

**Mogamat Nur Ally (Engineering Consultant)**

[Nurally7@gmail.com](mailto:Nurally7@gmail.com)



# Water Management System (WMS) Beta v1.2

**13<sup>th</sup> February 2021**

**Scope:**

This brief serves as an update to the original proposal document and details the design and operation considerations of WMS Beta version 1.2.

## **OVERVIEW**

This document serves as an update on the milestones proposed in the original document as well as a detailed walkthrough of the design and operation of Beta v1.2 of the prototype. This document also covers the “Instruction manual for adding of nodes to system” milestone of the proposed requirements.

## **CHANGELOG**

1. The router chosen to provide LoRaWAN network coverage and connection to “The Things Speak Network” was changed from the LG01 to the LG02.
  - 1.1. The WMS software was written for use with LG02 gateways of firmware version LG02\_LG08-5.3.1580178039 or later as MQTT communication in previous versions of the software will not work with our node scripts.
2. Arduino IDE version was changed to Arduino IDE v1.8.15 for software development as it was found to keep system behaviour responsive and consistent.

## **MILESTONES**

The original design milestones that have been met with this prototype version are as follows:

- Gateway communicates with one Node
- Gateway communicates with two Nodes
- ThingsSpeak website and MQTT database launched
- ThingsSpeak integrated with Gateway and Control loop
- Website communicates with mobile device/Laptop
- Instruction manual for adding of nodes to system
- OFF mode – allow 6 kl a month through the control loop then fully stop thereafter
- ON mode – track water usage, update logs using Tx line from node to gateway twice a day minimise current drawn from power bank by node

The remaining milestones are as follows:

- Control loop measures and controls water consumption
- Low battery signal received at LoRa Gateway and updates website
- First verification and acceptance test passed
  - Verification Pass condition determined by Consulting Engineer
  - Acceptance Pass condition determined by User/Client.

## **SYSTEM OVERVIEW**

The LG02 router serves as the gateway through which the Arduino Uno node and TTN communicates. Upload to update TTN uses MQTT machine-to-machine communication between the node and gateway where the gateway uses an internet access point to send updates to TTN. Download to update the node status uses the TalkBack API webapp to send commands from TTN to the gateway which is then broadcasted to all nodes subscribed to the router via MQTT. Uploaded data consists of water usage readings and device statuses whereas downloaded data consists of commands sent by a system administrator. TTN collects and records the data uploaded and the node MCU collects data and actuates on command.

### **List of components:**

**1. LoRa LG02**

To be used as network gateway

**2. Arduino UNO + LoRa Shield**

To be used as node MCU processing unit

**3. 2000mAh power bank**

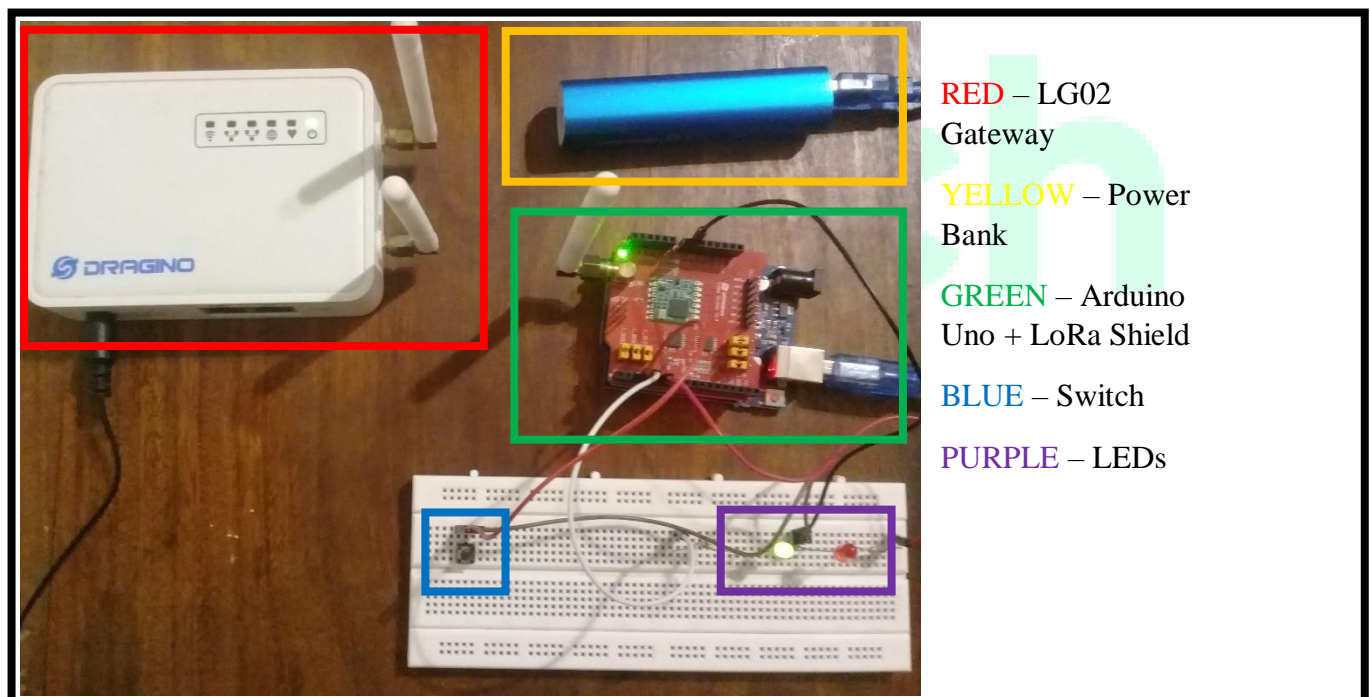
To be used as Power Supply for the Node

**4. 2 x 100Ohm resistor and 2 x LED**

To simulate the valve open and close operation

**5. 4-pin Switch**

To simulate the hydrometer measurement operation



# **SYSTEM SETUP AND INITIALISATION**

## **1. LG02 gateway**

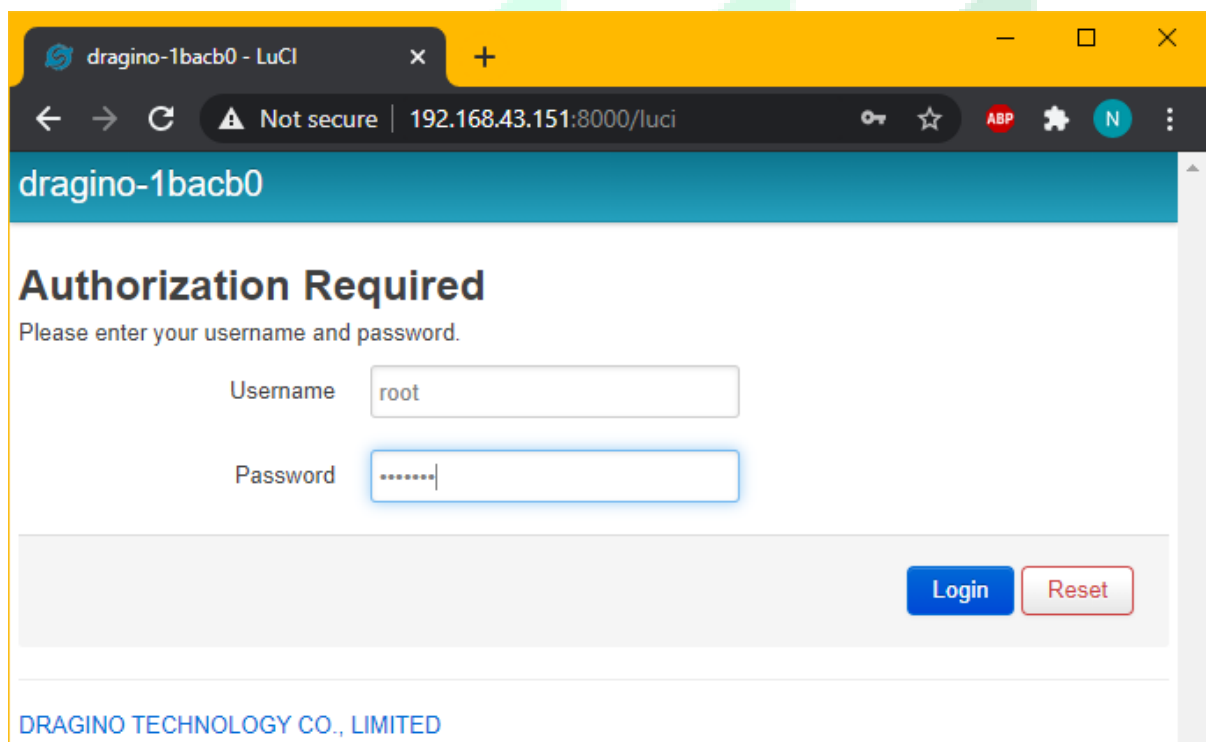
### **Access the LG02 gateway device**

Connect to the gateway via LAN cable and enter **10.130.1.1** into your browser to access the gateway settings.

Alternatively, one can also access the gateway remotely by ensuring the PC and gateway are clients on the same Wi-Fi network and entering the assigned gateway IP address into the browser using port 8000 to connect:

**FORMAT: “<IP address of gateway>:8000”**

This will be shown later in this section.

A screenshot of a web browser window displaying the login page for a Dragino-1bacb0 device. The browser's address bar shows the URL "192.168.43.151:8000/luci". The page title is "dragino-1bacb0". The main heading is "Authorization Required" with the instruction "Please enter your username and password." Below this, there are two input fields: "Username" with the text "root" and "Password" with masked characters ".....". At the bottom right of the form are two buttons: "Login" (blue) and "Reset" (red). The footer of the page reads "DRAGINO TECHNOLOGY CO., LIMITED".

dragino-1bacb0

### Authorization Required

Please enter your username and password.

Username

Password

DRAGINO TECHNOLOGY CO., LIMITED

**USERNAME: root**

**PASSWORD: dragino**

## Update the firmware to version LG08-5.3.1580178039 or later

Navigate to the “**System**” tab and access the “Backup/Flash Firmware” option.

The screenshot shows a web browser window with the address bar displaying '192.168.43.151:8000/cgi-bin/luci/admin/sy...'. The page title is 'dragino-1bacb0'. The navigation bar includes 'Status', 'System', 'Network', 'Service', and 'Logout'. The main content area is titled 'Flash operations' and has two tabs: 'Actions' and 'Configuration'. Under 'Backup', there is a 'Generate archive' button. Under 'Restore', there is a 'Perform reset' button and an 'Upload archive...' button. Under 'Flash new firmware image', there is a 'Flash image...' button. The 'Choose File' button is highlighted with a green box.

dragino-1bacb0 Status System Network Service Logout

### Flash operations

Actions Configuration

#### Backup

Click "Generate archive" to download a tar archive of the current configuration files.

Download backup: [Generate archive](#)

#### Restore

To restore configuration files, you can upload a previously generated backup archive here. To reset the firmware to its initial state, click "Perform reset" (only possible with squashfs images).

Reset to defaults: [Perform reset](#)

Restore backup: [Choose File](#) No file chosen [Upload archive...](#)

Custom files (certificates, scripts) may remain on the system. To prevent this, perform a factory-reset first.

#### Flash new firmware image

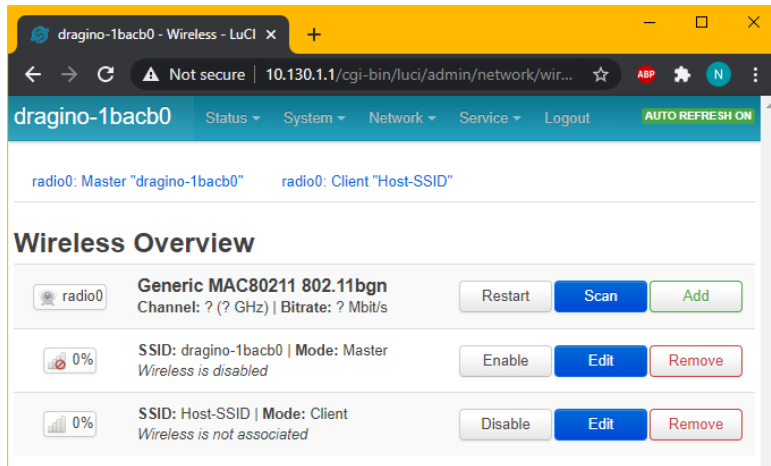
Upload a sysupgrade-compatible image here to replace the running firmware. Check "Keep settings" to retain the current configuration (requires a compatible firmware image).

Image: [Choose File](#) No file chosen [Flash image...](#)

Click on “**Choose file**” and select the update image from your computer’s local directory. Then click on “**Flash image...**”. Wait for the system to update and restart then proceed with the following steps.

## Connect to a Wi-Fi network

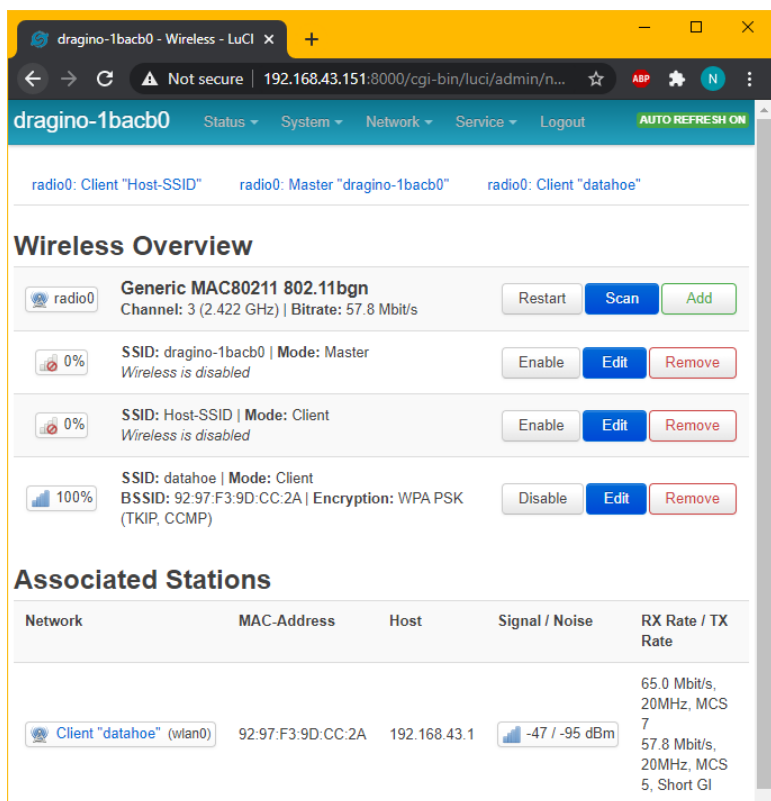
Navigate to the “**Network**” tab and select the “**Wireless**” option.



Select the “**Scan**” option to search for available Wi-Fi networks then select “**Join Network**” on your Wi-fi Network and enter the passphrase associated with your network.

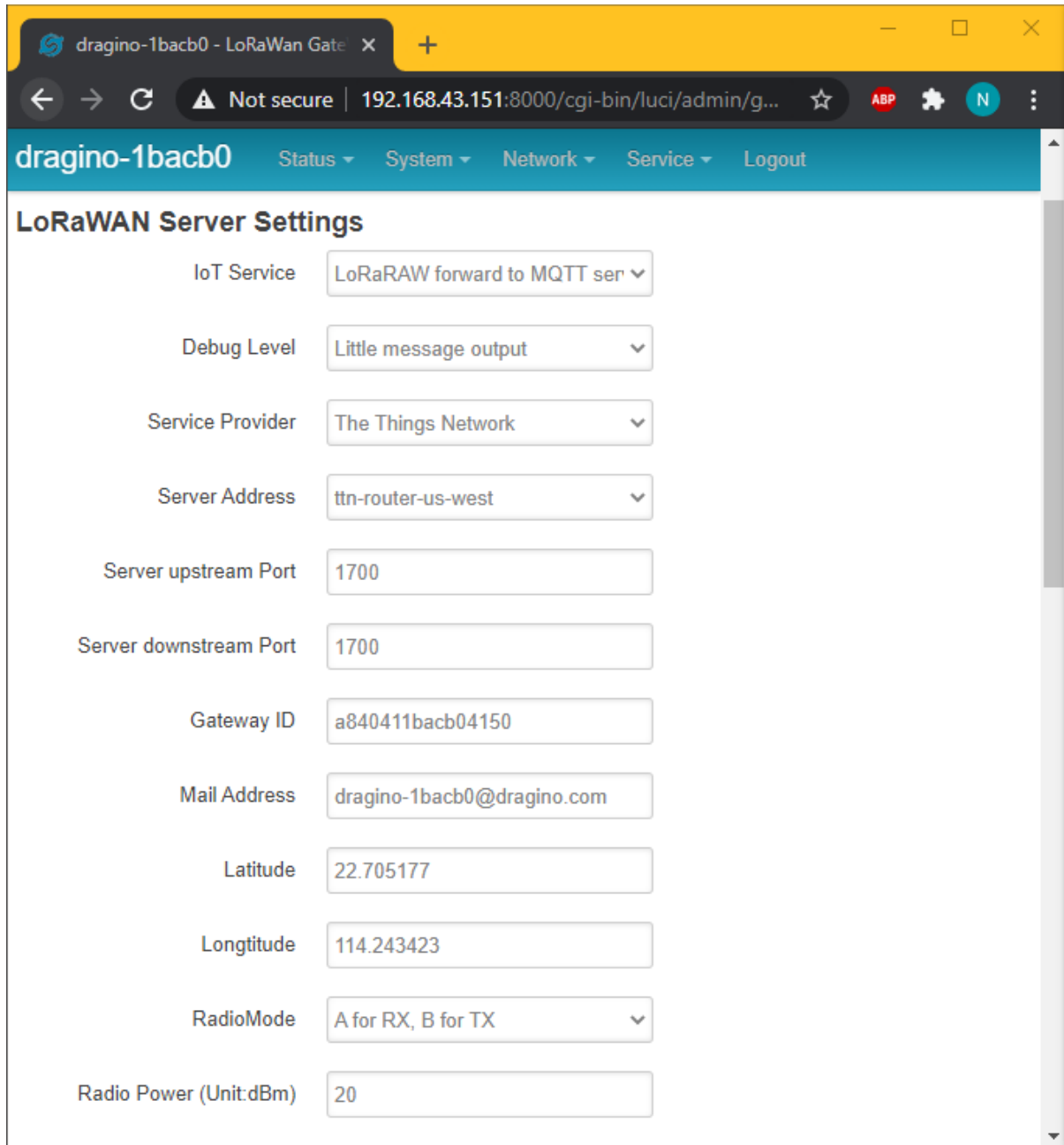
Once you have Saved and Applied your settings, navigate back to the “**Wireless Overview**” page and disable “**Host-SSID**” to allow your gateway to become a client on the new Wi-Fi network.

\*Once you have determined the IP address of the gateway, the gateway can now be accessed wirelessly after this step



### Change gateway settings to support MQTT format communication

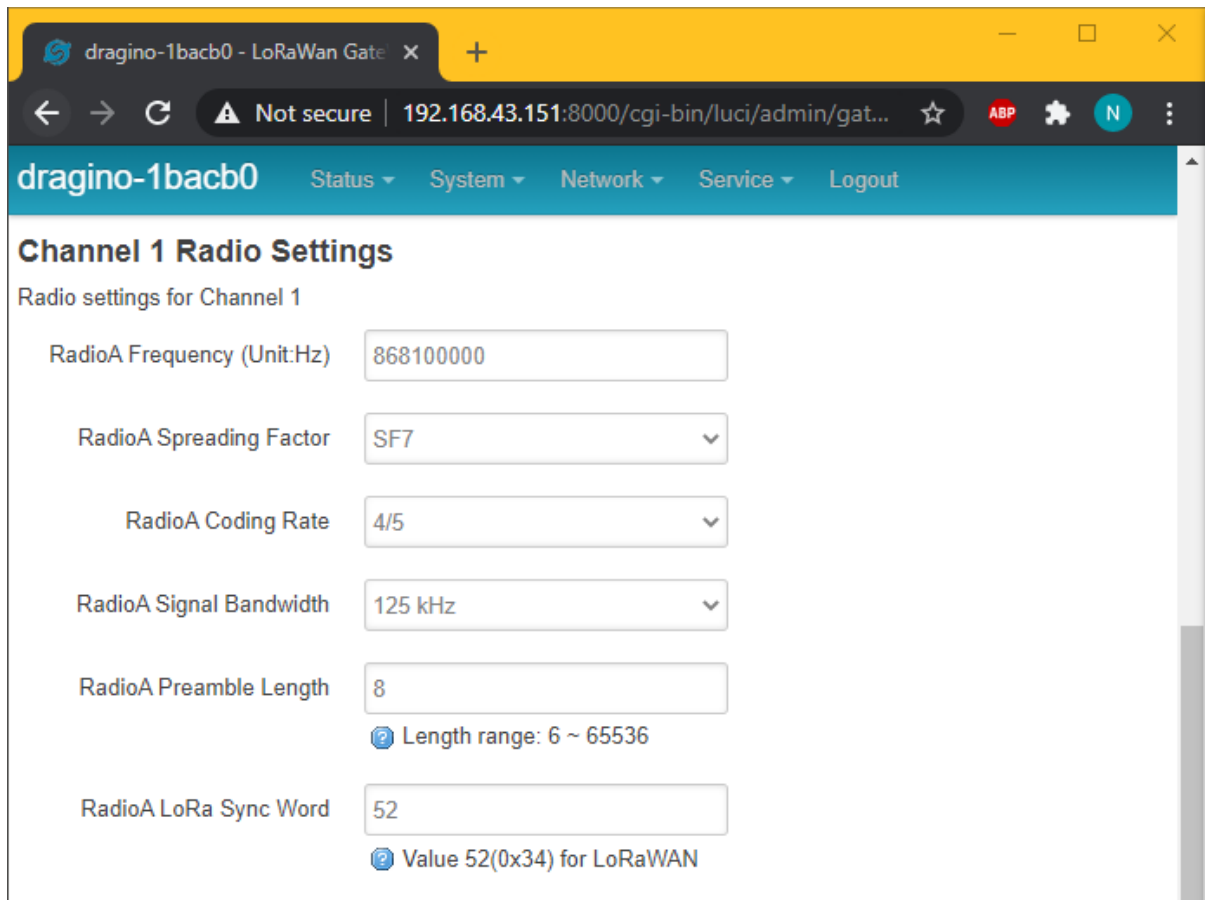
Navigate to the “**Service**” tab and select the “**LoRaWan Gateway**” option. Then change the “**IoT Service**” option to “**LoRaRAW forward to MQTT server**”, change the “**Service Provider**” to “**The Things Network**” and the “**Server Address**” to “**ttn-router-us-west**”



The screenshot shows a web browser window with the address bar displaying "192.168.43.151:8000/cgi-bin/luci/admin/g...". The page title is "dragino-1bacb0 - LoRaWan Gate". The navigation bar includes "Status", "System", "Network", "Service", and "Logout". The "Service" tab is selected, and the "LoRaWAN Server Settings" section is visible. The settings are as follows:

Field	Value
IoT Service	LoRaRAW forward to MQTT ser
Debug Level	Little message output
Service Provider	The Things Network
Server Address	ttn-router-us-west
Server upstream Port	1700
Server downstream Port	1700
Gateway ID	a840411bacb04150
Mail Address	dragino-1bacb0@dragino.com
Latitude	22.705177
Longitude	114.243423
RadioMode	A for RX, B for TX
Radio Power (Unit: dBm)	20

On the same page, ensure that both Channel Radio Settings have the following settings:



The screenshot shows a web browser window with the address bar displaying "192.168.43.151:8000/cgi-bin/luci/admin/gat...". The page title is "dragino-1bacb0". The navigation menu includes "Status", "System", "Network", "Service", and "Logout". The main content area is titled "Channel 1 Radio Settings" and contains the following settings:

Radio settings for Channel 1	
RadioA Frequency (Unit:Hz)	868100000
RadioA Spreading Factor	SF7
RadioA Coding Rate	4/5
RadioA Signal Bandwidth	125 kHz
RadioA Preamble Length	8 <small>? Length range: 6 ~ 65536</small>
RadioA LoRa Sync Word	52 <small>? Value 52(0x34) for LoRaWAN</small>

A-tech



Navigate to the “**Service**” tab and select the “**MQTT Menu**” option. On the MQTT Configuration page, ensure that the gateway has the following settings:

\*The “**Client ID [-i]**” has to be changed to the unique ID of the gateway

MQTT Server Profile	<div>ThingSpeak</div>		
Broker Address [-h]	<div>mqtt.thingspeak.com</div>		
Broker Port [-p]	<div>1883</div>		
User ID [-u]	<div>wmsdragino1</div>	Password [-P]	<div>C261JAMH5B3259U0</div>
Client ID [-i]	<div>dragino-1bacb0</div>		

### Publish

Enable Publish	<input checked="" type="checkbox"/>
Quality of Service [-q]	<div>0</div>
Topic Format [-t]	<div>channels/CHANNEL/publish/WRITE_API</div>
Data Format [-m]	<div>DATA&amp;status=MQTTPUBLISH</div>

### Subscribe

Enable Subscribe	<input checked="" type="checkbox"/>
Quality of Service [-q]	<div>0</div>
Topic Format [-t]	<div>status=MQTTSUBSCRIBE</div>

Save

Save&Apply

Cancel

A-tecn

Navigate to the “**Service**” tab and select the “**HTTP/HTTPS**” option. Enter the following link into the “**Downlink URL**”:

[https://api.thingspeak.com/talkbacks/41716/commands/execute.json?api\\_key=X4JDJ1DYNP LI2F1Q](https://api.thingspeak.com/talkbacks/41716/commands/execute.json?api_key=X4JDJ1DYNP LI2F1Q)

Also ensure that the General Settings are as follows:

The screenshot shows a web browser window with the address bar displaying "192.168.43.151:8000/cgi-bin/luci/admin/gat...". The page title is "dragino-1bacb0 - HTTP / HTTPS". The navigation bar includes "Status", "System", "Network", "Service", and "Logout". The main heading is "HTTP / HTTPS" with the subtitle "http/https connection to IoT Server". Under the "General Settings" section, the following options are visible:

- Enable SSL Connection**: ☒
- Enable HTTP Downlink**: ☒
  - ☒ Forward downlink data via LoRa
- Downlink URL**:
- Downlink Datatype**:
- Downlink Parameter**:
- Downlink Poll Interval**: 
  - ☒ unit:seconds.

At the bottom right, there are three buttons: "Save & Apply" (blue), "Save" (green), and "Reset" (red).

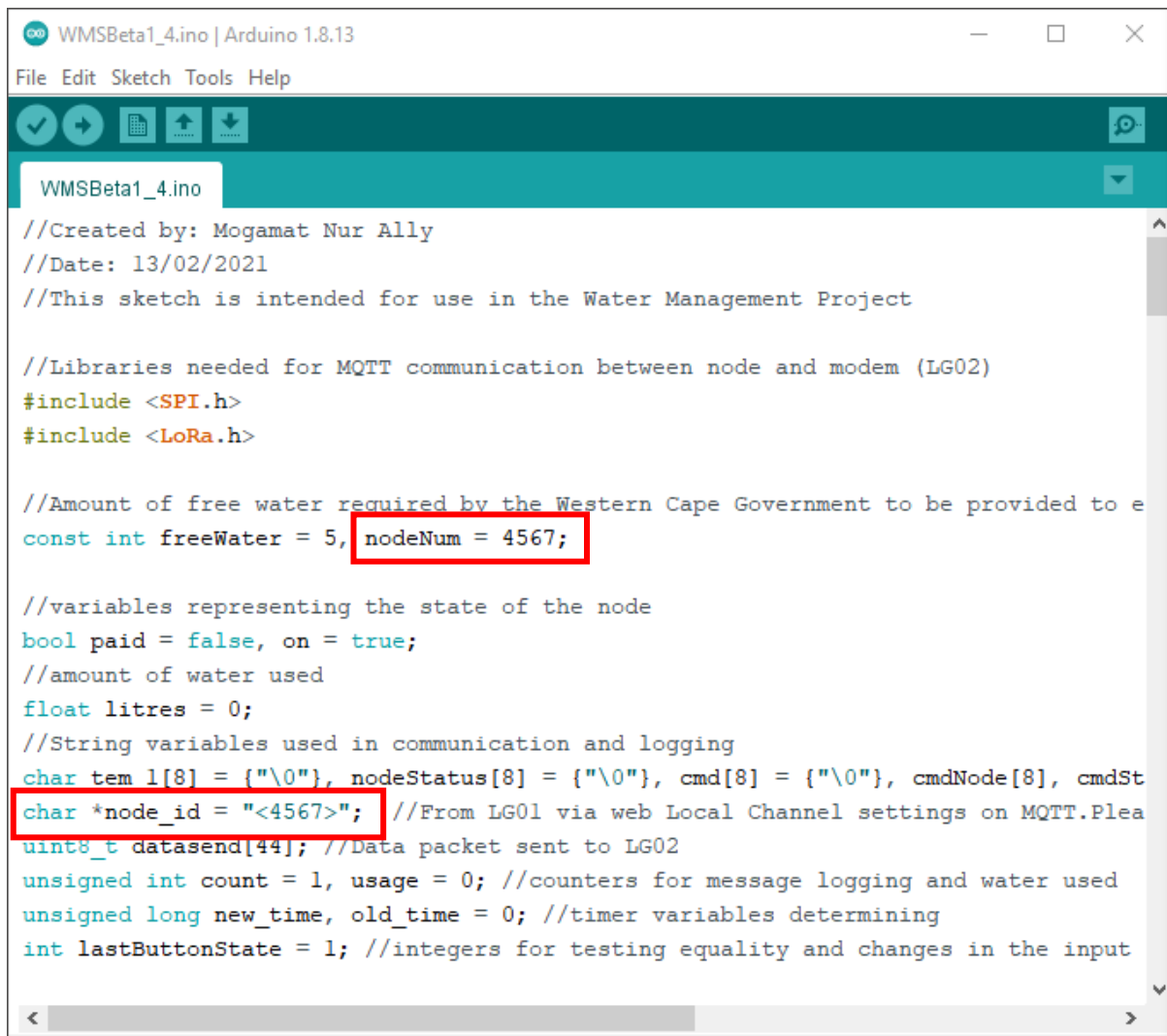
## 2. Arduino Uno + Lora Shield node

### Connect to and Name the Node

Connect to the serial port of the node via USB. Open the **WMSBeta1\_4.ino** sketch in Arduino.

Assign a number to the “**nodeNum**” and “**\*node\_id**” variables, these numbers must be identical and will become the name of the node.

Then Upload the sketch to the Arduino Uno + LoRa Shield (ensuring that the COM# of the Arduino IDE is correctly matched to the USB port, this setting can be changed and checked under the “**Tools**” tab)



```
WMSBeta1_4.ino | Arduino 1.8.13
File Edit Sketch Tools Help

WMSBeta1_4.ino
//Created by: Mogamat Nur Ally
//Date: 13/02/2021
//This sketch is intended for use in the Water Management Project

//Libraries needed for MQTT communication between node and modem (LG02)
#include <SPI.h>
#include <LoRa.h>

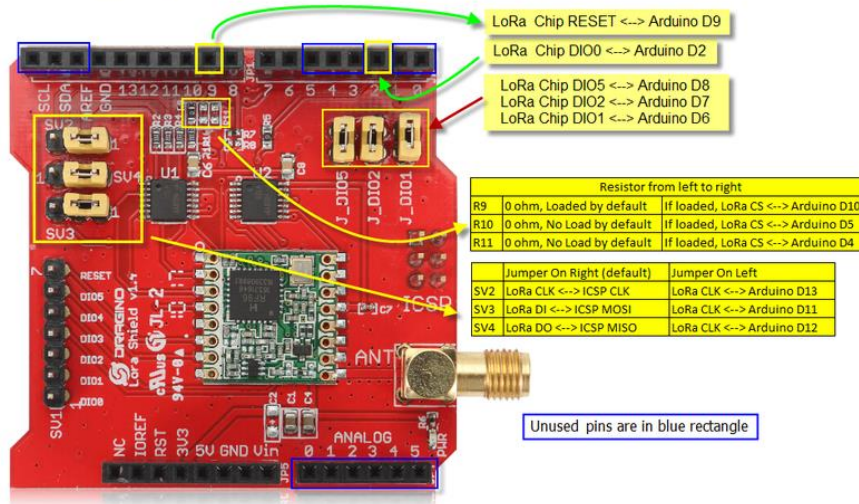
//Amount of free water required by the Western Cape Government to be provided to e
const int freeWater = 5, nodeNum = 4567;

//variables representing the state of the node
bool paid = false, on = true;
//amount of water used
float litres = 0;
//String variables used in communication and logging
char tem l[8] = {"\0"}, nodeStatus[8] = {"\0"}, cmd[8] = {"\0"}, cmdNode[8], cmdSt
char *node_id = "<4567>"; //From LG01 via web Local Channel settings on MQTT.Plea
uint8_t datasend[44]; //Data packet sent to LG02
unsigned int count = 1, usage = 0; //counters for message logging and water used
unsigned long new_time, old_time = 0; //timer variables determining
int lastButtonState = 1; //integers for testing equality and changes in the input
```

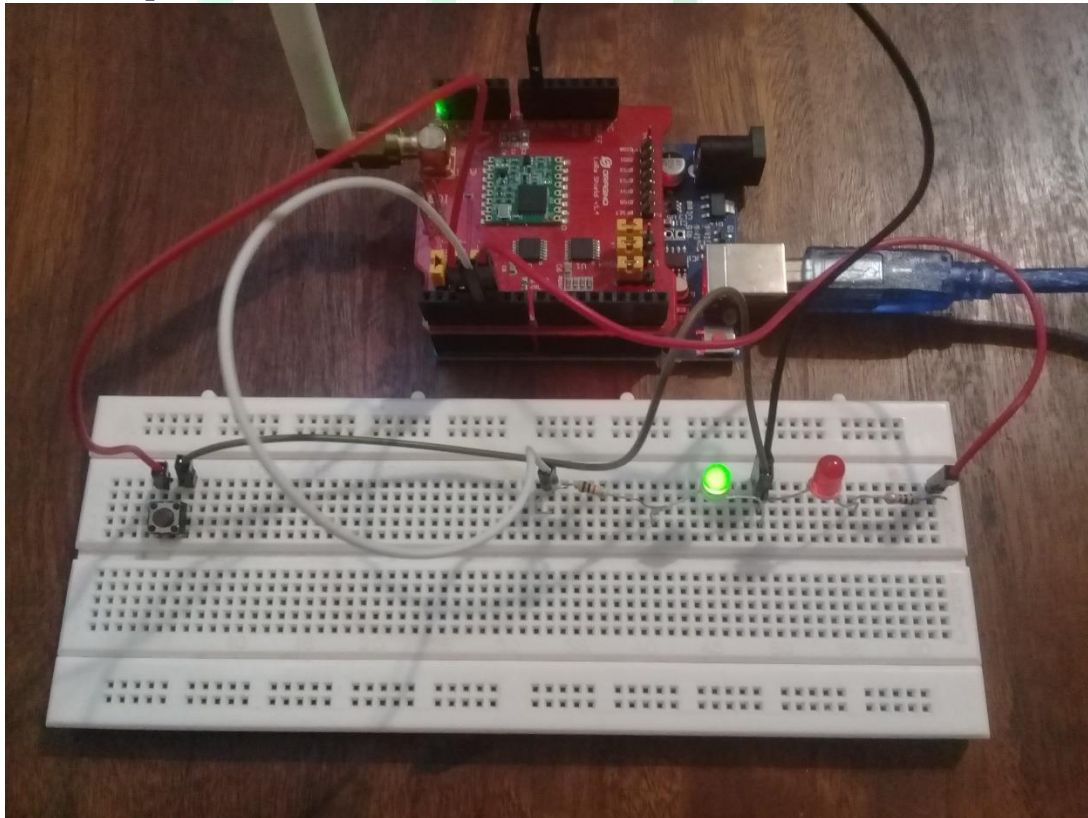
## Connect the switch and LEDs

Using the diagrams below as a guide, connect the positive end of the switch to **Digital pin 3** and the negative end to the on-board **GND** pin. Connect the positive end of the series resistor + Green LED combination to **Digital pin 4** and the negative end to the other available on-board **GND** pin. Connect the positive end of the series resistor + Red LED combination to **Digital pin 5** and the negative end to the other available on-board **GND** pin.

### Pin Mapping For LoRa



### Node Setup



### 3. Adding a new node to The Things Network

#### Create a new channel to collect and display the node data

Open your browser and navigate to thingspeak.com and log in using the following details:

**EMAIL:** [wmsproject001@gmail.com](mailto:wmsproject001@gmail.com)

**PASSWORD:** Wmsp0786

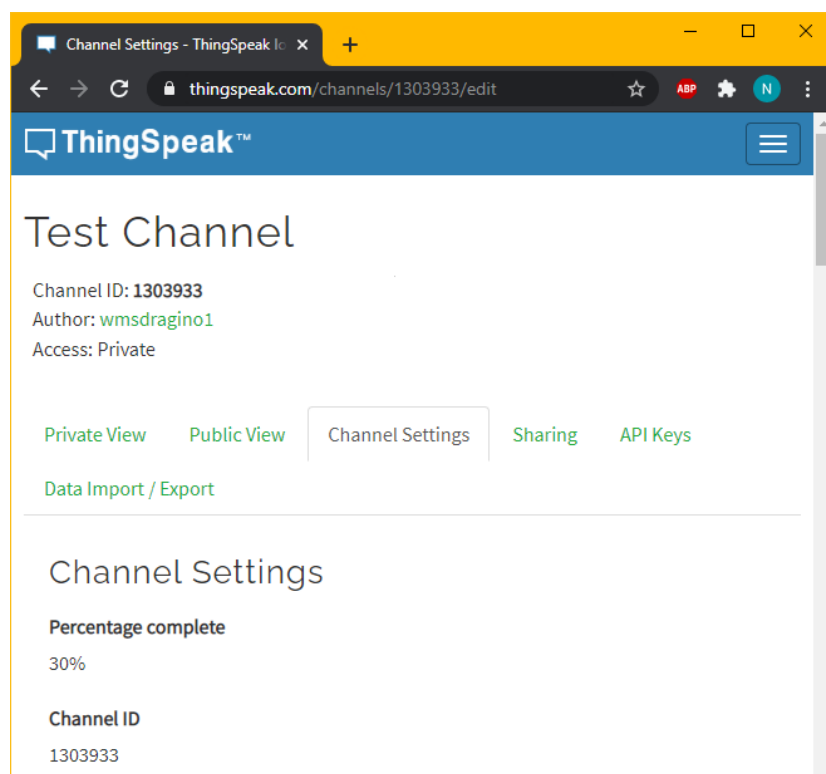
Navigate to the “Channels” and select the “My Channels” option. Select “New Channel” option.

Enter the node number under “Name” and activate the first 3 fields using the following naming convention:



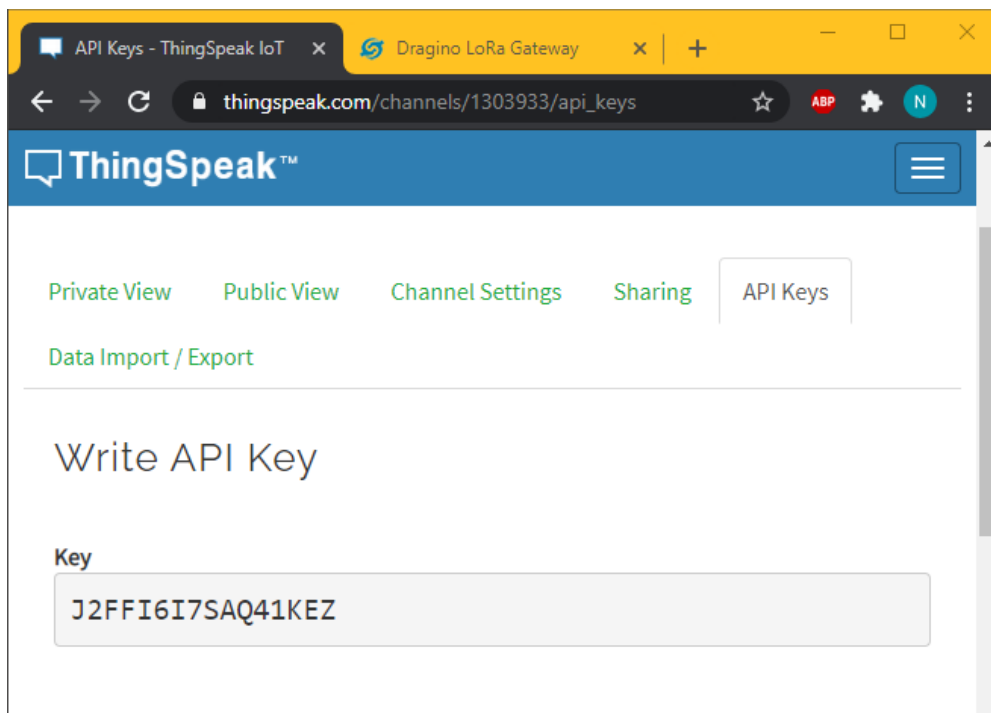
The screenshot shows the 'New Channel' form in the ThingSpeak interface. It contains three input fields, each labeled 'Field 1', 'Field 2', and 'Field 3' respectively. To the right of each field is a blue checkbox that is checked, indicating that the first three fields are activated for data collection.

Navigate to “My Channels” and open your channel. Go to “Channel Settings” and note the “Channel ID” number.



The screenshot shows the 'Channel Settings' page for a channel named 'Test Channel'. The page displays the following information: Channel ID: 1303933, Author: wmsdragino1, and Access: Private. Below this, there are tabs for 'Private View', 'Public View', 'Channel Settings' (which is selected), 'Sharing', and 'API Keys'. Under the 'Channel Settings' tab, there is a section for 'Channel Settings' showing 'Percentage complete' at 30% and 'Channel ID' as 1303933.

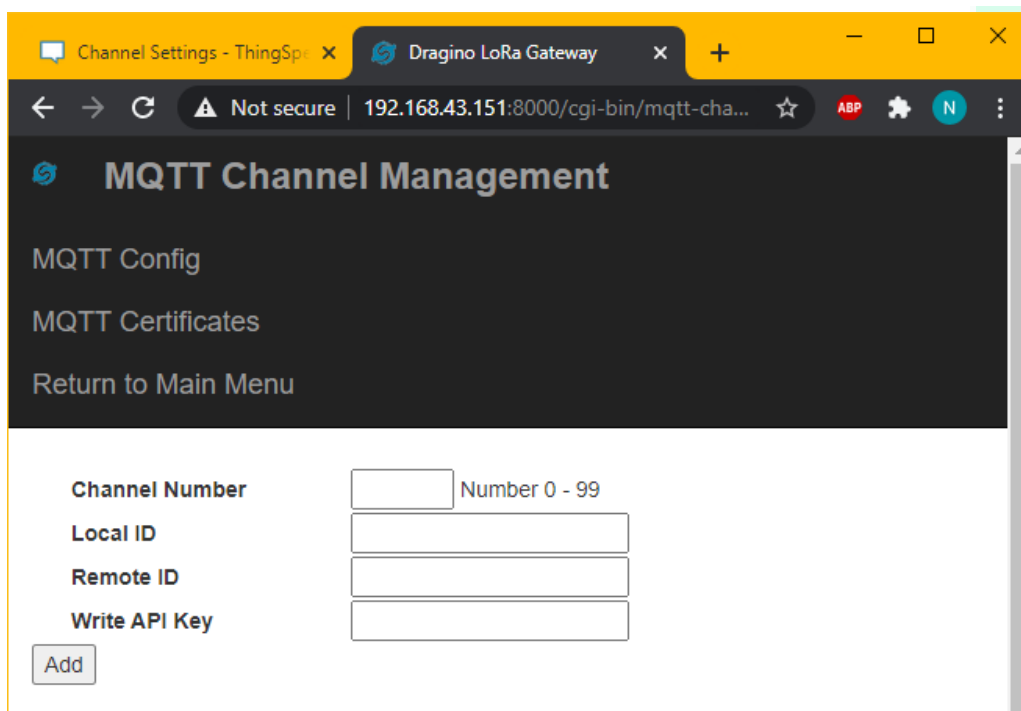
Then navigate to the “**API Keys**” tab of your channel and note the “**Write API Key**”.



### Create MQTT communication link in gateway

Access the LG02 gateway via LAN or Wi-Fi and log in. Go to the “**Services**” tab and select the “**MQTT**” menu. Once you are in the MQTT menu, navigate to the “**MQTT Channels**” tab.

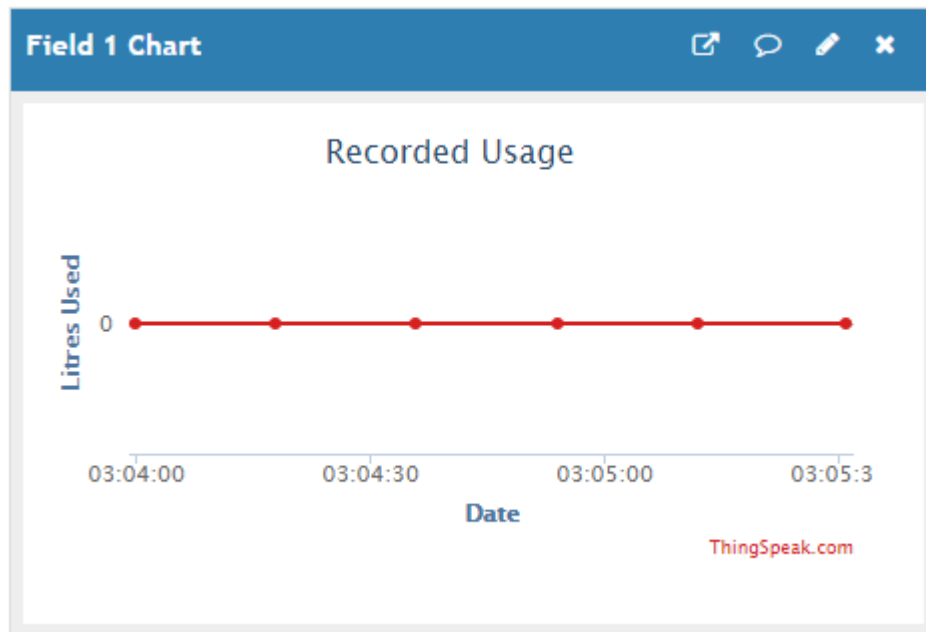
Insert any available number in the “**Channel Number**” field, insert the node number/ name of the node into the “**Local ID**” field, enter the Channel ID into the “**Remote ID**” field and enter the Write API Key of your channel into the “**Write API Key**” field. Click “**Add**”.



## **SYSTEM OPERATION**

### **1. Hydrometer operation**

To simulate the pulse signal produced by the Sensys220C hydrometer, a switch is used. When the switch is pressed and released, a rising-edge signal is sent to the node. The node registers this as a 500ml consumption. The node then logs the node number as well as the total usage recorded to the serial port as well as sends the data to the gateway, this “Update” operation takes place every 12 hours. When the gateway receives the data packet, it forwards the data to TTN and updates the channel to which the node is linked.

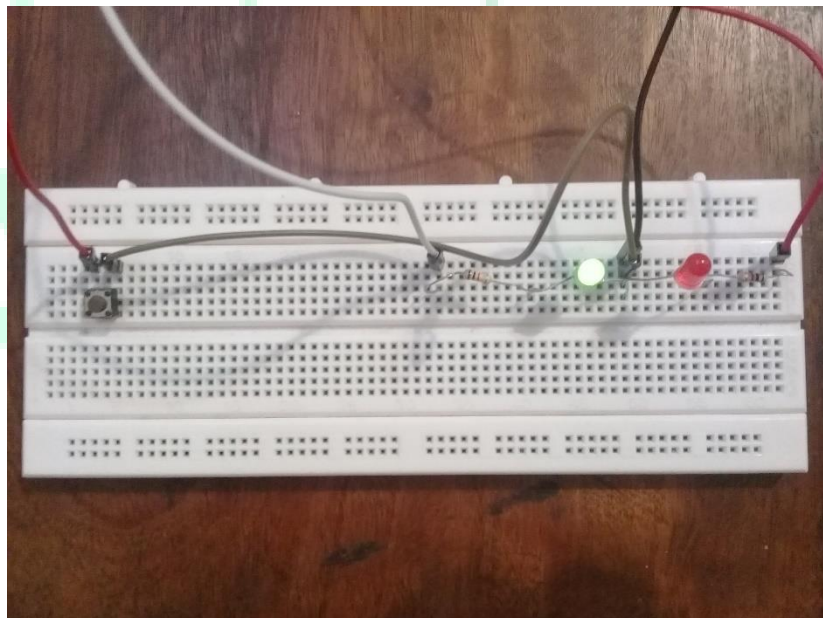
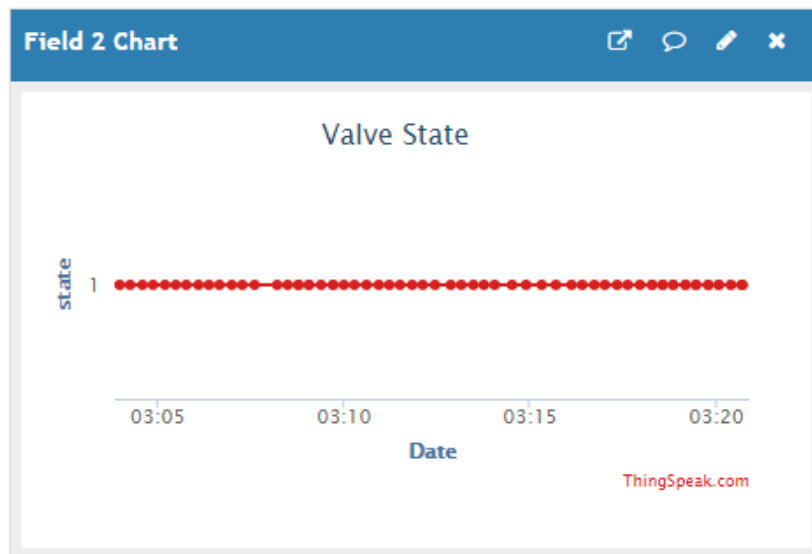


# A-tech

## 2. Valve operation

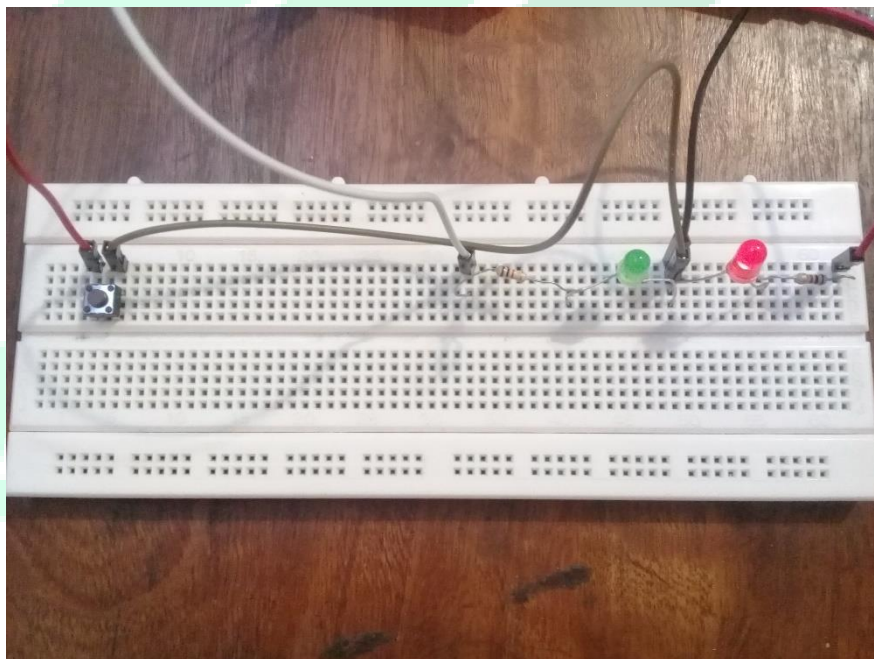
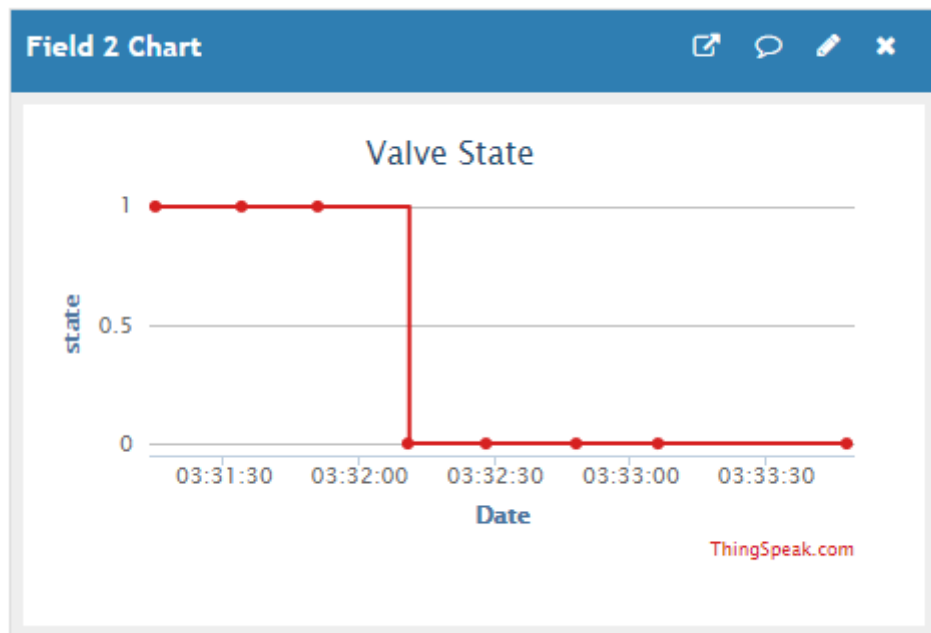
To simulate the state of the valve, two LEDs are used. When the valve is in the open position, the green LED will be lit and when the valve is closed, the red LED will be lit. This is to indicate the polarity of the voltage provided to the valve motor. When the “Update” operation takes place, TTN is updated with the status of the valve. On TTN a recorded value of “1” means that the valve is open and a “0” means the valve is closed.

The figures below show the system in the open valve state.

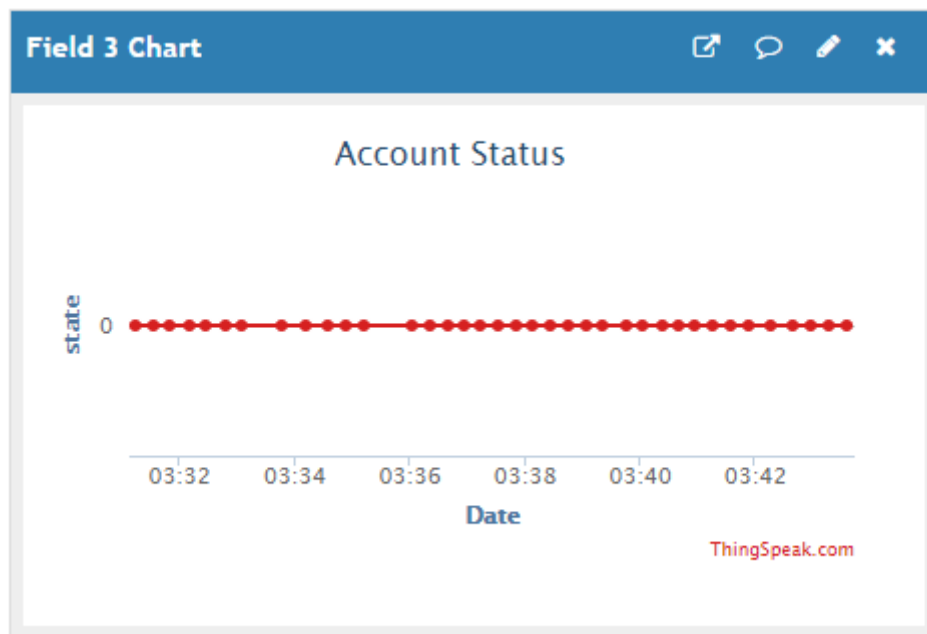




The figures below show the system state after the valve has been put in the close position.



The valve is also set to close automatically at a certain recorded usage depending on the status of the “paid” variable. This indicates that the consumer has used their free basic water amount and have not paid for further water consumption. The status of the “paid” variable linked to the node is also sent to TTN and displayed on the channel when the “Update” function takes place.

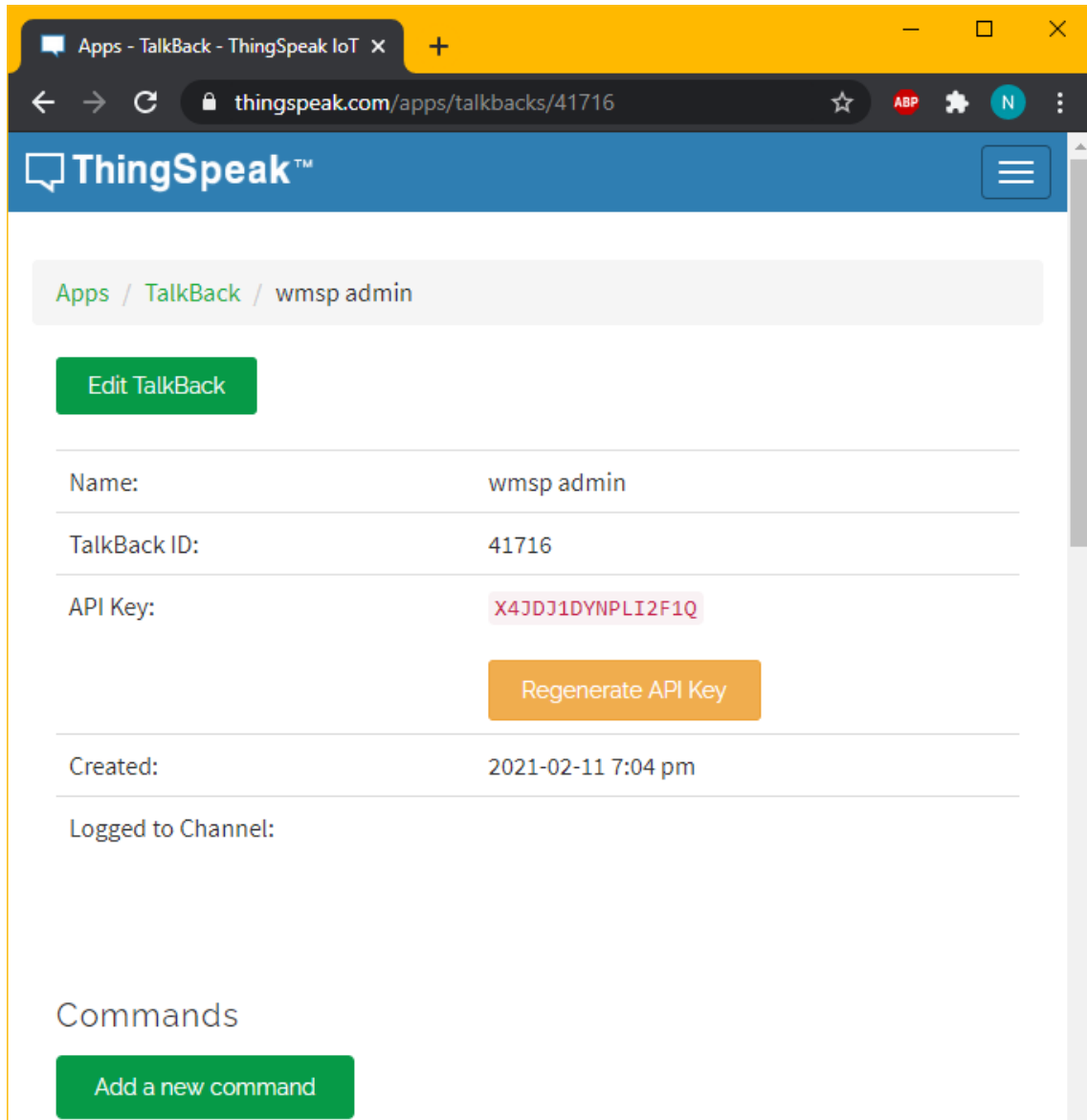


The system is currently set up to provide 6000 litres of water while the “paid” variable is set to false, thereafter if more water is recorded being used, the system will close the valve until the status of the account is change by the administrator using the “**TalkBack API**”

A-tech

### 3. Node control operation

On the thingspeak.com website, if the system admin navigates to the “**APPS**” tab and select the “**TalkBack API**” option, the system admin can broadcast commands to all nodes across the LoRaWan network.



Select the “**Add a new command**” option to open the interface that allows broadcasting to the LoRaWan network.

## Commands

Position	Command string
<input type="text"/>	<input type="text"/>
<input type="button" value="Save"/>	

The format of the command string broadcasted is as follows:

**<node number>\_<command>**

Only the nodes which match the <node number> will carry out the command.

The list of commands are as follows:

**OV: Open Valve**

**CV: Close Valve**

**CAS: Change Account Status (Paid to Unpaid or vice versa)**

**Eg: 4567\_CV**

**This command will close the valve of node number 4567 regardless of its status.**

## **SYSTEM TROUBLESHOOTING**

Please follow the “**MQTT forward instruction**” manual to test the LG02 gateway and Arduino UNO + LoRa Shield components and identify any faults with these components. The software and manuals needed to set up the test environment should be attached along with this document.

A test version of the node software should also be provided to test the operation of the node in the system once the gateway has been set up for MQTT communication. The test version updates every 6 seconds and has a free water usage limit of 5 litres to make testing the system functions easier.