Air Quality, Target and Image Processing, Web Interface Test Report

Project: UAVPAYG19 **WP Name:** AQS & TAIP & WVI Testing **WP Number:** WP-AQS-TAIP-WVI-02	**Type of Test:** Unit Test	
Test Article:	**Part Number:** N/A	**Serial Number:** N/A
System Requirements: REQ-M-05 REQ-M-06 REQ-M-03 REQ-M-04 HLO-M-2	**Test Equipment:** See "equipment used" section of each test	

Queensland University of Technology
Gardens Point Campus
Brisbane, Australia, 4001.

Test Operators:	**Test Engineers:**
Ryan Brooker Alex Switala Connor Harvey	Alex Gray Alex Switala Connor Harvey Ryan Brooker
Project Manager:	**Project Supervisor:**
Marissa Bowen	Dr Felipe Gonzalez

Test Summary

This test report contains the testing of the AQS, TAIP, and WVI subsystems after integration. This document contains images and information about the integration tests performed on the subsystems. A conclusion regarding subsystems requirements has been discussed. The subsystem System Requirements have been met. A discussion to improve the design further has been made in this document.

Revision Record

Document Issue/Revision Status	Description of Change	Date	Approved
1.0	Initial Issue	28/10/2022	M.B

Table of Contents

Paragraph **Page No.**

List of Figures

Figure **Page No.**

List of Tables

Table **Page No.**

Definitions

Acronym	**Definition**
WVI	Web Visualisation and Interfaces
TAIP	Target Analysis and Image Processing
ST	Sampling Tube
AQS	Air Quality Sensor
UAV	Unmanned Aerial Vehicle
QUT	Queensland University of Technology
API	Application Programming Interface
PC	Personal Computer
Wi-Fi	Wireless Fidelity
GPS	Global Positioning System
LAN	Local Area Network
GUI	Graphical User Interface

1 Introduction

The team members of Group 19 have been appointed to research, design, plan and implement an Advance Sensor Payload (ASP) for Unmanned Aerial Vehicle (UAV) target detection and air quality monitoring in GPS denied environments. The group has committed to the specified budget whilst implementing the project requirements stated by the client. The team has also committed to meeting the deadline date specified by the client with a full functioning ASP that has been tested to ensure the client requirements have been met. This test report covers the integration tests between the AQS, TAIP and WVI subsystems.

1.1 Scope

The scope of the project is to research, plan, design, implement and test the ASP for UAV target detection in GPS denied environments. This document contains the objectives of the test, the equipment used, in depth descriptions of the tests, results, an analysis of these results and a conclusion with recommendations. The purpose of this test document see if the tests satisfies the state System Requirements/HLO's in RD-1.

1.2 Background

The Queensland University of Technology's Airborne System Lab (ASL) has commissioned the group UAVPAYG19 to design and develop a payload capable in detecting specific objects, recording air quality data to be displayed on a web interface and to pierce a ground sample. This payload is to be attached to a S500 UAV which will complete an automated flight path. The payload is mounted on the bottom of the UAV using a provided bracket. This payload must contain all components to complete its required tasks. These components are:

- Raspberry Pi 3b+
- Raspberry Pi Camera
- Pimoroni Enviro+ sensor
- DF15RSMG 360 Degree Motor

The payload is required to identify three targets, a valve (In open or closed position), a fire extinguisher and an ArUCO marker. The Pimoroni sensor is to be used to record air temperature, pressure humidity, light and potentially hazardous gas level data. This data along with a live feed of the Raspberry Pi Camera is to be visualized on a Web Interface. Lastly a soil sample must be obtained using a sampling mechanism.

2 Reference Documents

2.1 QUT avionics Documents

RD/1	UA–System Requirements	UAVPayloadTAQ System Requirements
RD/2	UA –Customer Needs	Advanced Sensor Payload for UAV Target Detection and Air Quality Monitoring in GPS Denied Environments
RD/3	UAVPAYG19-PM-PMP-03	PMP document
RD/21	UAVPAYG-19-AQS-TR-01	Air Quality Sensor Test Report
RD/23	UAVPAYG-19-TAIP-TR-01	Target Acquisition and Image Processing Test Report
RD/24	UAVPAYG-19-WVI-TR-02	Web Vision Interface Test Report

2.2 Non-QUT Documents

3 Test Objectives

The primary objective of this test report is to test if the AQS, TAIP and WVI subsystems complete the requirement listed in Table 1. The test that has been completed will provide details to see if the subsystem requirements have been met, and where they have not been met then modification of the subsystem will be required, and further retesting completed. If all the test a proven successful, then there are no further modifications needed for the AQS, TAIP and WVI subsystems.

Table 1: AQS, TAIP and WVI Subsystem Requirements

Requirement Code	**Description**
REQ-M-05	The Web Interface is required to display real time air sampling data that is recorded directly from the UAVPayloadTAQ and updated dynamically throughout the duration of the flight.
REQ-M-06	The Web Interface is required to display the images of the targets that are taken directly from the UAVPayloadTAQ and updated every time a new picture is taken.
REQ-M-03	The UAVPayloadTAQ shall communicate with a ground station computer to transmit video, target detection and air quality data.
REQ-M-04	The target identification system shall be capable of alerting the GCS of a target's type.

4 Testing

Testing has been done to ensure that the subsystem solution that is deployed adheres to the requirements and adheres to those requirements reliably. For this test report there is no hardware testing as the integration of AQS, TAIP and WVI subsystem only consist of software components. Modifications will be made to the subsystem if the subsystem does not pass the testing.

4.1 Software Test

4.1.1 Software Test 1: Sensor Readings

Sensor readings (gases, humidity, pressure, temperature, and light) are displayed on the Web interface visually.

Equipment used:

The equipment used in this test consists of the following:

- Laptop
- Wireless Router
- Safari or Google Chrome browser
- MySQL Database
- Raspberry Pi
- Enviro+

Procedure:

Following is the procedure used to conduct the test:

1. Start the MySQL Database service on the laptop
2. Start the Web Interface Server on the laptop by using the command ./node server in the server files directory.
3. Connect the Enviro+ to the Raspberry Pi
4. Run the main script on the Raspberry Pi
5. Follow the processes laid out in RD/21 to calibrate the AQS system
6. Using the browser navigate to http://localhost:3000
7. Observe the main window of the WVI to confirm that data is being collected

Results and Evidence:

The results from this test showed that as the Payload recorded AQS data and sent data to the web server. Recent AQS data is then displayed in a readable graph on the main interface page (Figure 1) with logged AQS data accessible on the AQS tab in the navigation (Figure 2). The result from this test is successful as the test has met its requirements of displaying the AQS data.

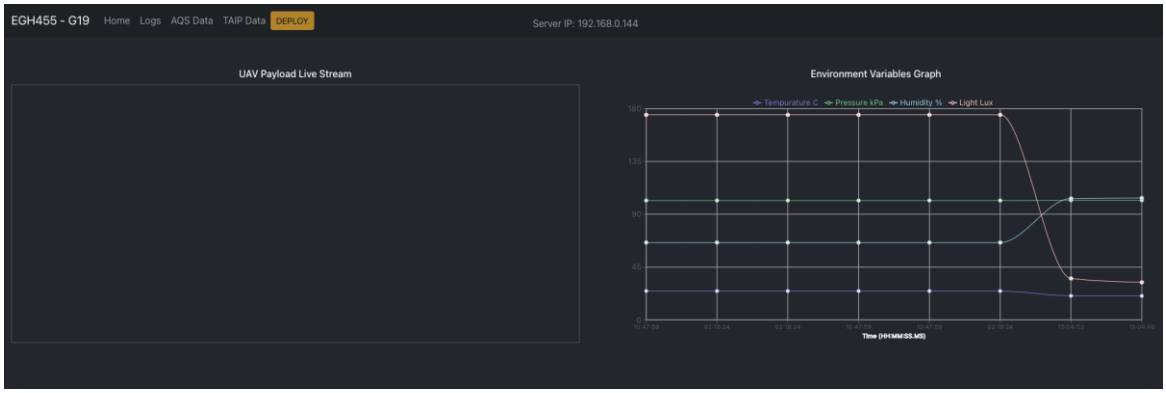


Figure 1: AQS sensor data displayed on the Web Interface

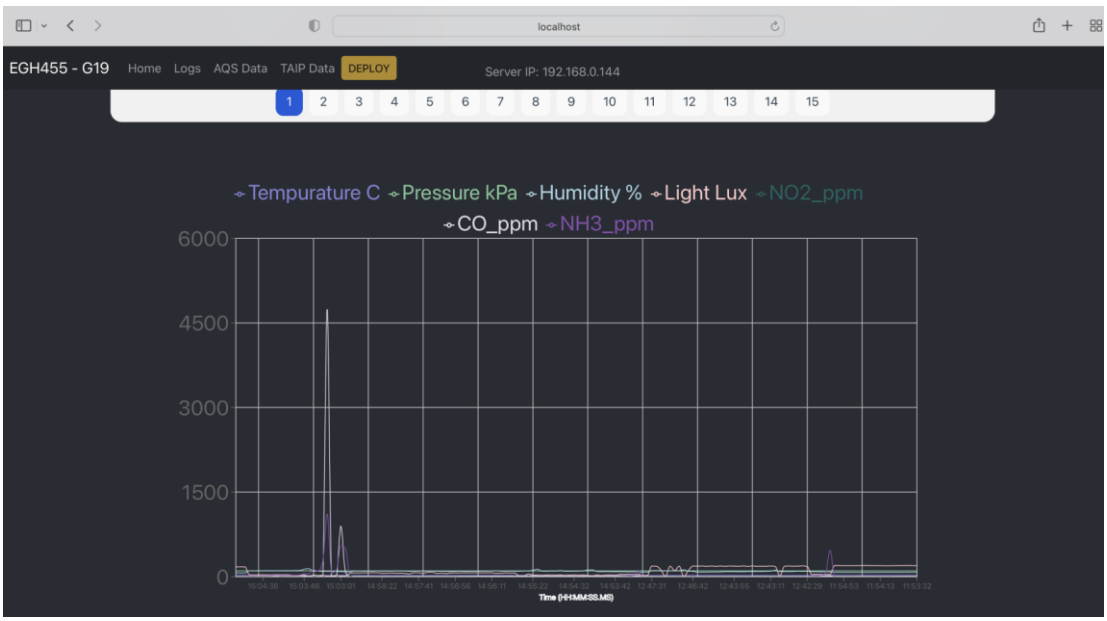


Figure 2: AQS sensor logged data displayed on AQS page

Video Evidence: https://youtu.be/LGQwPKWBF4g

4.1.2 Software Test 2: Live ArUCO and Target detection on web interface, along with alerts

This test was performed to find out if data was being successfully sent from the payload to the webserver. The video and imagery data properly needs to be sent to the server, accessible by the GCS, within 10 seconds of capture.

Equipment used:

The equipment used in this test consists of the following:

- Raspberry Pi
- Camera
- Webserver
- Ground Control Station

Procedure:

1. Set up the raspberry pi and webserver were set up
2. Attach the Pi camera to the Pi
3. Next set up the Ground Control Station
4. Then execute the main script on the raspberry pi
5. Then record the Ground Control Station screen while moving the raspberry pi camera so that it was facing towards the laptop screen which is running the test video
6. Check how the detection of the live movement movements function

Code:

The main code that was run on the raspberry pi was the IntegrationV2.py main script which took care of sending data to the web server with a one-way connection. This was run alongside the web server.

Results and Evidence:

When the camera was faced towards the test video it was shown to be able to accurately detect what was on screen. This contained:

- Type of target
- Time of capture
- Current and previous target detections
- A live stream feed
- Capture information when a target was detected

All information regarding this was captured was transmitted within 10 seconds, with a measured delay of less than 2 seconds between capture and it being viewable on the web server.

When the camera was faced towards the ArUCO marker the distance values were displayed on the screen in the form of the current Co-ordinates. Along with this, the marker was being highlighted with values being detected under it. This also was available through the live stream with the current and previous targets being known. This information can be seen in Figure 3.

When this would be seen on the screen the framerate of the video would noticeably improve as the model would be stopped on these occasions allowing for much more new data to be sent.

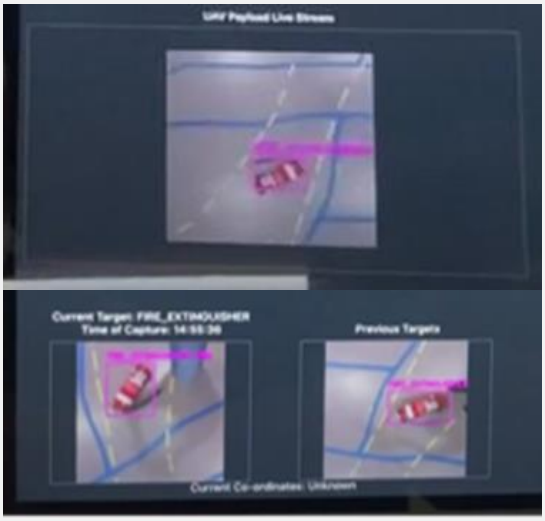
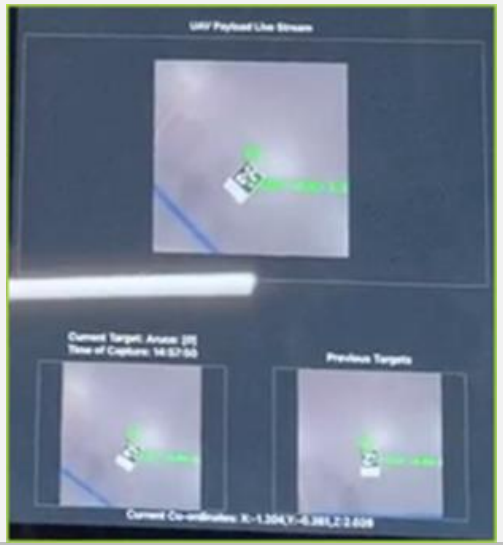
Livestream and Image Output with Target Identification and vocalisation	**Livestream and image output with Coordinate data and ArUCO Detection**
	

Figure 3 Target and ArUCO detection

Video Evidence Flight on Day: https://www.youtube.com/watch?v=0qbnhZ04XqA

4.1.3 Software Test 3: GCS video, target detection and AQ data transmitting from payload

This test aims to confirm that TAIP and AQS data is able to transmit all required data from the payload to the Ground Control Station. This will consist of displaying an image of a target in front of the camera and confirming that all data is displayed on the WVI.

Equipment used:

The equipment used in this test consists of the following:

- Laptop
- Wireless Router
- Safari or Google Chrome browser
- MySQL Database
- Raspberry Pi
- Enviro+
- Raspberry pi camera
- Fire extinguisher image

Procedure:

Following is the procedure used to conduct the test:

1. Start the MySQL Database service on the laptop
2. Start the Web Interface Server on the laptop by using the command ./node server in the server files directory.
3. Connect the Enviro+ to the Raspberry Pi
4. Run the main script on the Raspberry Pi
5. Follow the processes laid out in RD/21 to calibrate the AQS system
6. Using the browser navigate to http://localhost:3000
7. Observe the main window of the WVI to confirm that data is being transmitted

Results and Evidence:

Overall, this test was a success. This is can be seen in Figure *4* where the TAIP data can be seen on the far right confirming that this data can be transmitted. In addition to this the AQS data can be seen displayed as well. Note that the gas data is not accurate as it has not been properly calibrated. This is considered acceptable for this test as accuracy is not a concern here. As all data can be seen in the WVI and is updating this is considered a success.

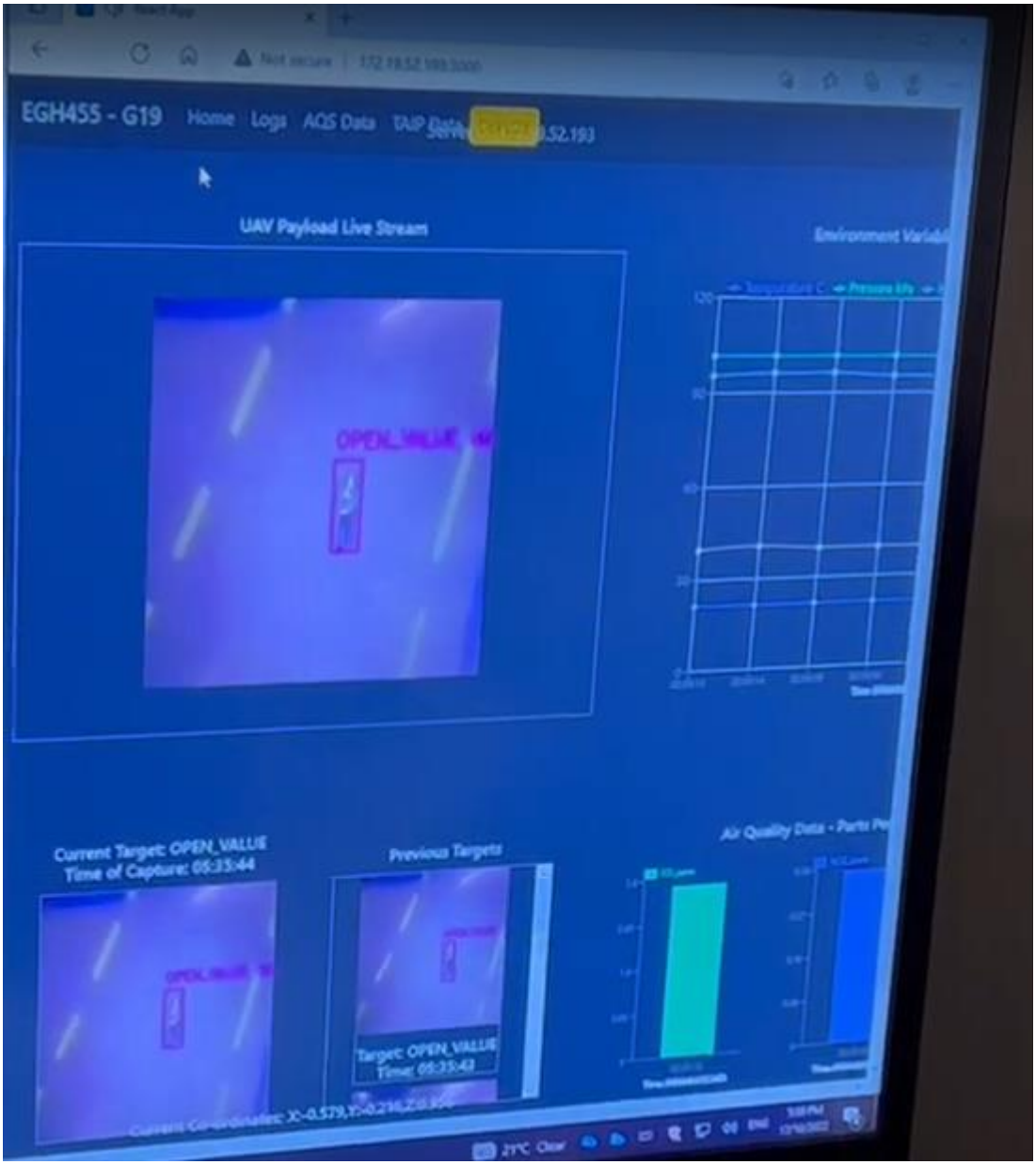


Figure 4:WVI Receiving Data

Video Evidence: https://www.youtube.com/watch?v=LGQwPKWBF4g

4.1.4 Software Test 4: Pose of Aruco markers on web interface

Live pose estimation (x, y, z) coordinates from Aruco markers are shown (when detected) on the web interface.

Equipment used:

The equipment used in this test consists of the following:

- Raspberry Pi
- Camera
- Webserver
- Ground control station

Procedure:

1. First, setup up the raspberry pi and webserver ensuring that with the camera attached to the pi
2. Then setup the GCS
3. Execute the main script (IntegrationV2.py) on the raspberry pi
4. View the screen of the GCS to see how the ArUCO mark is detected using a pre recorded video.

Code:

The main code that was run on the raspberry pi was the IntegrationV2.py main script which took care of sending data to the web server with a one way connection, this was run alongside the web server.

Results and Evidence:

Coordinate Estimation: No Marker Detected	Coordinate Estimation: Marker Detected
	
Current Co-ordinates : Unknown	Current Co-ordinates : X: -1.204 , Y: -0.381, Z:2.406

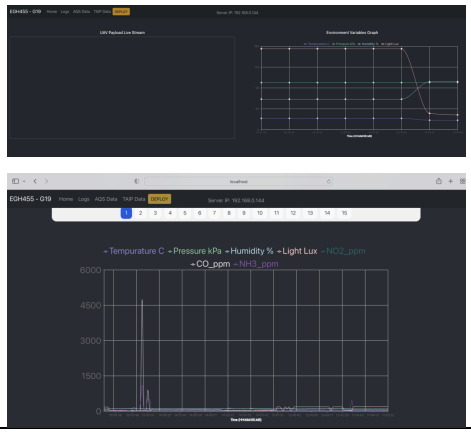
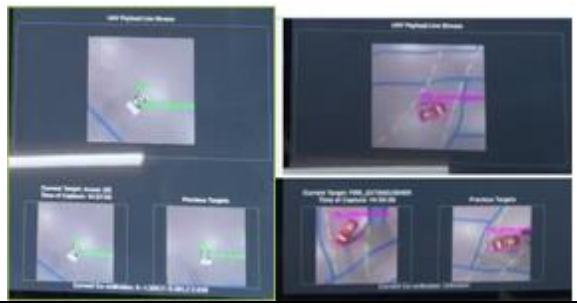
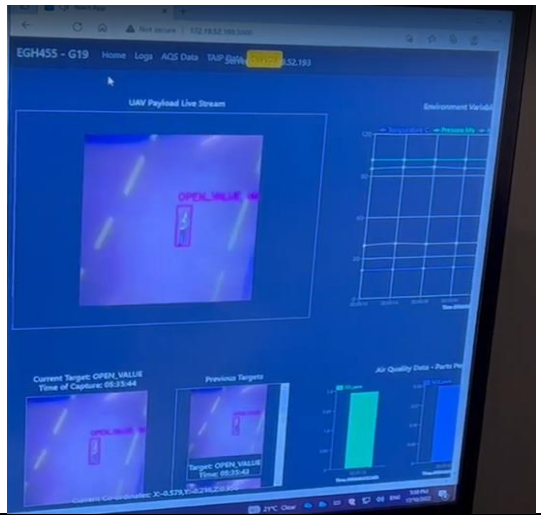
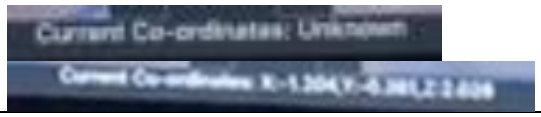
Figure 5 Co-ordinate Detection

The markers were able to be detected with the coordinates being sent to the web server during detection, the drone could only know its position however when a marker was on screen thus every time one was not on screen the current co-ordinates were displayed as being unknown. This worked well and further satisfied the requirements of needing to display the co-ordinate data to the web server.

5 Results

Table 2 provides an overview of all tests completed and what requirement the test is aimed to complete. As can be seen from the table all tests passed successfully indicating that no major design changes are required.

Table 2: Results from tests

Test Title	Result	IMG	Requirement Met
Software test 1	Pass		REQ-M-05
Software test 2	Pass		REQ-M-06 REQ-M-04
Software test 3	Pass		REQ-M-03
Software test 4	Pass		REQ-M-03

6 Analysis

All tests discussed in this report have been completed with the intension to verify that each subsystem is functional after integration occurs with the other systems.

6.1 Software Analysis

6.1.1 Software Test 1: Sensor Readings

The purpose of this test was to confirm that the AQS sensor readings are transmitted and displayed on the Web Interface. The test showed that the data is displayed across 4 graphs which are updated dynamically when the Web Interface received new data from the AQS Sensor. The test also showed that logged data could be accessed through the Web Interface.

6.1.2 Software Test 2: Live ArUCO and Target detection on web interface

The purpose of this test was to confirm that the detected targets and ArUCO marker data was able to be transmitted to, displayed, and viewed on the web interface. Along with this, proper alerts were being played and printed to the screen above the image detected when there was a detection. This test was successful as it was confirmed that targets and alerts were correctly being transmitted and displayed.

6.1.3 Software Test 3: GCS video, target detection and AQ data transmitting from payload

The purpose of this test was to confirm the capability of the TAIP and AQS data to be sent simultaneously to the GCS. There was an initial issue due to typos in the json object that was being sent that caused no data to appear in the database and therefore was not transmitted properly. However, once this issue was fixed there was no issues in transmitting the required data to the GCS. This makes this test successful.

6.1.4 Software Test 4: Live pose of ArUCO markers on web interface

The purpose of this test was to confirm that ArUCO marker XYZ coordinate data could be displayed on the web interface when a marker was in sight, and otherwise be unknownt. This was able to be confirmed as being a successful test as the data was correctly transmitted to the interface when ArUCO marker were in sight.

7 Conclusions and Recommendations

Overall, the AQS, TAIP, WVI subsystems have passed all integration testing. The testing has shown that the Web Interface is able to display real-time TAIP and AQS data, which is updated dynamically to the page. The integration testing also showed that the payload can communicate to the Web Server to transmit this data. Furthermore, the TAIP identification system integrates with the Web Interface to vocalized identified targets one the data has been received to the Web Interface. Thus, all tests are successful.

An improvement that could be made is for the AQS data being displayed on the Web Interface graphs should be displayed reading old data from the left and new data imported to the right-hand side on the graphs. This would increase the readability of the graphs; however, this does not impact the success of the requirements met.

Table 3: Requirements Met

Requirement Code	Description	Requirement Met
REQ-M-05	The Web Interface is required to display real time air sampling data that is recorded directly from the UAVPayloadTAQ and updated dynamically throughout the duration of the flight.	**Met: -** Software test 1
REQ-M-06	The Web Interface is required to display the images of the targets that are taken directly from the UAVPayloadTAQ and updated every time a new picture is taken.	**Met: -** Software test 2
REQ-M-03	The UAVPayloadTAQ shall communicate with a ground station computer to transmit video, target detection and air quality data.	**Met: -** Software test 4 and Software test 3
REQ-M-04	The target identification system shall be capable of alerting the GCS of a target's type.	**Met: -** Software Test 2

8 Appendix

No Appendix