

# **Remote Sensing for Forest Fires**

## **Verification and Validation**



**UBC Cloud Innovation Centre**

**Capstone Team CG-23**

**Mark Lee, Austin Li, Ethan McKeen, Gurjot Sandhu, Christy Zhang**

**2024-04-30**

## Table of Contents

<b>Table of Contents .....</b>	<b>1</b>
<b>1. Project Context .....</b>	<b>2</b>
<b>2. Testing Plan (Verification) .....</b>	<b>3</b>
2.1 Hardware Testing .....	3
2.2 Software Testing .....	4
<b>3. Validation Plan .....</b>	<b>6</b>
3.1 Ease of Use of User Interface (UI) and Data Readability .....	6
3.2 Scalability .....	6
3.3 Security and Privacy .....	6
3.4 Clarity of Fire Notifications .....	7

## 1. Project Context

This project aims at early prediction and detection of fires. By forecasting the likelihood of fire outbreaks in a given region, it has the capacity to alert individuals planning to travel there, allowing them to postpone their trips and thereby prevent potential injuries and economic losses. Furthermore, the project's fire detection and monitoring systems provide support to firefighters in resource allocation and enable swift responses to emergency situations. For further context about the project, refer to the Design Document.

This project has an IoT device which is a board containing a microcontroller (MCU) and LoRa transmitter. The board then connects to various sensors such as temperature, humidity, and soil moisture. A LoRaWAN gateway is the communication bridge between the IoT device and AWS. AWS stores and processes the sensor data which can then be displayed to the user through web application. All components are to be verified and validated to successfully meet requirements as well as client expectations. Verification is defined as testing the product to check that it meets the requirements set out in the Requirements Document. Validation is defined as checking that the client's expectations are met. Lastly, the following sections numerically refer to requirements and these numbers correspond to the respective requirements in the Requirement Document.

## 2. Testing Plan (Verification)

Verification of our product involves testing the hardware as well as the software components. For hardware, it is paramount to verify that the sensors provide accurate data reliably to the microcontroller (MCU), and subsequently send this data over LoRa to the LoRaWAN gateway. In addition, the gateway has to be able to send the data to the cloud. For software verification, AWS has to be able to process and store device and satellite data. Moreover, this data should be displayed on a web page where the end user can access it. More details about the tests including the expected results can be found in the Hardware and Software Testing subsections below.

### 2.1 Hardware Testing

- IoT Device
  - **Test 1:** Verify that the device can read sensor data from all connected sensors.
    - Expected Result: The device is able to get sensor data that can be processed or sent to the gateway.
    - Test Result: **Pass** - device is able to read data from both BME688 sensors which includes temperature, humidity, pressure, CO<sub>2</sub>, and volatile organic compounds (VOCs).
  - **Test 2:** Verify that the device will continue to operate with the loss of one or more sensors. This can be done by physically disconnecting sensors.
    - Expected Result: The device continues to operate and transmit data from the sensors still present.
    - Test Result: **Pass** - device continues to send data from the remaining BME688 sensor if one is disconnected.
  - **Test 3:** Verify that the device is able to read updated sensor data at fixed time intervals. The device's environment can be changed within a time interval to view change in data.
    - Expected Result: The device reads the latest sensor data at that point in time.
    - Test Result: **Pass** - device is able to read latest data, test was done by switching from indoor environment to an outdoor environment.

- **Test 4:** Verify that the power supply (battery) is able to last a minimum of a year. This will be done by taking the average current draw of the device over a 24 hour period in operation and extrapolating it to find out how long it lasts.
  - Expected Result: The power supply lasts at least a year.
  - Test Result: **Untested** - unable to test due to time constraints and continued development of the device. Will rely on power specifications of the MCU and sensors to compute this.
- **LoRa Communication**
  - **Test 1:** Ensure that the MCU is able to transmit accurate sensor data over LoRa to the gateway.
    - Expected Result: The gateway receives the same data that is being read by the MCU.
    - Test Result: **Pass** - the protocol buffer received by the gateway when decoded contains the same sensor data as the MCU.
  - **Test 2:** Ensure that MCU transmits sensor data at a fixed time interval over LoRa to the gateway.
    - Expected Result: The gateway receives updated data from the MCU at every fixed time interval.
    - Test Result: **Pass** - the device sends data at a fixed time interval.
- **Device Enclosure**
  - **Test 1:** Put the enclosure in a secure outdoor location for two days (preferably rainy weather) without any electronics.
    - Expected Result: Enclosure prevents any penetration of water as well as dust.
    - Test Result: **Partially Tested** - the enclosure was left outside in the rain for 30 minutes where no water got inside. Unable to test for full duration due to time constraints.
  - **Test 2:** Put the enclosure in a secure outdoor location for two days (preferably rainy weather) with the IoT device fitted after passing test 1.
    - Expected Result: Enclosure prevents any penetration of water as well as dust and the device is able to function.

- Test Result: Untested - due to time and weather constraints.

## 2.2 Software Testing

- Web Application
  - **Test 1:** View data displayed on the React web page and compare that with the data present in the database.
    - Expected Result: The frontend data is the most up to date data from the database and the data matches exactly.
    - Test Result: **Pass** - the data pulled by the web application is always the most up to date data and matches exactly with the database.
  - **Test 2:** Compare actual sensor location with where the sensor is visible on map on the web page.
    - Expected Result: The locations are the same.
    - Test Result: **Pass** - the marker location does appear at the location of the device.
  - **Test 3:** Verify that users are able to subscribe for fire notifications for a given region/area.
    - Expected Result: Users are added to subscription list with correct data (region/area, phone number, email).
    - Test Result: **Pass** - the logged in user can subscribe to a location, and from their user attributes, email and phone number can be retrieved
- AWS Services
  - **Test 1:** Simulate a fire at a location in software and check if notifications are sent.
    - Expected Result: The users who are subscribed for that area receive email and sms notification.
    - Test Result: **Pass** - if a fire is detected at a location, email and sms notifications are sent to subscribed users within 25 km radius.
  - **Test 2:** Verify that the AWS database is receiving data from devices at regular intervals.
    - Expected Result: There is stored data from devices that is updated at regular intervals.

- Test Result: **Pass** - AWS receives device data at the roughly the same interval it is sent from the device. (Some additional time is spent in transmitting it over LoRa and the network, and decoding the data)

### 3. Validation Plan

The validation plan encompasses tests that need to be interpreted by the team to ensure that the client's expectations and requirements are met. Thus, to assess the success of these tests we will use an evaluation criteria for each test.

#### 3.1 Ease of Use of User Interface (UI) and Data Readability

The UI is meant to be accessible to as many people as possible. As a result, UI should be easy to use and clear to a non-technical audience. In addition, the data that is presented in the UI should be easily readable. Below is the criteria for validation and the success of this meets the requirements 4.1.6 and 4.2.1.

Evaluation Criteria:

- Font size is large enough to be read on desktop and mobile interfaces
- A new user is able to navigate the web page without additional resources
- Data is presented in manner which can easily be digested by user (e.g. clearly labeled graphs or charts, use of colors to signify risk)

Results: **Meets Evaluation Criteria**

#### 3.2 Scalability

Meeting the following criteria results in meeting the requirement 4.2.4.

Evaluation Criteria:

- Variable number of sensors can be connected to the MCU with minimal changes to software
- Cloud infrastructure can scale with varying number of users as well as IoT devices

Results: **Meets Evaluation Criteria**

#### 3.3 Security and Privacy

Requirements 4.2.5 and 4.2.7

Evaluation Criteria:

- Only authorized users have the ability to add or control the IoT devices
- Database containing sensor data and subscription list is not public facing



- Encryption of data when being transmitted

Results: **Meets Evaluation Criteria**

### 3.4 Clarity of Fire Notifications

The notifications sent to users when a fire is detected should be clear and concise. This is defined in the following criteria and meeting this criteria results in the success of requirement 4.2.6.

Evaluation Criteria:

- Notification does not exceed 160 characters to keep it concise and not exceed SMS character limit.
- Notification contains the location where fire is detected, the time of detection, and a link to the web application for further details.

Results: **Meets Evaluation Criteria**