
f. In order to make it easy and control the Pi as well as the operation desktop (using 1 desktop), VNC (used to control desktops and work from different desktop or give access to others to use it) was used. This was enabled in Pi desktop first > menu
   > preferences > Raspberry Pi configuration > Interface > VNC enabled > select OK.

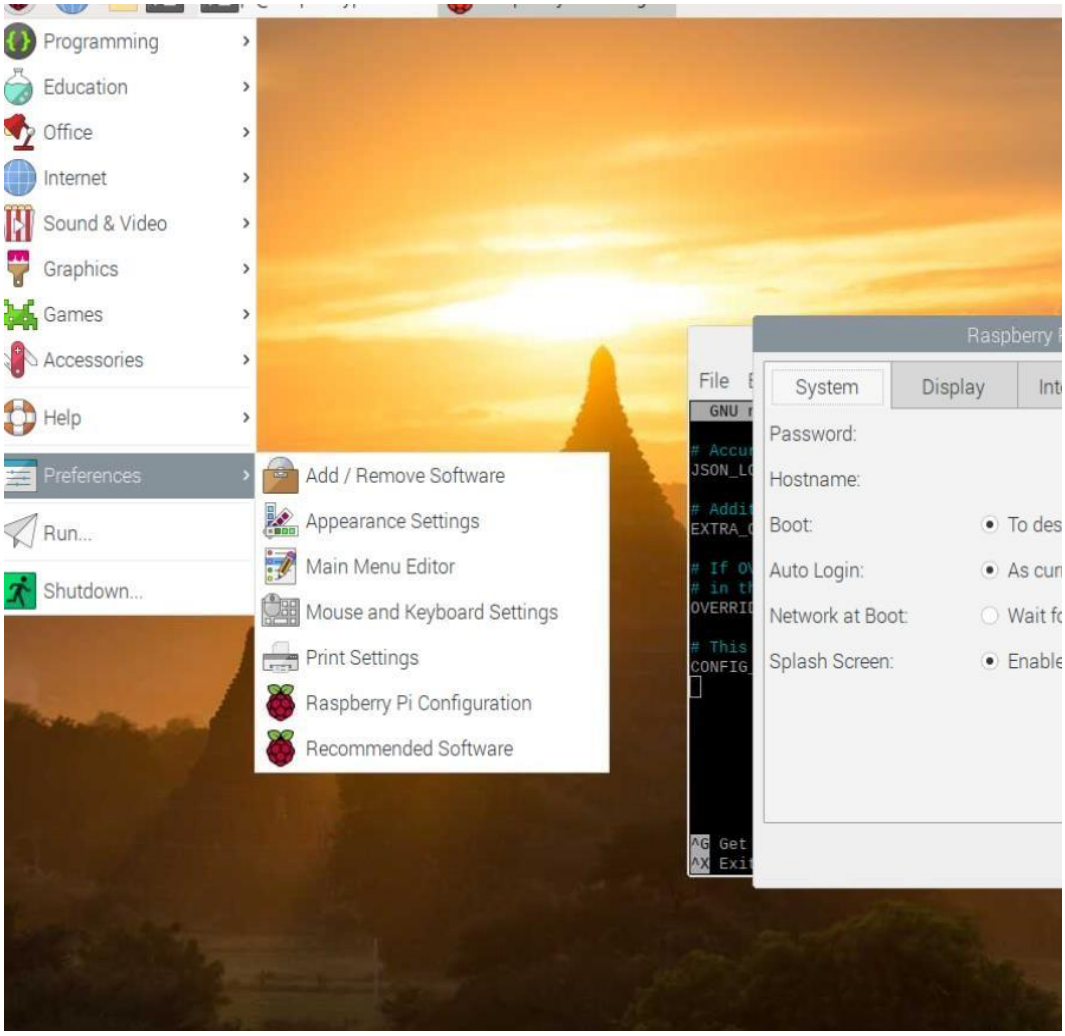


Figure 28: Raspberry Pi interface or desktop view

g. Additional support or help could be found at:
https://flightaware.com/adsb/piaware/install

Figure 29: Steps to download and install Pi Aware

h. Alternative references: https://www.youtube.com/watch?v=-7Brpm46CYU&t=89s
i. (Note: When none of the above techniques worked, I tested with general commands such as:
    a. wget https://flightaware.com/adsb/piaware/files/packages/pool/piaware/p/piaware-support/piaware-repository_5.0_all.deb

b. sudo dpkg -i piaware-repository_5.0_all.deb
c. sudo apt-get update
d. sudo apt-get install dump1090-fa

j. After reboot, I ran the code: sudo nano /etc/default/dump1090-fa
k. Simultaneously I logged into the account, and it displayed Figure 30.

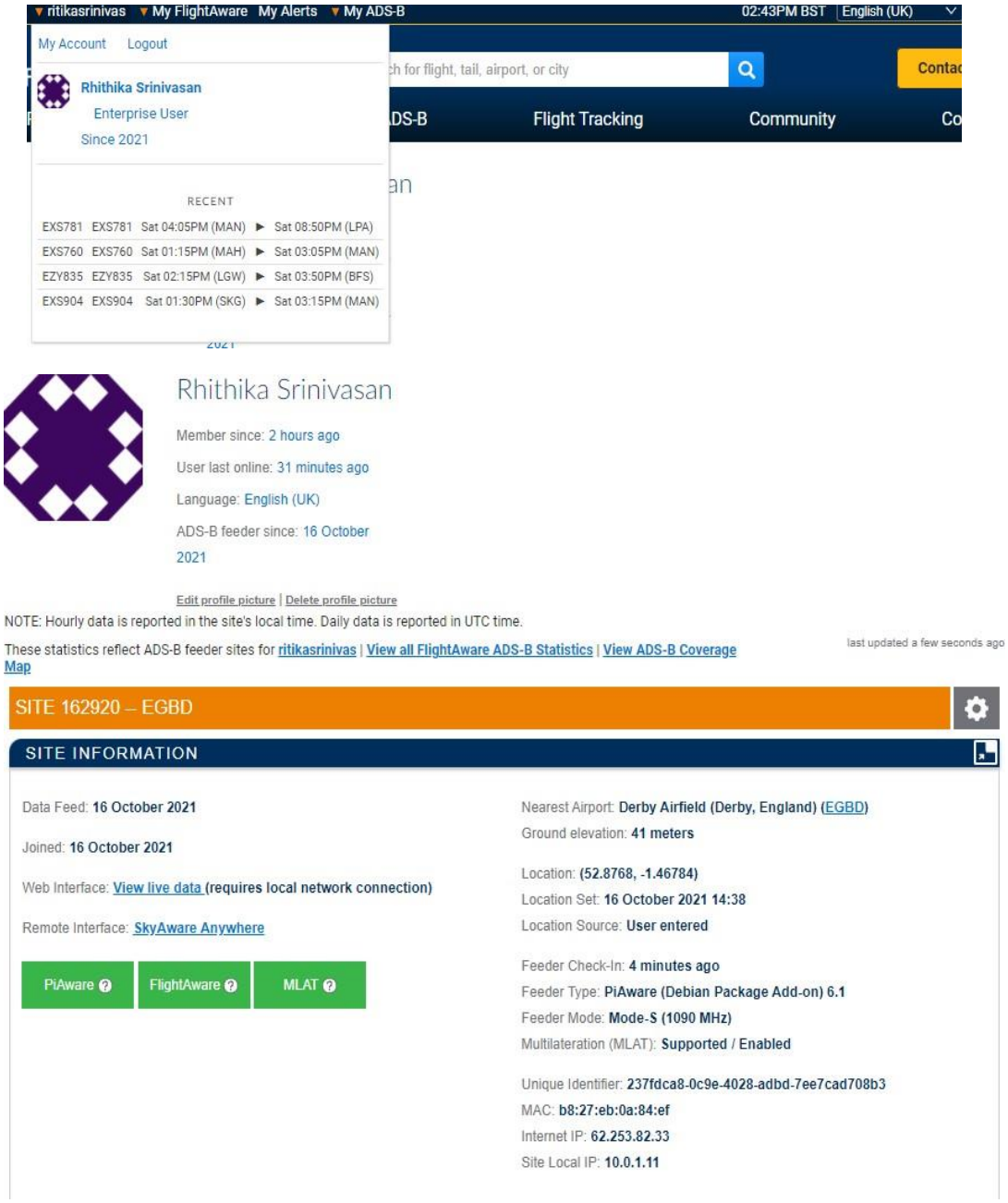


Figure 30: Pi Aware (Flight Aware on Raspberry Pi)

l. I used https://www.latlong.net/ to assess my latitude and longitude. It was important to make sure that the software was picking the right co-ordinates.
m. The IP address of Raspberry Pi desktop was similar and connected to the IP ofFlightAware (I used 10.0.1.11).
n. The table 3 shows functions performed for different codes:

To find IP address in Raspberry Pi desktop	ifconfig
To copy and paste in Raspberry Pi desktop	Ctrl+shift+copy/paste or right click to copy or paste
To see FlightAware data	ipaddress/dump1090-fa/ (for this I used 10.0.1.11/dump1090-fa/
Post reboot code	sudo apt-get install dump1090-fa

Table 3: Quick access codes and their functions for Raspberry Pi and FlightAware project.

Various tools were used to complete this project. They include:

Hardware

a. Raspberry Pi 3
b. FlightAware
c. Co-Axial cable
d. Antenna
e. HDMI Cable
f. Power supply of 5 Volts and 2 AmpsSoftware

a. Raspberry Pi 32 bit with desktop
b. Pi Imager
c. FlightAware
d. Pi Aware
e. VNC

The pictorial representation of the results of this project are as follows with description toeach picture:

VRCO was located in a remote zone as they had to deal with manufacturing and production. A few flights were recorded when tracked with antenna. As depicted in Figure 40, 41, 42, and 43, I was able to study details such as aircraft reported in various sites, hourly collection, graph coverage, positions, and the data of users closer to the base I was working from.

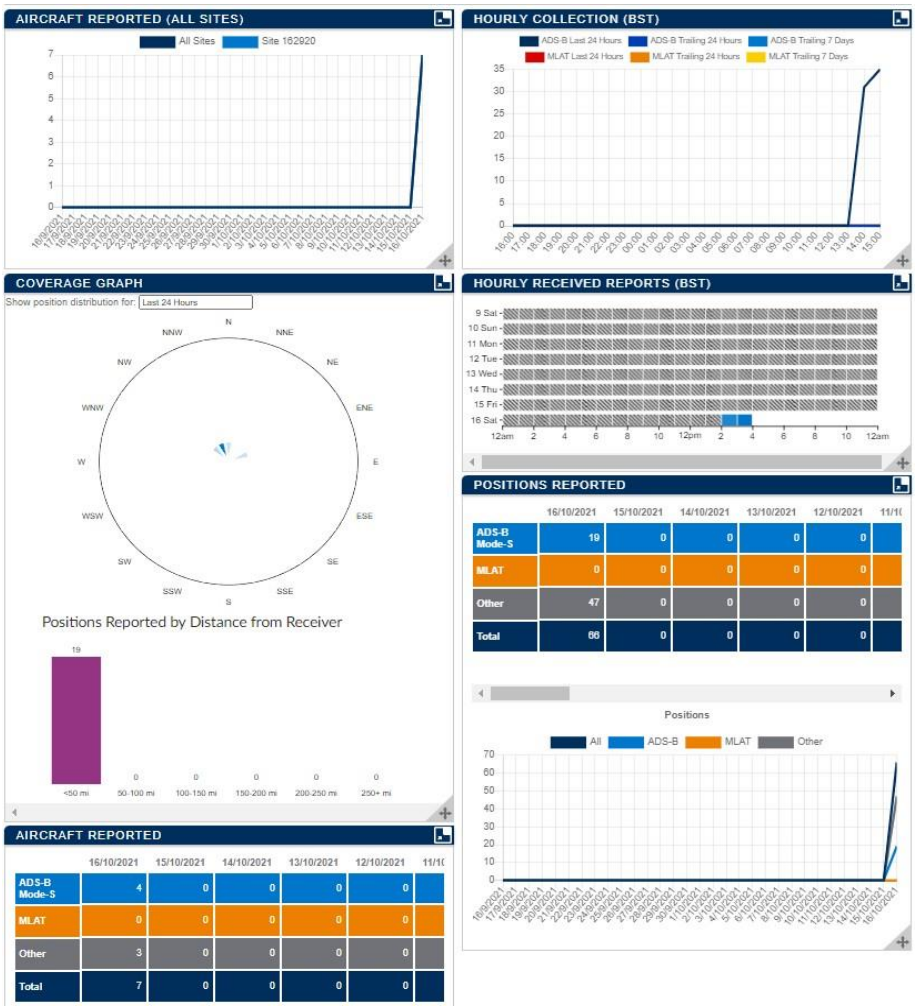


Figure 31: Details of reported aircraft, positions reported, aircrafts appeared on an hourly basis.

NEARBY SITES

DISTANCE	USER	FLIGHTS	POSITIONS	NEAREST	FEEDER TYPE	MLAT	JOINED	LAST SEEN
(viewing)	Rhithika Srinivasan	7	66	EGBD	PiAware	✗	16 Oct 2021	9 minutes ago
1	lorrainemullane	1,920	390,146	EGNX	PiAware	✓	9 May 2021	3 minutes ago
1	mossyrik78	707	60,851	EGNX	PiAware	✓	21 Jan 2021	5 minutes ago
2	Philip Ball	1,702	372,149	EGNX	PiAware	✓	6 Jan 2021	3 minutes ago
2	Phil Tillson	1,284	164,397	EGNX	PiAware	✓	25 Jan 2017	4 minutes ago
2	Chris Heath	596	36,784	EGNX	PiAware	✗	3 Oct 2021	5 minutes ago
2	CraigWalkerEndsor	1,942	489,936	EGNX	PiAware	✓	3 Oct 2018	5 minutes ago
2	Tomasz Janiqa	1,431	167,981	EGBD	PiAware	✓	23 Mar 2021	4 minutes ago
4	alternateal	838	109,894	EGNX	PiAware	✓	5 Dec 2020	4 minutes ago
4	BEN BEXSON	867	107,493	EGBD	PiAware	✓	30 May 2020	3 days ago
4	g7npw	2,079	525,571	EGNX	PiAware	✓	12 Jul 2019	5 minutes ago
6	jlitchfield1971	1,163	115,057	EGNX	PiAware	✓	22 Aug 2021	5 minutes ago
6	stuart jones	0	0	EGNX	PiAware	✗	9 Nov 2016	25 days ago
6	m0glj	1,073	150,545	EGNX	PiAware	✓	4 Sep 2018	7 hours ago
6	quentin corbett	1,858	557,203	EGNX	FlightFeeder	✓	22 Jun 2020	4 minutes ago
7	Bryan Winning	1,057	106,467	EGNX	PiAware	✓	1 Aug 2018	5 minutes ago
7	Ian Oakey	1,187	211,302	EGNX	PiAware	✓	3 Jul 2020	3 minutes ago
7	Joe Wright	1,669	422,659	EGNX	PiAware	✓	12 May 2020	4 minutes ago
7	Oddbloke	786	90,065	EGNX	PiAware	✓	6 Jul 2021	5 minutes ago
7	xraydelta101	1,662	212,880	EGNX	PiAware	✓	11 Feb 2021	5 minutes ago
8	max Spencer	2,004	378,631	EGBS	PiAware	✓	25 Apr 2021	4 minutes ago
8	bungeefield	679	36,998	EGNX	PiAware	✓	29 Dec 2020	3 minutes ago
8	Paul Hallam	2,164	592,231	EGBD	PiAware	✓	15 Mar 2020	4 minutes ago
8	Alan Swift	1,882	417,065	EGNX	PiAware	✓	20 Jan 2019	4 minutes ago
8	CrichtonGolfPilot	1,896	429,200	EGNX	PiAware	✓	4 Jul 2021	5 minutes ago

Figure 32: Indication of nearby users and flights recorded on their system, distance between us, positions, feeder type (mostly Pi and Flight Aware services were recordedeasily).

Notable examples such as in Figures 42 and 43, 44 and 45, 46 and 47 shows the Airlines Jet2 760, G-UZHF, and Jet2 904. These airlines had all the information displayed and hence I chose these as examples. It displayed the time of departure and arrival, place of

departure and arrival, duration of the journey, terminal of departure and landing, airline type, and airline specifications.

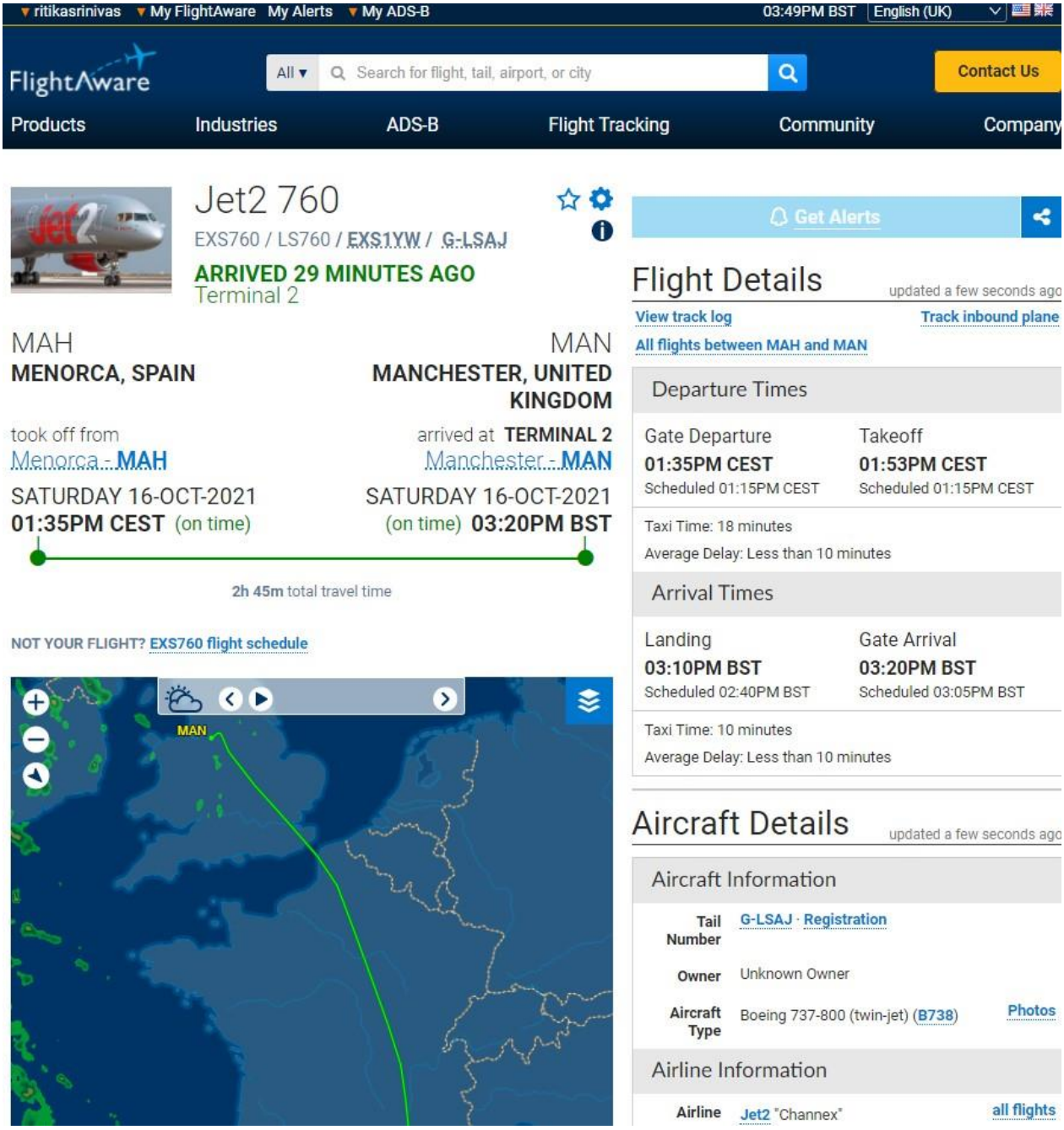
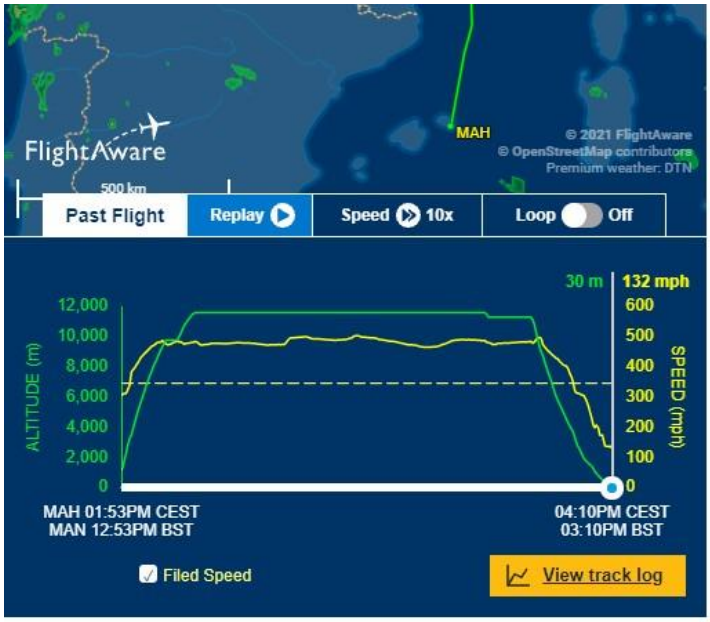
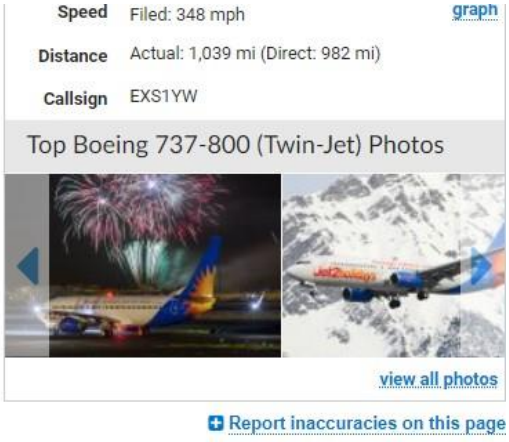


Figure 33: Tracking of Jet2 760





Upcoming Flights

Date	Departure	Arrival	Aircraft	Duration
Sunday 17-Oct-2021	*01:15PM CEST* Menorca - MAH	*03:05PM BST* Manchester - MAN	B738	*2h 50m*

Past Flights

Date	Departure	Arrival	Aircraft	Duration
Saturday 16-Oct-2021	01:35PM CEST Menorca - MAH	03:20PM BST Manchester - MAN	B738	2h 45m
Friday 15-Oct-2021	01:15PM CEST Menorca - MAH	02:56PM BST Manchester - MAN	B738	2h 41m
Sunday 10-Oct-2021	01:13PM CEST Menorca - MAH	03:03PM BST Manchester - MAN	B738	2h 50m
Saturday 09-Oct-2021	01:15PM CEST Menorca - MAH	02:52PM BST Manchester - MAN	B738	2h 37m
Friday 08-Oct-2021	01:15PM CEST Menorca - MAH	02:57PM BST Manchester - MAN	B738	2h 42m

Figure 34: Past flights and upcoming flights of the Jet2 760

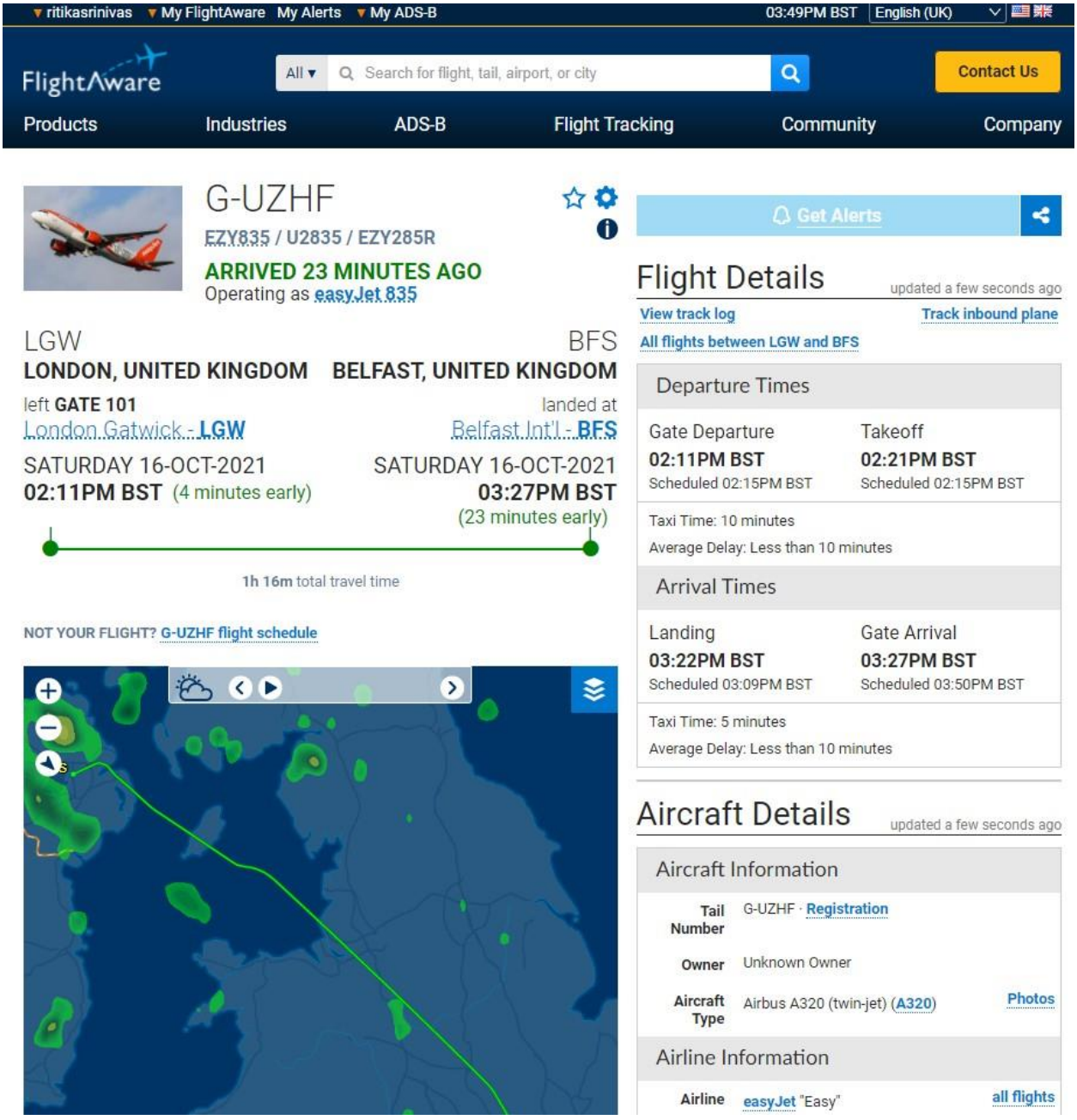


Figure 35: Tracking of G-UZHF

Report inaccuracies on this page

Upcoming Flights

Date	Departure	Arrival	Aircraft	Duration
Saturday 16-Oct-2021	*06:30PM BST* London Gatwick - LGW	*09:01PM CEST* Geneva Cointrin Int'l - GVA	A320	*1h 31m*
Saturday 16-Oct-2021	*04:25PM BST* Belfast Int'l - BFS	*05:42PM BST* London Gatwick - LGW	A320	*1h 17m*

Past Flights

Date	Departure	Arrival	Aircraft	Duration
Saturday 16-Oct-2021	02:11PM BST London Gatwick - LGW	03:27PM BST Belfast Int'l - BFS	A320	1h 16m
Saturday 16-Oct-2021	10:29AM CEST Malaga - AGP	12:14PM BST London Gatwick - LGW	A320	2h 45m
Saturday 16-Oct-2021	05:39AM BST London Gatwick - LGW	09:08AM CEST Malaga - AGP	A320	2h 29m
Friday 15-Oct-2021	09:23PM EEST Athens Int'l, Eleftherios Venizelos - ATH	11:07PM BST London Gatwick - LGW	A320	3h 44m
Friday 15-Oct-2021	02:55PM BST London Gatwick - LGW	08:33PM EEST Athens Int'l, Eleftherios Venizelos - ATH	A320	3h 38m
Friday 15-Oct-2021	09:30AM GMT Keflavik Int'l - KEF	01:28PM BST London Gatwick - LGW	A320	2h 58m

Figure 36: Past flights and upcoming flights of the G-UZHF

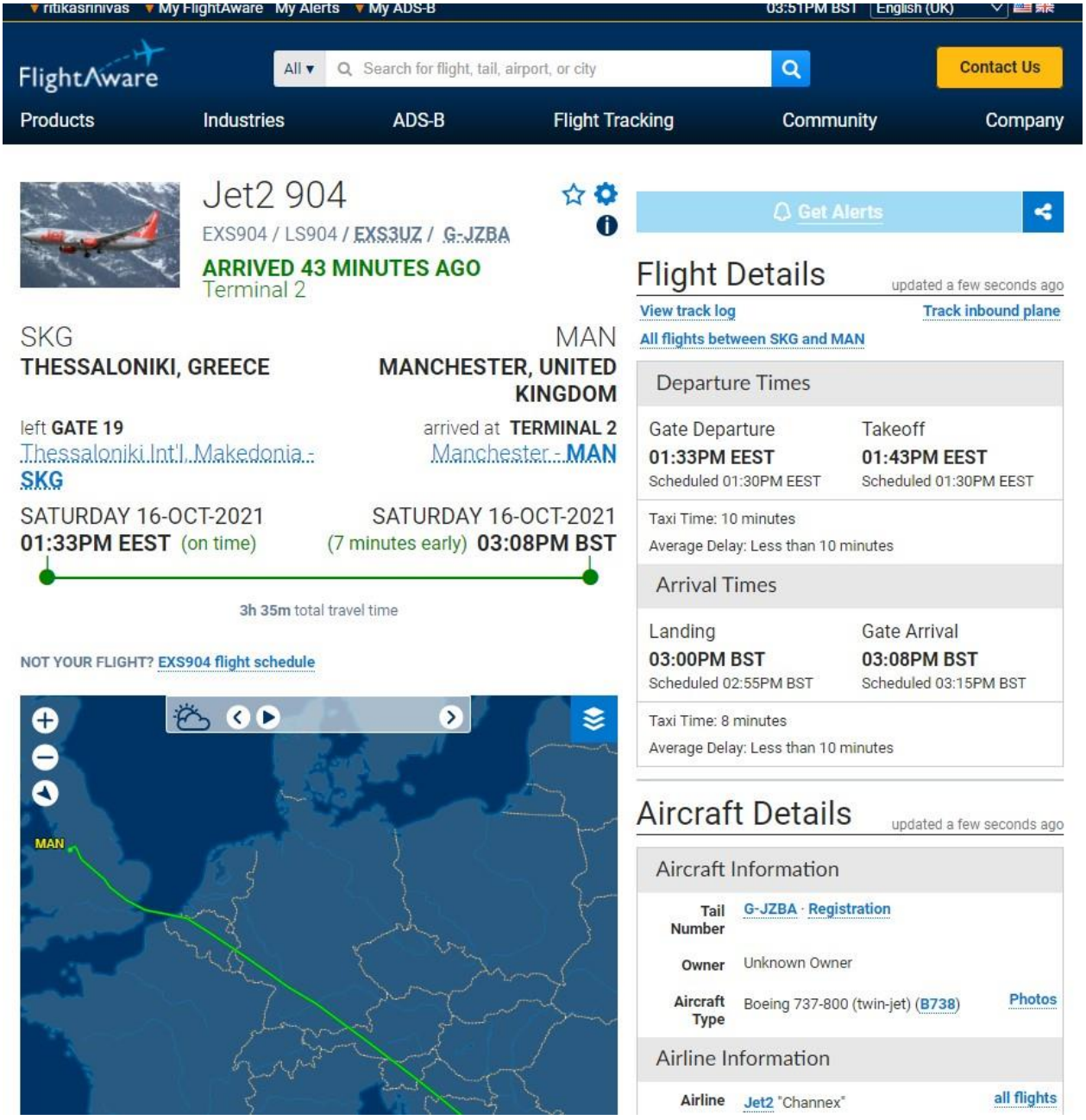
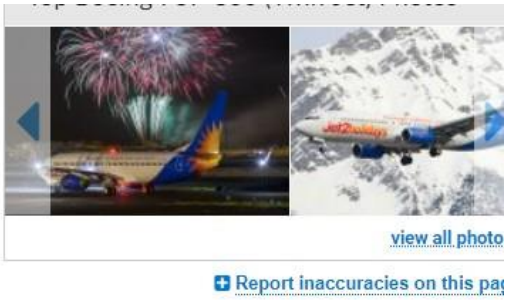


Figure 37: Tracking of Jet2 904



Past Flights

Date	Departure	Arrival	Aircraft	Duration
Saturday 16-Oct-2021	01:33PM EEST Thessaloniki Int'l, Makedonia - SKG	03:08PM BST Manchester - MAN	B738	3h 35m
Saturday 09-Oct-2021	01:25PM EEST Thessaloniki Int'l, Makedonia - SKG	03:13PM BST Manchester - MAN	B738	3h 48m
Tuesday 05-Oct-2021	01:27PM EEST Thessaloniki Int'l, Makedonia - SKG	02:55PM BST Manchester - MAN	B738	3h 28m
Saturday 02-Oct-2021	01:30PM EEST Thessaloniki Int'l, Makedonia - SKG	03:23PM BST Manchester - MAN	B738	3h 53m
Tuesday 28-Sep-2021	01:51PM EEST Thessaloniki Int'l, Makedonia - SKG	03:45PM BST Manchester - MAN	B738	3h 54m
Saturday 25-Sep-2021	01:37PM EEST Thessaloniki Int'l, Makedonia - SKG	03:15PM BST Manchester - MAN	B738	3h 38m
Thursday 23-Sep-2021	01:21PM EEST Thessaloniki Int'l, Makedonia - SKG	03:15PM BST Manchester - MAN	B738	3h 54m

View more flight history | Purchase entire flight history for EXS904

Figure 38: Past flights and upcoming flights of the Jet2 904

Problems, solutions, and troubleshooting: Some common mistake that I faced and learnt have been shared below.

a. Use appropriate SD card reader.
b. Use only a power supply of 5 volts and 3-amps and not a charger cable with theconnection to the screen port.

c. Your SD card might not be recognized by the computer (use this link to troubleshoot): https://www.youtube.com/watch?v=7ZnrlHYOKIw
d. Make sure to select the correct HDMI port on the screen or display.
e. Connect FlightAware after installing the Pi Aware and wait for 5 minutes at least(best 30 minutes) to get it working.
f. Take the antenna outside for better data and more flights display.
g. Best alternative for VNC is Any Desk (no need to be connected over same Wi-Fitoo).
h. Connect the mouse and keyboard to the raspberry pi directly and not to the pc orscreen (do not use Bluetooth ones, are not recommended and compatible).
i. VNC requires the usage of same Wi-Fi and IP address for well-established connection.

Reflection: This project was tested for the purpose of using it on the dashboard of the XP4 with the idea that it would display all the necessary and additional information to the pilot. The plan was to have an interactive dashboard system where the google or apple accounts would be connected and the music would be played according to the preferences, news of your location, arrival and departure destination details such as temperature and time would be displayed. This project contributed to that purpose and research. Skills: First experience with Raspberry Pi and this will be use for my upcoming SURE project where I will require to use the same for sensor readings.

3.1.5 Manufacturing

This was not entirely a project for me but I had the wonderful opportunity to work and be involved in the manufacturing of the XP4 first model. I was able to see the exterior shell manufacturing and the interior (seats, floor, etc.) I am restricted to share details but this was a small gist of what I experienced and the most I enjoyed.

3.1.7 Hybrid Projects of VRCO Ltd

The hybrid projects did not have mentor as most of the information was available to me. I would research and send an instruction of what was required to be done with the files and my research work to my colleague named Ms. Zuza, who was also an intern. She helped my complete my projects from home when required. The following has explanation on projects that I did from home.

3.1.6.1 Jet Engine Parts- 3D printing and working principle.

The purpose of this project was to 3D print the Jet Engine parts and to understand the basic workings of the Jet Engine. This was to contribute to the display of the company models as well as for STEM education (especially for school student). I did my research for the parts and sent the design.stl files to the office. They would print the parts and send it over to my place along with the tools required to make the surface smooth. I used the XTC 3D print kit which had the chemicals and equipment needed to produce a smooth external structure for the printed parts. The chemicals were mixed in 1:2 ratio to obtain perfect result. Figure 48 depicts the parts that I printed and Figure 49 depicts the parts that were planned to be printed.

Figure 39: Printed Jet Engine Parts

I segregated the parts and double checked the infill, time to print, layout, supports, colors, etc. Figure 50 shows the plan and my details that were constantly checked during the printing.

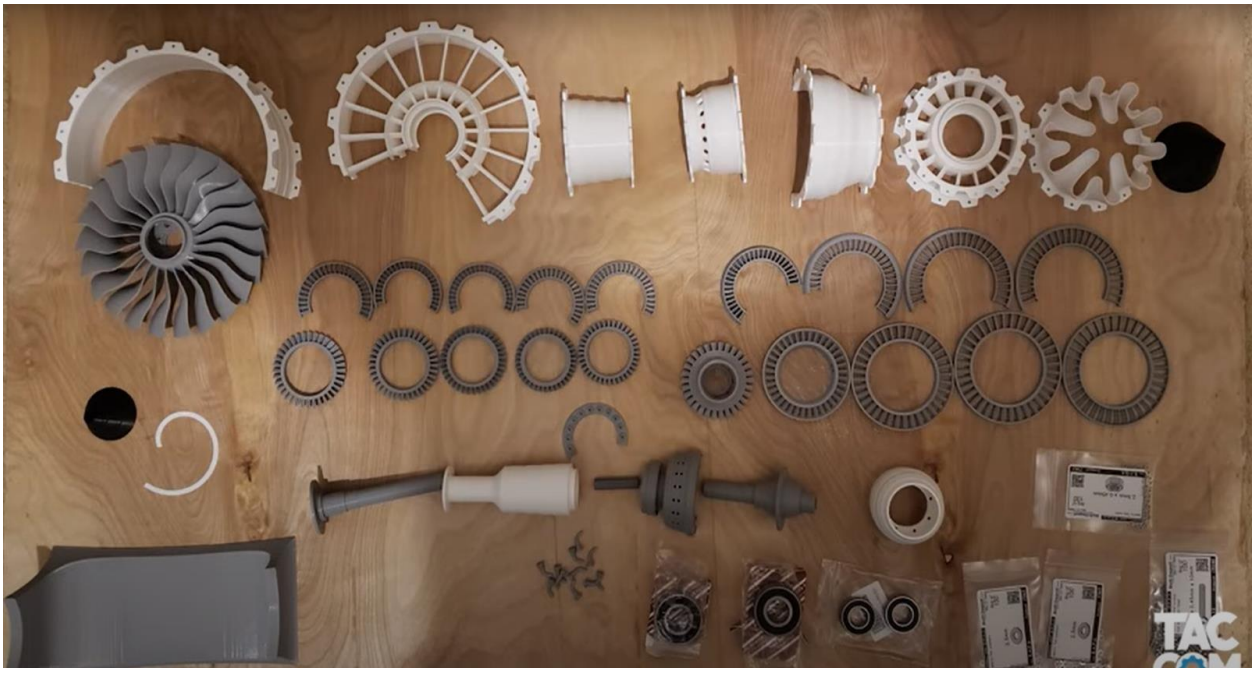


Figure 40: Planned Jet Engine Parts.

Name	Time	Color	Layer	Infill	Weight	Supported Weight	Build Note	Finished Notes
SpinnerTip	0:16	Gray	0.1	30	1	1		
LP Shaft - Mid	0:23	Grey	0.15	30	2	2		Could change to 0.2 height
Swirl Dome Plate	0:27	Grey	0.15	30	2	2		Could change to 0.2 height
SpinnerCone	3:04	Black	0.1	30	9	9		
Nozzle Cone	6:54	Black	0.1	30	21	21		
FlatStripe	0:12	White	0.1	30	1	1		Could change to 0.2 height
HPC Spool	8:43	White	0.15	30	41	41		Could change to 0.2 height maybe
Mixer	13:37	White	0.1	30	25	25		Could change to 0.15 height maybe
LPT Spool	7:11	White	0.15	30	33	40	Uses Supports	Could change to 0.2 height maybe
Comb & HPT Case	8:53	White	0.15	30	29	52	Uses combination of custom tree supports and standard supports	Could change to 0.2 height maybe
HP Core Casing	9:11	White	0.15	30	35	63	Uses Support	Could change to 0.2 height maybe
LPT Casing	12:52	White	0.15	30	50	83	Needed support...but I don't like it I'd like to try flipping this model	Some strings on the overhangs, but not too bad to remove
Fan Casing - Modded	13:54	White	0.15	30	63	63	Uses Remix File to save a TON of material ...due to not needing supports	Could change to 0.2 height maybe
Nozzle	19:54	White	0.15	30	78	97	Doesn't hollow correctly Needs specialty supports	
Fan Stator Casing	28:33	White	0.15	30	107	137	Needs specialty supports	
LP Shaft - Front	6:06	Grey	0.15	30	21	21		Could change to 0.2 height maybe
Stand	11:38	Grey	0.2	10	115	115		
LP Shaft - Aft	7:20	Grey	0.15	30	25	40	Uses Support	Could change to 0.2 height
Fuel Injector (x11)	4:29	Grey	0.05	30	3	3	Print 11 in 1 run Ensure settings are dialed in really well	
Comb Liner - Modded	6:48	Grey	0.1	30	27	26	Uses modified (remixed file for support)	Some strings on the overhangs, but not too bad to remove
HPC - S1	1:14	Grey	0.15	30	4	5	Used to dial in Stator settin	May wanna go down to 0.1 height
HPC - S2 & S3 & S4 & S5	4:27	Grey	0.1	30	14	15		
HPC - R1	1:32	Grey	0.1	30	17	18	Used to dial in Rotor settings	
HPC - R2 & R3 & R4 & R!	6:23	Grey	0.1	100	25	27		
HPT	3:12	Grey	0.1	30	17	17		
LPT - S1 & S2 & S3 & S4	8:11	Grey	0.1	30	29	32		
LPT - R1 & R2	11:18	Grey	0.1	30	21	26		
LPT - R3 & R4	12:51	Grey	0.1	30	24	32		
Fan	29:38	Grey	0.1	30	108	114	Make sure speed is low	Supports are a pain to break away at the tips. Would be nice if some gaps were there
Total	#####				**947**	**1128**		

Figure 41: Jet Engine part list, time, infill, layer, weight, support, and notes.

The engine was to be assembled and I decided the assembly parts and required components for assembly. Below is the list of dimensions of the tools for printing the part at 100%. I considered half the dimensions of the list below as I printed the parts at 50% the size. The components were:

a. 2x-6003 ball bearing (17x35x10)

b. 2x-6204 ball bearing (20x47x14)
c. 77x- A-2 hex nut (2.5x0.45mm)
d. 77x-A-2 pan head screw (2.5x0.45mm)
e. 146x-A-2 washer (2.5mm)
f. Half / for 50% print:
g. 2x- ball bearing (8.5x17.5x5)
h. 2x- ball bearing (10x23.5x7)
i. Hex nut (1.25x0.225mm)
j. Pan head screw (1.25x0.225mm)
k. Washers (1.25mm)

Once the parts were approved, they were ordered and then sent into the assembly workshop. That was my role for this project. This taught me the actual assembly and order or arrangement of a jet engine. It was a learning and development session to improve skills on the jet engine which I will learn about in detail during the final year of my course.

3.1.6.2 Telemetry Research

This project was to understand and provide details of ADSB telemetry for the company as they had the idea of incorporating it with the avionics system of the XP4 model. A report of around 2000 words was submitted to the company with explanations to topics such as MAVlink communication, various communication protocol (NETCONF/ yang, RESTCONF, gRPC, MDT), ADSB telemetry, safety management, telemetry system, and its benefits.

3.2 Raine Dev Ltd

The mentor for all my projects with Raine Dev Ltd was Mr. Oleg, Senior Designer. I reported all my projects and he monitored my progress. He would analyze and provide valuable advice, recommendations to learn the skills, and feedback on my work which I found was helpful in personal development. Looking back from my first design to the latest design, I am happy that I improved greatly. I understood the designing techniques and the beauty involved in UI/UX designing and web designing.