UnifAI Technology

DAVE 2.0 Administrator's Guide

March 2021

Introduction

DAVE (Data, Analytics and Visualisation Engine) is UnifAI Technology's data platform, user interface and dashboard.

DAVE is a platform which enables Users to integrate and manage data from multiple sources, including existing datasets and data from third-party platforms, sensors or devices.

The platform has a powerful embedded analytics capability which enables Users to run real-time analytics across multiple parameters and sensors over time and space.

DAVE helps Users to link their data to Neural Networks created within UnifAI Technology's ANNA platform (Advanced Neural Networks and AI).

The purpose of this guide is to show Administrators how to operate DAVE. It will cover how to:

- Sign-up and sign-in.
- Set-up accounts.
- Configure an account for the required dashboard parameters.
- Set parameter thresholds and create alerts and alarms.
- Create or connect analytics functions and Neural Networks.
- Import, register and configure sensors, individually or in groups.
- Invite new users to the account and set permissions and access controls.

Short cut: return to start of document

Contents

Short cut: return to start of document

Short cut: return to start of document

1. Overview

DAVE has two levels of **Accounts**:

- **Master Account**, in which Account Owners can create or delete new Sub Accounts.
- **Sub Account**, in which Administrators can access all functionality for that Sub Account, but cannot create or delete the Sub Account itself.

DAVE has three types of **Users**:

- **Owners**, are Administrators that 'own' Accounts. Only Owners of Master Accounts can create or delete Sub Accounts.
- **Administrators**, are Users who can view the dashboard and access all visualisation functions, and who can access their Account Setup pages to setup and configure the account details, parameters, events, sensors, and additional users' access and permissions for that Account.
- **Users**, who can only view the dashboard and access all visualisation functions.

For every Account, one Administrator is the **Account Owner**. The Owner is the only person who can authorise the Account to send email notifications. The Owner can pass on ownership of the Account to any other User for that Account.

	Master Account	**Sub Account**
Owner	• Create, administer and delete connected Sub Accounts. • Authorise the Master Account to send email notifications to Users. • All other Administrator functions.	• Authorise the Sub Account to send email notifications to Users. • All other Administrator functions.
Administrator	• Administer the Master Account and any connected Sub Accounts. • Set-up and configure accounts, dashboards, sensors and Users. • Use the Data Import Feature. • Manage User permissions. • Access all DAVE visualisation tools.	• Administer the Sub Account. • Set-up and configure accounts, dashboards, sensors and Users. • Use the Data Import Feature. • Manage User permissions. • Access all DAVE visualisation tools.
User	• Access all DAVE visualisation tools	• Access all DAVE visualisation tools

- If an organisation has a Master Account, the first User in an organisation will set up the organisation's Master Account. That User will be the Master Account Owner and an Administrator.

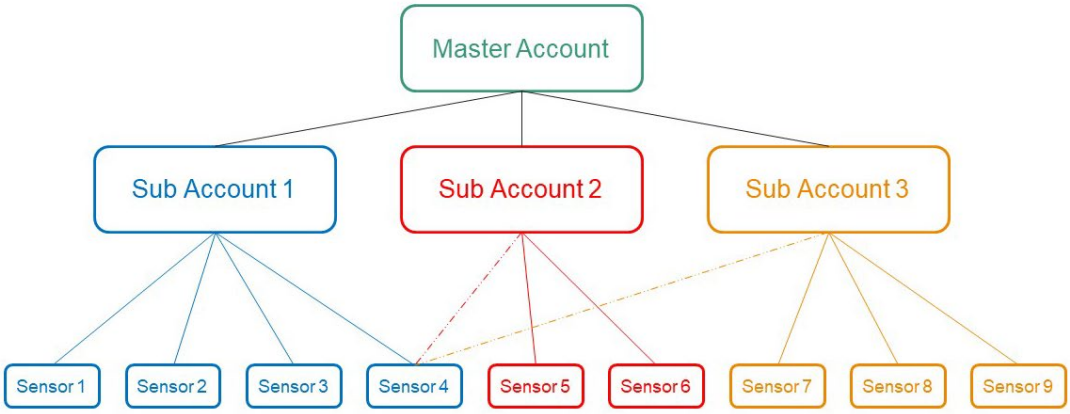
- From the Master Account, the Administrator can set up Sub Accounts and invite new Users.

Short cut: return to start of document

- From Sub Accounts, Administrators can set up Sensors and data inputs, and share access to Sensors and data with other accounts.

- Every Sensor is 'owned' by one Account (a Master Account or a Sub Account), which has control over that Sensor and the data from that Sensor. An Administrator for that Account can share that sensor with one or more additional Accounts. At any time, an Administrator for the 'owning' Account can unshare that sensor with the additional Accounts.

- To illustrate this, in the image below Sensor 4 is owned by Sub Account 1. An Administrator for Sub Account 1 has shared Sensor 4 with Sub Accounts 2 and 3. This means Sub Accounts 2 and 3 can view the data from Sensor 4, but cannot share it onwards and cannot change the Sensor configuration.



For example, the Umbrella Corporation is an environmentally conscious UK based company which deploys air quality sensors at all of their factories and offices. They manage their DAVE Master Account centrally, and have three regional offices covering England, Scotland and Wales. Each regional office manages a Sub Account, each with multiple air quality sensors connected.

In the example above, Sensor 4 could represent a set of new prototype sensors owned and managed by the Scotland region. They have shared this set of sensors with the other two regions so that they can be viewed and assessed for potential wider adoption.

Short cut: return to start of document

2. Registration, Sign-in, Account Set-up: Master Account

- When an organisation has a new Master Account.
- For those that do, the first User in the organisation will be invited by UnifAI to sign up to DAVE, and to set up the organisation's Master Account. That User will be the Master Account Owner and the first Administrator.

You will receive an email with the licence key (which is also your Sign-up Key). You need this to set up your Account.

Click on the link in the email and complete the form, using the same email address that received the Sign-up Key.

When the sign-up process has been completed, you will be sent a further email which will contain a temporary password and a link to the DAVE sign-in page.

Click on the link and sign into DAVE using your chosen Username and the temporary password.

The first time you sign in you will be asked to change your password for security reasons.

You may wish to bookmark this page or save it to the homepage of your mobile device or tablet for future ease of access.

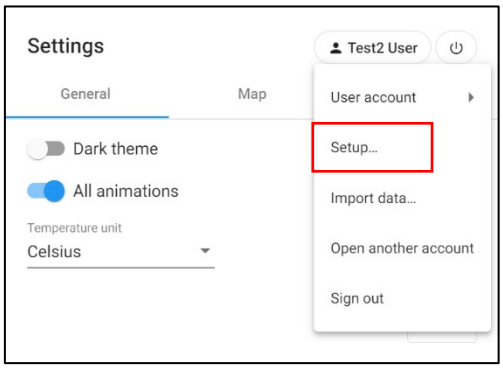
Your new Master Account is now created and ready for set up.

- When you first login to your new Master Account, you will see the Home Screen, which is a map. At this point there will be no additional data or information.
- You now need to set up your Master Account, and create Sub Accounts.
- From the home screen, select **Settings**  in top right-hand corner.

Short cut: return to start of document

- A box will open.

- Select the **Account options** button the top right corner of the box (it will be labelled with your name or User Name) and then select Setup from the drop down menu.



- This will take you into the administration pages for the Master Account, where you can setup and configure the account:

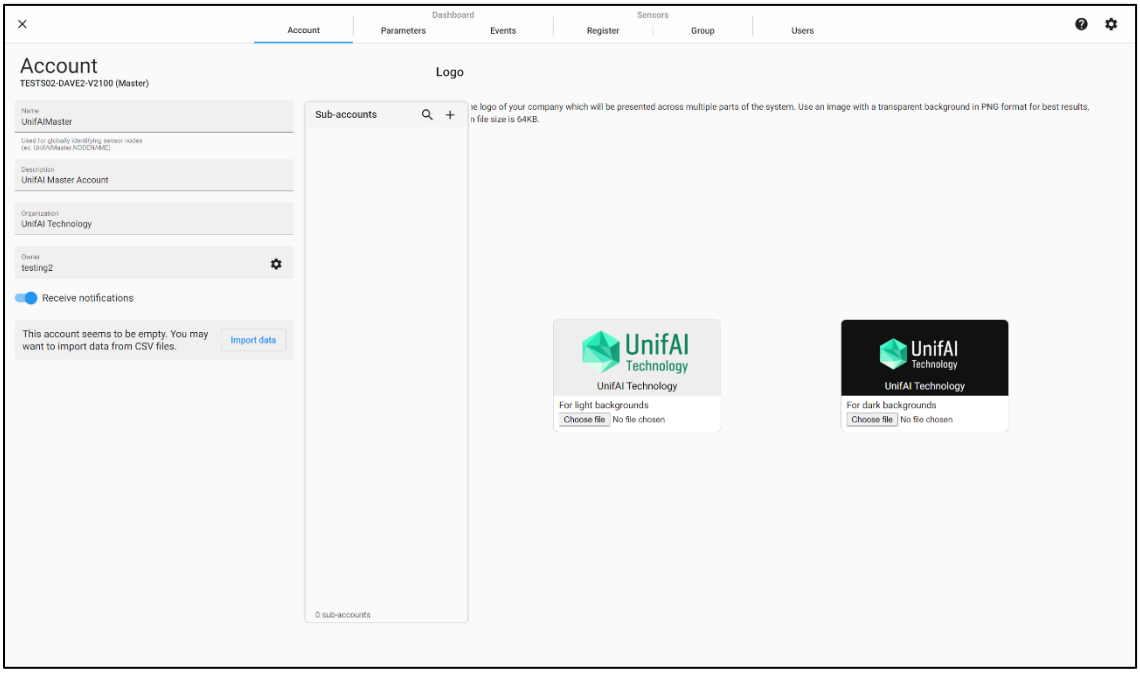
    - Set-up the Master Account details.
    - Import bulk data via CSV files.
    - Configure your Dashboard by defining the parameters which will be visible.
    - Connect, register and configure any Sensors.
    - Invite any additional Users for this Account, and set their permissions.

- As an Administrator, you can change any of these at any time.

- If you are the Owner, you can also create/delete Sub Accounts.

- The setup process will be smooth if you have a clear view on what parameters you want to show, from what sensors and to whom, paying particular attention to your preferred naming convention for parameters and sensors, and permissions for people.

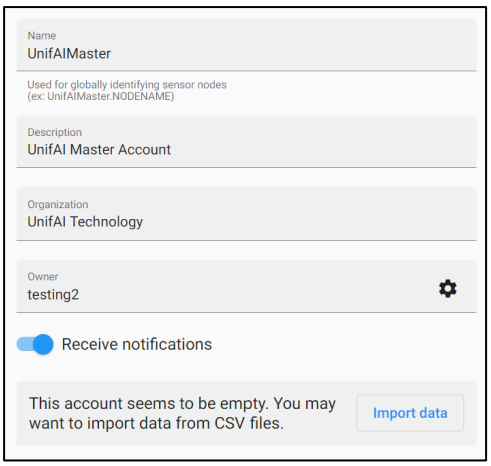
- You start with your account:



Short cut: return to start of document

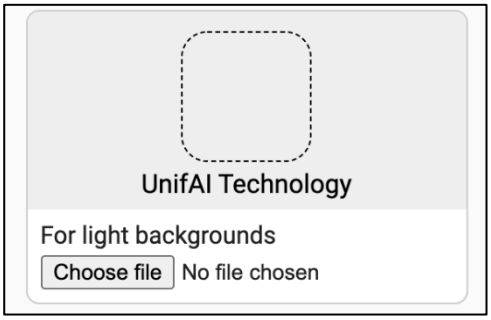
- The account setup section enables Users to personalise the way their account is described, and to upload their organisation logo, which will appear in the top-left corner of the visualisation screen in DAVE.

2.1 Names



- On the left, you can change your **Account Name**, **Description** and **Organisation Name** if required.
- These are the names that will be used to identify the Account. This provides flexibility so that Users may structure their account and sub-account names in line with their business or branding structures.
- If you are the Account Owner, you can change the Account Owner to another User of this account. You can only do this if you have already added additional Users.
- If you are the Account Owner, you will see the **Receive notifications** toggle underneath the names. Only the Account Owner can see this. Toggling this on or off controls whether this Account is authorised to send email notifications to Users when the system detects an Alarm, Warning or Parameter Alert.
- Below this, if the account currently has no data, you will see the **Import data** button which will take you the Data Import Feature. This feature enables the bulk import of existing datasets via CSV files.

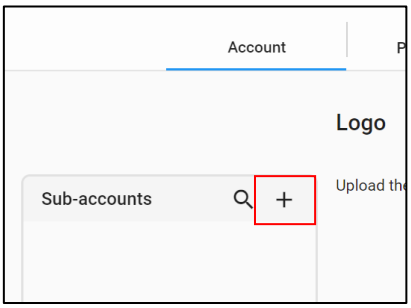
2.2 Logo



- As the Master Account, you can upload your organisation logo by selecting the **Choose file** button and uploading an image file. There is an option to select different images for light and dark backgrounds. Image files must be 64KB or less.
- Sub Accounts will able to select these logos, or upload different ones or alter them.
- The images selected will appear as logos in the top-left corner of the visualisation screen in DAVE.

Short cut: return to start of document

2.3 Creating Sub Accounts



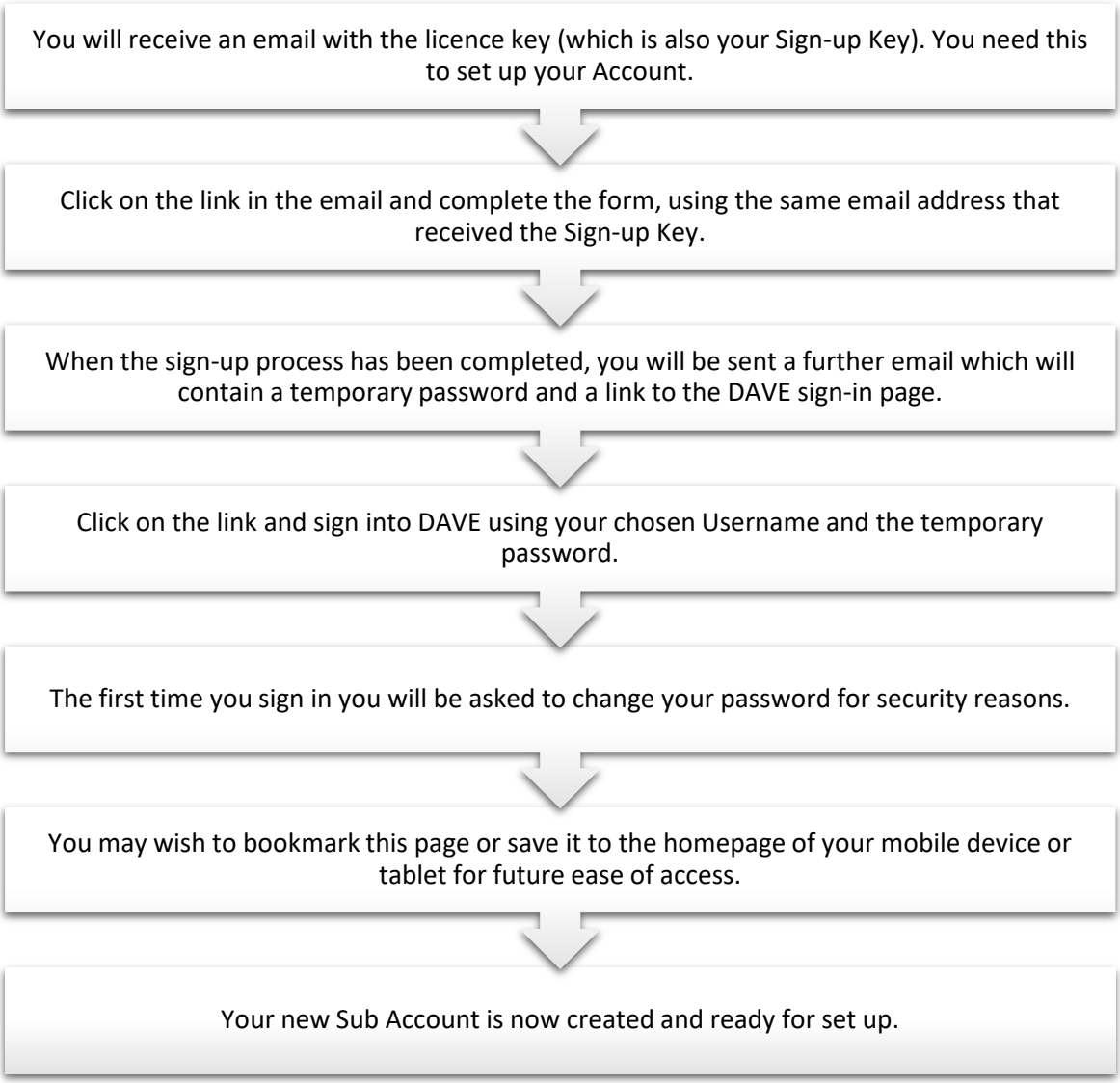
- To the right of the account names, you will see **"Sub-accounts."** Click on the **Plus** sign. +
- DAVE will automatically create and open a new Sub Account to this Master Account. The
- The number of Sub Accounts that can be created and new Users that can be invited may be limited by your licence agreement. If so, additional Sub Account licences and User seats can be purchased if required.

- Each Sub Account is created empty. You can either set it up yourself, or you can invite a new User to it and give them Admin privileges to allow them to set it up. This is explained in the **User Set-up section**.

Short cut: return to start of document

3 Sign-up, Sign-in, Account Set-up: Sub Account

- The Master Account Owner can create Sub Accounts for your company or organisation.
- Master Account Administrators can invite new or existing Users to that account.
- A new User to a Sub Account can be given one of two levels of access:
    - **Administrators**, can set up the Sub Account, including administering sensing parameters, importing new sensors, and inviting additional Users.
    - **Users**, can access visualisation functions only.
- When a Master Account User invites you to join a new Sub Account:



- When you first login to your new Sub Account, you will see the Home Screen, which is a map. At this point there will be no additional data or information.

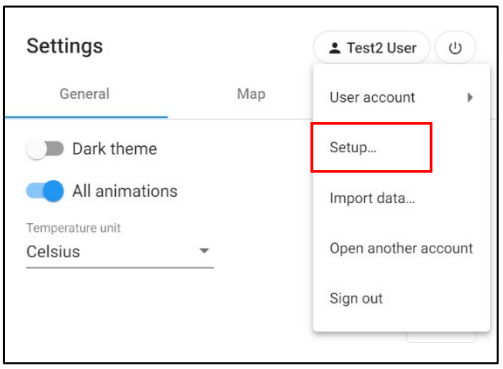
Short cut: return to start of document

- You now need to set up your Sub Account.

- From the home screen, select **Settings**  in top right-hand corner.

- A box will open.

- Select the **Account options** button the top right corner of the box (it will be labelled with your name or User Name) and then select Setup from the drop down menu.



- This will take you into the administration pages for the Sub Account, where you can setup and configure the account:

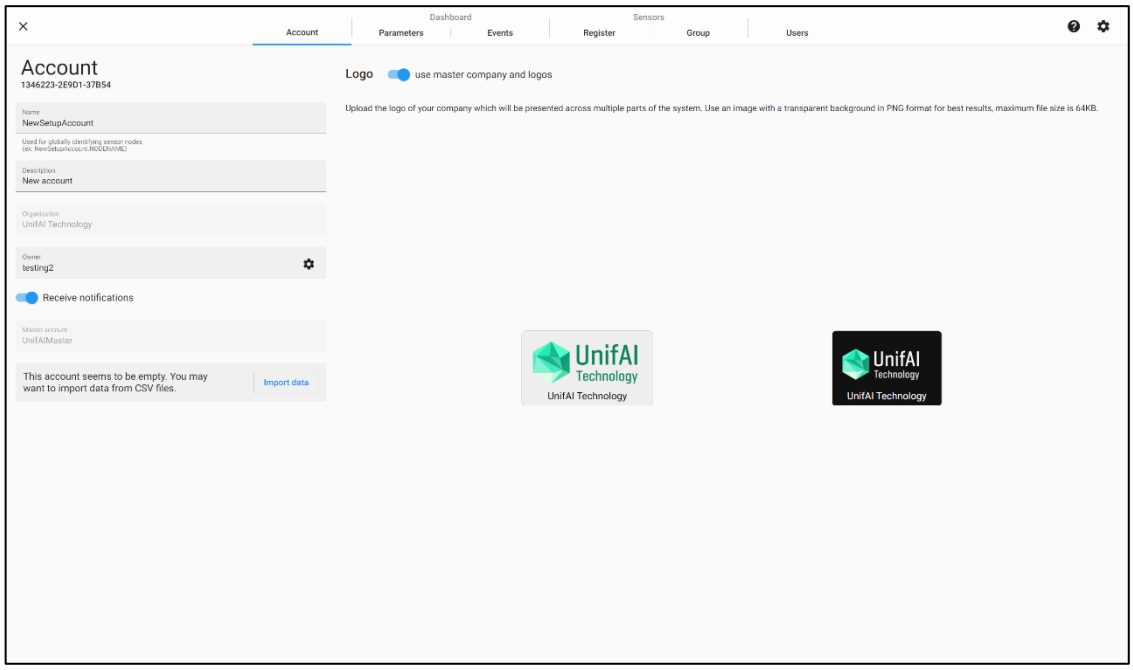
    - Set-up the Sub Account details.
    - Import bulk data via CSV files.
    - Configure your Dashboard by defining the parameters which will be visible.
    - Connect, register and configure any Sensors.
    - Invite any additional Users for this Account, and set their permissions.

- As an Administrator, you can change any of these for this Sub Account at any time.

- The setup process will be smooth if you have a clear view on what parameters you want to show, from what sensors and to whom, paying particular attention to your preferred naming convention for parameters and sensors, and permissions for people.

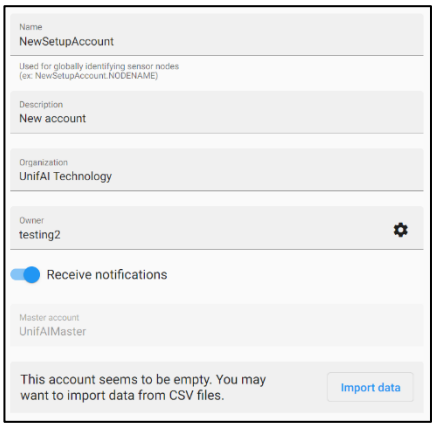
Short cut: return to start of document

- You start with your Sub Account:



- The account setup section enables Users to personalise the way their account is described, and to upload their organisation logo, which will appear in the top-left corner of the visualisation screen in DAVE.

3.1 Names



- On the left, you can change your **Sub Account Name**, **Description** and **Organisation Name** if required.

- These are the names that will be used to identify the Account. This provides flexibility so that Users may structure their account and sub-account names in line with their business or branding structures.

- If you are the Account Owner, you can change the Account Owner to another User of this account. You can only do this if you have already added additional Users.

- If you are the Account Owner, you will see the **Receive notifications** toggle underneath the names. Only the Account Owner can see this. Toggling this on or off controls whether this Account is authorised to send email notifications to Users when the system detects an Alarm, Warning or Parameter Alert.

- Below this, if the account currently has no data, you will see the **Import data** button which will take you the Data Import Feature. This feature enables the bulk import of existing datasets via CSV files.

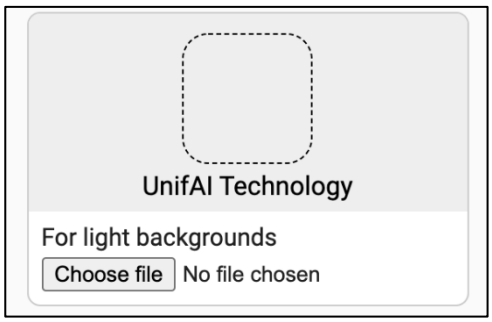
Short cut: return to start of document

3.2 Logo

- When setting up a sub-account, you can choose to use the Master Account logos from your organisation selecting the **Logo** button at the top of the screen. If you toggle the Logo button on, you will be unable to change the Organisation Name on the left, which will default to the Organisation Name for the Master Account.

- Alternatively, you can add your own image by selecting **Choose file** button and uploading an image file. There is an option to select different images for light and dark backgrounds. Image files must be 64KB or less.

- The images selected will appear as logos in the top-left corner of the visualisation screen in DAVE.



Short cut: return to start of document

4 Dashboard Setup

The dashboard setup section enables Users to create the Parameters and Events that, when enabled, can be shown in the DAVE dashboard.

4.1 Adding Parameters

There are two types of Parameters:

- Direct Parameters, which show data received directly from a sensor.
- Indirect Parameters, which are created in DAVE from Direct Parameters, using neural networks or mathematical functions to derive additional information.

4.1.1 Direct Parameters

- To set up Direct Parameters, the User must know what sensors they will be setting up in the account, what parameters those sensors are measuring, and how those parameters are 'named' in the API. If you don't know this information, you should speak with your Administrator.
- For the purposes of this guide, we have prepared a Training Sensor called Test01 which you can practice with. This is a simulated water quality sensor, but the process is applicable to any sensor. This practice is the **Tutorial**.

Table 1 below shows a list of Direct Parameters that you can practice setting up.

Parameter (mandatory)	**Unit (recommended)**	**Decimal places (recommended)**	**Range (recommended)**
EC	µS/cm	0	0 to 5000
ORP	mV	0	-200 to 1000
pH	Blank	2	0 to 14
Temp	C	1	0 to 60
Battery	Blank	0	0 to 3

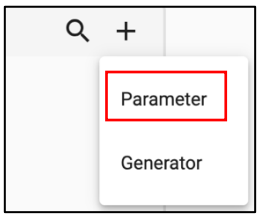
Table 1

- In the menu at the top of the page, click on the **Parameters** tab.

- In the box on the left of the page, click on the **Plus**  sign.
- A small drop down box will open.

Short cut: return to start of document

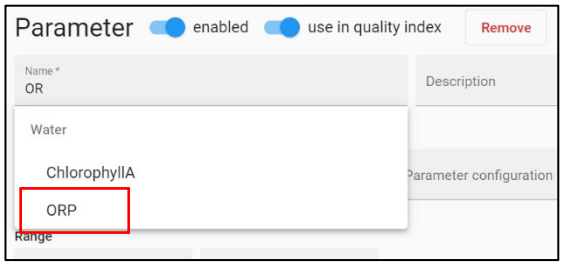
- Select Parameter from the menu.



- This will take you to the Parameter set up page.

- In **Name*** start typing the name of one of the parameters from Table 1, such as ORP.

- If the Parameter is already known to the system, it will appear in a drop down list. Select the correct name from the list.



- If the Parameter is new to the system, you should type the name of the Parameter.

- Note: data on Direct Parameters from sensors is passed to DAVE via API. The data will be labelled, and the Name of the Parameter in DAVE should match the Name used in the API for the Parameter to work. If you don't know this information, you will need to speak with whoever in your organisation managed the APIs and set up the sensors. (*For example, if the API and the Parameter setup in DAVE both call a parameter 'Salinity', then the data will be correctly attributed. If the data from the API is labelled 'Salinity' and in DAVE the user names the parameter 'Salt', DAVE won't match the two and this parameter will not be visible in the dashboard*).

- When you select the name of an existing parameter from the drop down list, the **Description** field will automatically self-populate. This Description can be over-written if appropriate.

- For a new Parameter name, the Description field will remain empty and can be completed manually.

- The Description is what the parameter will be called in the Dashboard gauges and graphs.

- Assigning **Tags** is optional and can help with parameter management. Users can search the parameter list by tag. This could be helpful if there are multiple parameter types, for example: Air, Water, Energy etc.

- Above Name, Description and Tags are two buttons labelled **enabled** and **use in quality index**.

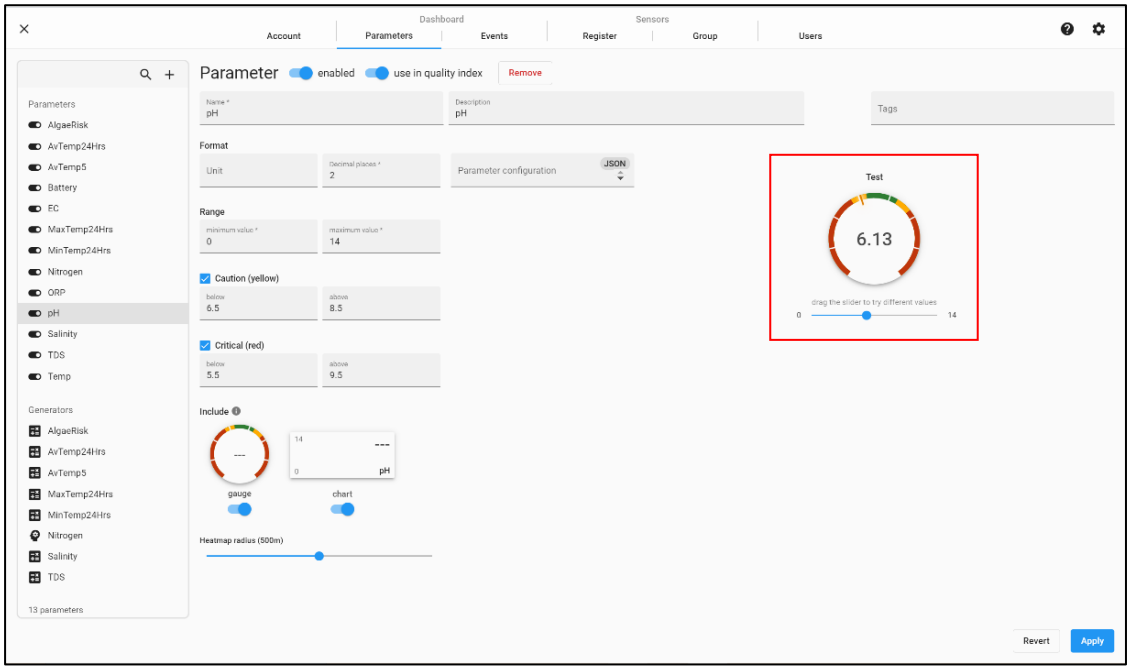
- Toggling **enabled** on and off turns that parameter on and off in the dashboard without deleting it. If a parameter is not enabled, it won't be visible in the dashboard but will continue to be calculated in the background.

- Toggling **use in quality index** on allows a User to identify a parameter that is used to create a Quality Index. A Quality Index is a powerful means of summarizing complex data and

Short cut: return to start of document

facilitating its communication. Only those parameters used to calculate an Index should be selected. To be used in a Quality Index, the parameter will also require “red limits” to be defined**.**

- Update the **Unit**, in the case of ORP this will be mV (millivolts).
- Choose how many decimal places to view. This will depend on the likely range and amplitude of the parameter.
- Set the **Range** selecting the minimum and maximum values. The appropriate numbers to choose will depend on a number of factors. For example:
    - pH can only be between 0 and 14. A number outside this range would be meaningless.
    - EC for a water quality sensor may depend on the output range of the sensor/probe specified by the supplier, which may differ for fresh water and salt water.
- Optional: if required you can customise the dashboard gauges with a ‘traffic light’ visual by setting thresholds for values: **Critical** (red), **Caution** (yellow) or green. Do so by selecting the levels ‘above which’ and/or ‘below which’ each colour applies. The easiest way to learn how to use this setting is to have a play. You can see how your selection will look for different outputs by dragging the slider left and right in the Test gauge on the right-hand side.

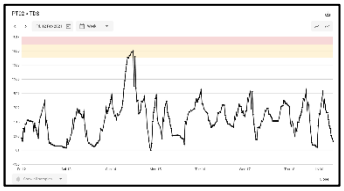


- When a parameter value crosses the pre-set **Critical** threshold, a visual Parameter Alert in the form of a red circle with an exclamation mark shows on the Dashboard in two places:

- On the relevant Gauge.

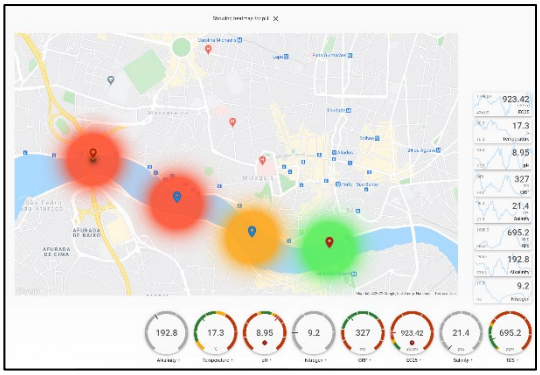
- On the Sensor List on the left of the Home Page.

- Parameter Alerts on the home page are live alerts, only showing while the parameter is in the red and disappearing when the parameter returns to yellow or green. Viewing the historic chart for a parameter will show when a parameter has previously been in the red or yellow.



- If **Receive Notifications** is on, Users will receive an alert by email when a Parameter Alarm is triggered, and when it ends.

- Optional: if you wish to make use of the dashboard **Heat Map** functionality you can use the slider at the bottom to set the visual radius for each sensor. This will enable the heat map view in the Dashboard, as shown on the right.



- The appropriate radius will depend on the use case, how many sensors are deployed, and how far they are from each other.

- Repeat as above for all Parameters in Table 1.

- Note that when you add a new Parameter for an existing sensor, it will remain empty in the dashboard until the sensor next transmits data.

4.1.1.1 Customising Gauges with Text

- Ordinarily a gauge will show a numerical value but there may be situations where a text label is more appropriate.

- To illustrate this capability in the **Tutorial**, we will use the battery. For the Training Sensor, the battery level is received by DAVE through the API as the number 0,1,2 or 3. A full battery would read “3” and a low battery would read “1.”

Short cut: return to start of document

- Go to the Battery parameter and add the following to the caution (yellow) and critical (red) range boxes:

    - Caution below: 2
    - Caution above: *(leave blank)*
    - Critical below: 1
    - Critical above: *(leave blank)*

- The gauge will now show as green, yellow or red when the battery level is in the top third, middle third or bottom third of capacity.

- To configure the gauge to show text instead of the number, we will show you how to use a simple function.

- Select the JSON button in the Parameter Configuration field near the centre of the screen. A drop-down menu will appear.

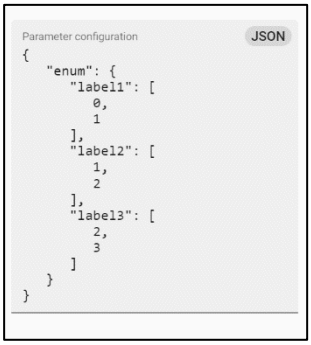
- This menu enables you to add a number of pre-prepared example functions to customise the view of the gauge, or you can write your own.

- You can easily adapt the example functions with the values and text of your choice.

    - **Low/High text** is a simple function that will replace the number in a gauge with the words “Low” or “High” if the parameter value is in the bottom or top 25% of the parameter range. Otherwise, the gauge will show the number value. You can simply apply this, and if necessary change the labels to whatever text you want.

    - **Label preset** has a sub-menu of more complex options with more layers, with a choice of up to eight layers of detail.

- Select **3 labels**, and a pre-prepared function will appear. This gives you the correct structure, and all you need to do is change the values and the text to those you require.



- In this case the values are correct, but replace the text with the following:

    - Replace the word “label1” with “very low”
    - Replace the word “label2” with “low”
    - Replace the word “label3” with “normal”
    - ***Do not change anything else (such as punctuation, brackets or spaces).***

Short cut: return to start of document

- The function should now look like this:

```
{
  "enum": {
    "very low": [
      0,
      1
    ],
    "low": [
      1,
      2
    ],
    "normal": [
      2,
      3
    ]
  }
}
```

- If you are uncertain about what to edit in the function, you can copy and paste the above into the box instead.

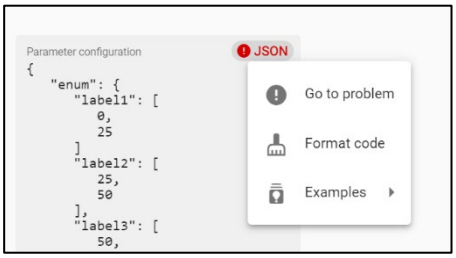
- Click **Apply**. Then you can test the function by going back into the Battery parameter and moving the slider in the test gauge.

- This function tells DAVE that if the parameter value is between 0 and 1, you want the gauge to say ‘very low’, between 1 and 2 ‘low’ and 2 and 3 ‘normal’.

- The dial will still move around the gauge and point to green, yellow or red in the correct place, but the text message will be displayed instead of the number.

- This principle can be replicated with all parameters and can operate with up to 8 layers.

- If the User makes an error when altering the function, DAVE will turn the JSON button red and provide a button saying “**Go to problem**,” which will move the cursor to the part of the function where the problem is. If a User accidentally makes a small alteration the function during editing there is also a “**Format code**,” which can automatically correct minor errors.



4.1.2 Indirect Parameters

- Indirect Parameters are derived from one or more Direct Parameters, using mathematical formulas or neural networks.

Short cut: return to start of document

- This is done by creating a **Generator**, which generates the Indirect Parameter from one or more Direct Parameters.

- Generators for Indirect Parameters can be created in one of two ways:

    - Using a mathematical **Formula** (this could be a simple formula written by the Administrator, such as determining a rolling average over a number of readings, or potentially complex pre-determined library formula made available by UnifAI).

    - Using a trained Artificial **Neural Network** (for example, in the case of water quality sensing, UnifAI's AI has been trained with millions of lines of historic water quality data and can use the Direct Parameters from Table 1 to derive multiple additional parameters such as Nitrates, Phosphates, Dissolved Oxygen, etc, all of which would otherwise require expensive specialist sensors or laboratory tests).

- For the **Tutorial**, we will continue to use the Training Sensor to demonstrate the process for a simulated water quality sensor, and we will build on the Direct Parameters you have already set up.

Table 2 below shows a list of Indirect Parameters that you can practice setting up.

Parameter (mandatory)	Type	Unit (recommended)	Decimal places (recommended)	Range (recommended)
Salinity	Formula	PSU	1	0 to 40
TDS	Formula	mg/l	0	0 to 1500
Nitrogen	Neural Net	mg/l	2	0 to 5

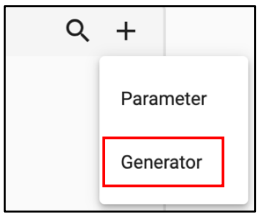
Table 2

4.1.2.1 Using a pre-determined 'library' formula

- In the box on the left of the page, click on the **Plus**  sign.

- A small drop down box will open.

- Select **Generator** from the menu.



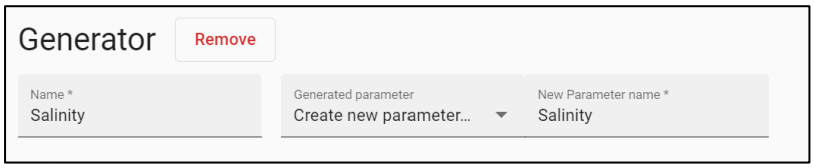
- This will take you to the Generator set up page. Generators are used to create Indirect Parameters.

- In **Name*** type the name of the Indirect Parameter you are setting up from Table 2. For the **Tutorial**, start with Salinity.

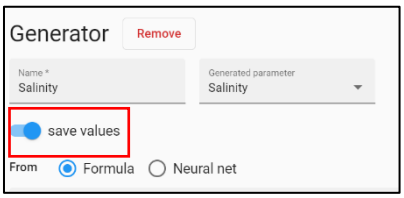
- When you type the Generator Name, the **New Parameter Name** field on the right will automatically self-populate. This is the visible name of the new parameter you are creating which will appear on the gauges and graphs in the dashboard, and you can overwrite this with a different choice if required.

Short cut: return to start of document

- The middle field, called **Generated Parameter**, is used when you want to connect a Generator to an existing Parameter that has already been set up. For now, leave this on "Create new parameter."



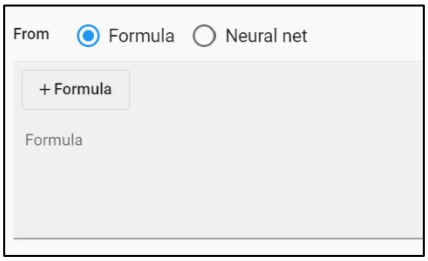
- Below **Name** is a button labelled save values.



- A Generator creates an output based on one or more inputs. The **save values** function gives the User the choice of generating the output whenever the sensor sends a reading and to save those outputs (save values on), or only generating an output from this Generator 'on the fly' when needed (save values off). If the save values toggle is off, the output from this Generator will not be recorded in the database, meaning you will not have a record of historic values from this Generator. This can be useful when a generator is created for common use in other parameters but does not need to be saved in the database. This allows commonly used functions to be created once and deployed many times.

- If you are uncertain, leave **save values** toggled on.

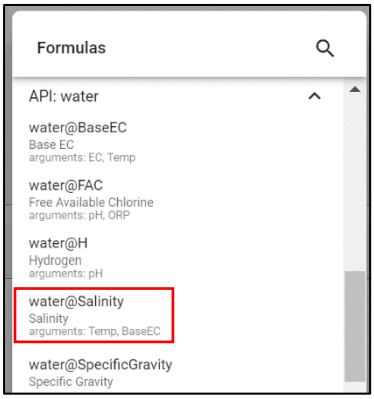
- The Salinity Generator uses a formula and not a neural network, so ensure **Formula** is selected, and then click **+Formula**.



- A drop-down menu will appear with a list of available 'library' formulas. For the **Tutorial** these are all water related, but these could just as easily relate to air quality, energy use or more.

- Note that under each formula is a short list of "arguments." **These are parameters that must be set up and enabled in order for this formula to work**.

- For example, for water@Salinity to work, both Temp and BaseEC must also be set up and configured. If you are following this tutorial, you have already set up Temp, but not BaseEC.



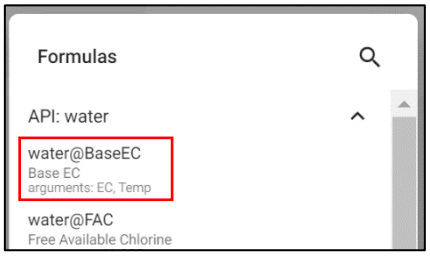
Short cut: return to start of document

- Select **water@Salinity** and click **Apply**.

- Look at the list on the left of the screen. You will see that the Generator named “Salinity” has appeared in the Generator list, and the new Parameter named “Salinity” has appeared in the Parameter list.

- Select **Salinity** in the **Parameter** list, and you can now configure the parameter units, range, gauges and heat map in the same way as you did for the Direct Parameters.

- Now set up BaseEC in the same way, using **water@BaseEC**. This Generator requires Temp and EC “arguments”, both of which you have already set up.



- Now set up TDS in the same way, using **water@TDS**. This Generator only requires the “argument” EC, which you have already set up.

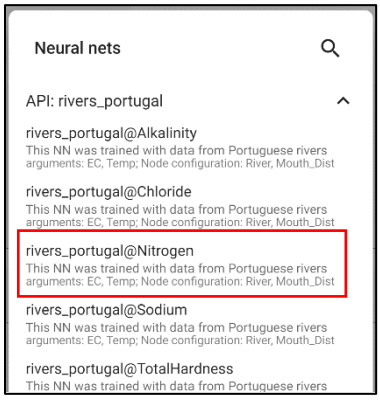
- If you don’t wish to see one of the underlying “arguments” in the dashboard, such as BaseEC, you can hide it from the Settings box in the Dashboard – see User Guide for instructions on how to do this.

- Note that when you add a new Indirect Parameter for an existing sensor, it will remain empty in the dashboard until the sensor next transmits data. In some cases, it may require more than one reading before data will show in the dashboard (some Indirect Parameters require several readings over a period of time).

- Also note that when you add or create new Generators for an existing sensor, you must go to the Sensor Setup page and ensure that the new Generators are selected for each sensor or sensor group, otherwise you will not see them in the dashboard.

4.1.2.2 Using a Neural Network

- Setting up an Indirect Parameter using a Neural Network is exactly the same, other than ensuring that **Neural Net** is selected instead of **Formula**, and clicking **+Neural net**.



- As with Formula Generators above, under each Neural Network is a short list of “arguments.” **These are parameters that must be set up and enabled in order for this Neural Network to work**.

- For Neural Networks to work correctly, there is an additional variable that must be correctly configured – the Node Configuration on the **Sensors Register** page. The “arguments” shows the variables that need to be correctly configured in the Node Configuration field on the Sensors Register page, in this case *River* and *Mouth_Dist*. For some Neural Networks, this will be *Zone*. This tells DAVE which

Short cut: return to start of document

Trained Neural Network version to use. Node Configuration is explained in the Sensor Setup section.

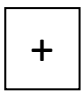
- In this example, name the Generator "Nitrogen" and select the Neural Net called **rivers_portugal@Nitrogen**.
- Then click **Apply**.
- You have successfully added four Generators to your account, and used them to set up three Indirect Parameters. One of these new parameters (Nitrogen) is being derived in real-time by a trained Artificial Neural Network.

4.2 Creating New Formulas

- The previous section demonstrated how to set up an Indirect Parameter with a Generator that used a pre-determined 'library' formula already written and embedded in DAVE.
- Users can also create new Generators, and new Indirect Parameters, using their own mathematical functions and formulas. These can use single or multiple existing parameters as inputs. There is a comprehensive array of mathematical functions available, with a number of examples shown in Appendix 2.
- These can be used in any combination and with any other parameters available to create new Indirect Parameters using simple or highly complex formulas.
- Simple examples might include:
    - A rolling average over a period of time (such as average temperature over the last 6 hours).
    - The highest or lowest value over a period of time (such as the highest pH over the last 24 hours).

4.2.1 User-created formulas

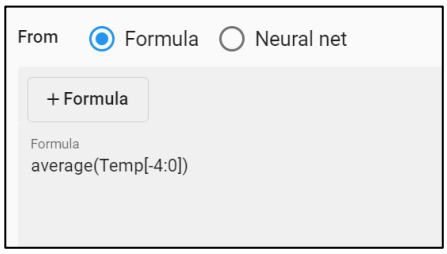
- In this **Tutorial** exercise you will create a new Generator, and generate an Indirect Parameter to show an average of temperature over the last 5 readings.
- In the Dashboard Parameters setup page, click on the **Plus** sign on the left and select **Generator** from the drop-down menu as before.



- You can name the new Generator whatever you like. For the **Tutorial** name it AvgTemp5, and ensure **Formula** is selected and **save values** is on.
- Instead of selecting the **+Formula** button as before, you will manually add a formula in the box below the button.

Short cut: return to start of document

- Click in the Formula box to add the formula:



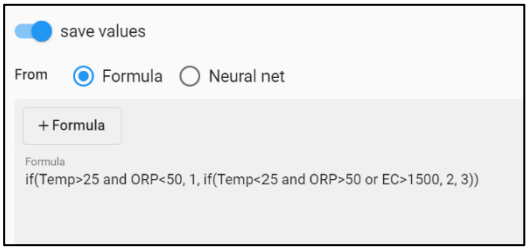
    - **average(Temp[-4:0])**
    - *(DAVE treats the current reading as 0, the last reading a -1, the one prior to that as -2 etc).*

- Once complete, press **Apply**.
- As before, the Generator and Parameter lists on the left will automatically populate with the new Generator and Parameter, both called AvgTemp5. In this example, the output will not be visible in the dashboard until 5 new readings have been received from a sensor.
- This functionality can be very powerful.
- The outputs from a new parameter such as this can be used as inputs for other formulas, creating highly complex layered functions.
- Note that when you add a new Indirect Parameter to an existing sensor with a formula looking at change over time, like this one which is averaging the past five readings, it will remain empty in the dashboard until the sensor has transmitted enough readings (in this case, five).

4.2.2 More complex example

- Some complex outputs can depend on the interplay between multiple parameters, and don't necessarily produce a numerical value.
- For example, in a water use case an Indirect Parameter can be created to monitor the risk of an Algae Event based on three existing parameters.
- In the Dashboard Parameters setup page, click on the **Plus** sign on the left and select **Generator** from the drop-down menu as before.

- Name the new Generator AlgaeRisk, and ensure **Formula** is selected and **save values** is on.
- Click in the Formula box below the **+Formula** button and add the formula:
    - **if(Temp>25 and ORP<50, 1, if(Temp<25 and ORP>50 or EC>1500, 2, 3))**



- Click **Apply**, and then go into the new Parameter called AlgaeRisk.
- Set the **Range** with a minimum value of 0 and a maximum value of 3.
- Set **Caution** (yellow) below 2 and above 1.

Short cut: return to start of document

- Set **Critical** (red) below 3 and above 2.
- In **Parameter configuration**, select the JSON button followed by:
    - Examples>Label Preset>3 labels
- Edit the pre-set JSON function:
    - Replace the word “label1” with “Risk”
    - Replace the word “label2” with “High Risk”
    - Replace the word “label3” with “Event”
    - ***Do not change anything else (such as punctuation, brackets or spaces).***

- Test the results by moving the slider on the Test Gauge.
- Press the **Apply** button.

4.3 Adding Events

- In addition to monitoring and displaying values for individual parameters, DAVE has the tools to monitor complex patterns of behaviour and change between those parameters – looking for multi-dimensional correlations over time and across parameters.
- When certain patterns are identified, DAVE registers an **Event**.
- An event triggers an **Alarm** (with a red alarm clock icon) or a **Warning** (with an exclamation mark in a black triangle icon), which is shown on the Dashboard and can trigger an email notification to the User.

Short cut: return to start of document

- Events are shown visually on the dashboard, and recent Events are listed chronologically. Viewing the historic chart for any parameter or set of parameters will show when any past readings coincided with one or more Events.

- Alarms and warnings are essentially the same but exist to provide users with the flexibility to distinguish between events for higher and lower levels of prioritisation.

- If **Receive Notifications** is on, Users will receive an alert by email when an Event occurs.

- There are two types of Events:

    - **Library Events**, which are created by DAVE using a combination of mathematical functions and artificial intelligence, and available from the Events 'library.'

    - **User Generated Events**, which are created by the User.

4.3.1 Library Events

- In the menu at the top of the page, click on the **Events** tab.

- In the box on the left of the page, click on the **Plus**  sign.

- This will take you to the Events set up page.

- Select **Alarm** or **Warning**.

- If required, choose a "**Time off**" period. This is the cooling off period after an Alarm or Warning is triggered during which it will not be repeated.

- Select **+System Event**

- From the drop-down menu, select **water@HeterotrophicPresence**

- The **Name** and **Description** fields will automatically be populated.

- Select **Apply**.

- You have set up an Event in DAVE, and in a water quality use case you would receive an alert if the system detects a pattern of parameters and variables consistent with a heterotrophic bacteria event.

Short cut: return to start of document

- Note that when you add or create new Events for an existing sensor, you must go to the Sensor Setup page and ensure that the new Events are selected for each sensor or sensor group, otherwise you will not receive Alarms or Warnings.

4.3.2 User Generated Events

- Users have the flexibility to create their own **User Generated Events** using their own mathematical functions and formulas. These can use single or multiple existing parameters as inputs. There is a comprehensive array of mathematical functions available, with a number of examples shown in Appendix 2.

- These can be used in any combination and with any available parameters to create new Events using simple or highly complex formulas.

- In the box on the left of the page, click on the **Plus**  sign.

- In **Name*** type “AB,” and in **Description** type “Algae Bloom.”

- Select **Alarm**, and in the **Time off** add “3” and select **hours**.

- In the **Conditions** box below the **+System Event** button, copy and paste the following:

```
{
  "conditions": {
    "minGrowth": [
      "Temp >= 20 and Temp <
45 and pH >= 5 and pH <= 8
and ORP < 200",
      "Minimum growth
conditions for Algae Bloom
event"
    ],
    "optimGrowth": [
      "minGrowth and Temp
>= 35",
      "Optimum growth
conditions for Algae Bloom
event"
    ],
    "AGEvent": [
      "minGrowth and (EC -
EC[hh(-24)].prev >= 300)",
      "Algae Bloom Event"
    ]
  }
}
```


Short cut: return to start of document

- Select **Apply**.

- This Event creates a series of triggers that show the stages of probability for an Algae Bloom event in water, where the first condition creates a scenario based on temperature, pH and ORP and is then used in a higher level second formula with an added test based on temperature, and then a third test which is based on changes in conductivity over time that is an indicator of biological activity.

4.4 Argument Mapping

- Argument Mapping tells the system to map a parameter to an input, or a parameter to an output.

- For example:

    - A User may wish to use a **Generator** which has a formula that uses ORP as an input, but the sensors connected to the account uses the label “Redox” for ORP.

    - Argument Mapping enables the Administrator to tell DAVE that Redox is equal to ORP.

- To do this a User enters the parameter within the formula and then the parameter that will be used, as follows:

```
{
  "ORP": "Redox"
}
```

- The User may now use the systems formula without having to re-create it.

- There may be occasions where a specific value needs to be attribute to a parameter or a formula.

- They can be added using one of the following examples:

```
{
  “Redox”: “10” or
  “Redox”: “sys@formula”
}
```

- These functions enable the User to have more control over systems functions.

Short cut: return to start of document

5 Sensor Setup

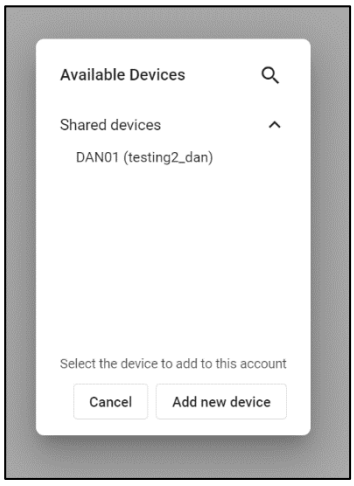
- The sensor setup section enables Users to import and configure individual sensors or groups of sensors.
- To set up a new sensor, it must be pre-registered with UnifAI and connected to the system by API (see Appendix 4).
- There are two ways a new sensor can then be added to an Account:
  - Shared Device: One or more sensors may have already been set up in another Account and shared with the User, or;
  - New Device: The User can add a new device, provided they know the Device ID and Claim Key.
- The User must know what sensors they will be setting up in the account, what parameters those sensors are measuring, and how those parameters are 'named' in the API. If you don't know this information, you cannot proceed.

5.1 Setting up a sensor

- In the menu at the top of the page, click on the **Register** tab.

- In the box on the left of the page, click on the **Plus**  sign.

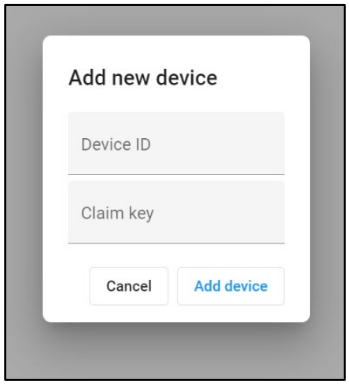
- This will open the **Available Devices** box.
- Any devices that have been setup by another User and shared with your Account will be listed here.



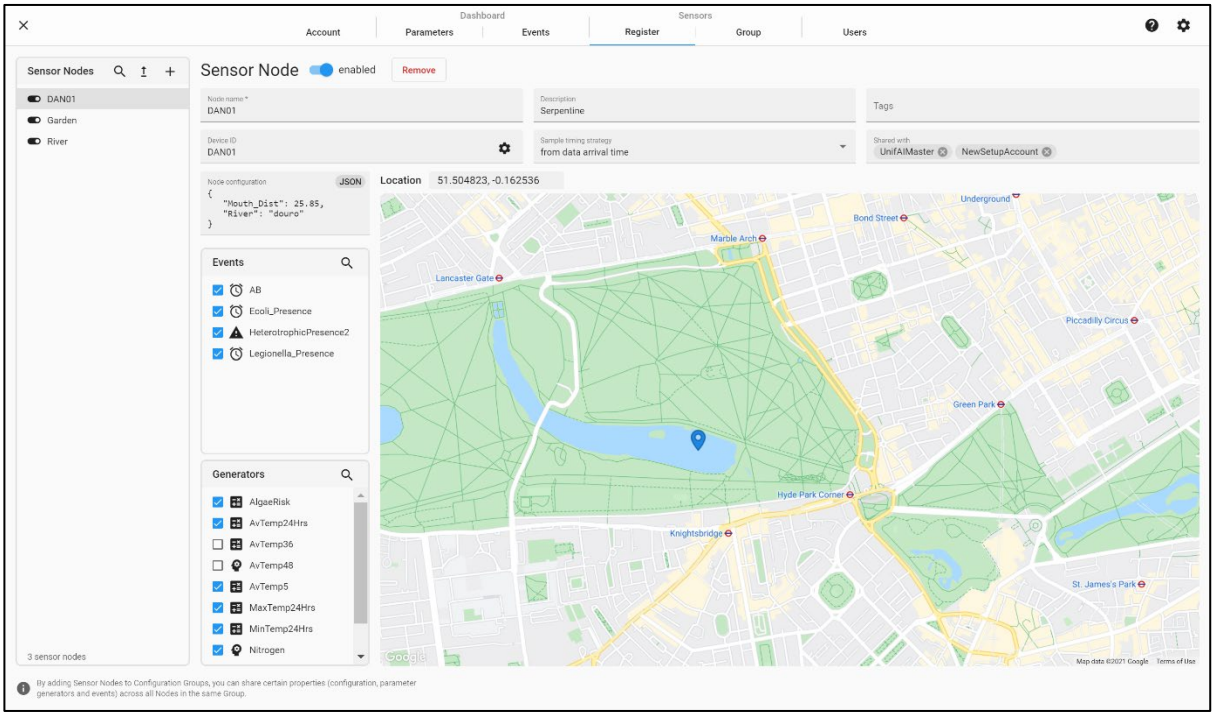
- To set up a shared device, click on the sensor that you wish to set up.

Short cut: return to start of document

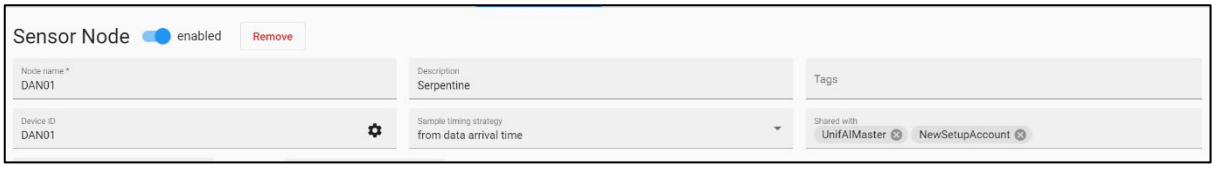
- Alternatively, click on **Add new device** at the bottom of the box. You will require the Device ID and Claim Key for the sensor you wish to add.



- Either of these actions will open a new **Sensor Node** setup page for the sensor you have selected.



- At the top, the **Node Name**, **Description** and **Device ID** are automatically set to the Device ID name of the imported sensor. You can change any of these if required. This provides flexibility so that Users may name and structure their sensor networks in line with their operational processes.



- We recommend you only change the **Description**, which is what the sensor will be called on the Dashboard map.

Short cut: return to start of document

- When a new sensor is added, the system will automatically match the Device ID with the API for the pre-registered sensor. It is therefore important not to change the Device ID unless you know why you are doing so (a Device ID might be changed when a physical sensor is being replaced, and matched to the same configuration and historical data as the old sensor). **Do not change the Device ID for the Test Sensor.**

- Assigning **Tags** is optional and can help with device management, such as tagging sensors by groups (buildings, rivers…), functionality (air quality, water quality…), location (London, Cardiff…), even by month or year of deployment. Tags can be used when searching for specific sensors in the Sensor List in the Map Layout or Grid Layout screens of the dashboard.

- Click on **Sample timing strategy**

- A small drop-down box will open with two options:

    - "**from data arrival time**." This is the default setting. This is for sensors that transmit all data samples at the time they are taken. If this setting is selected, the system will register the time of a data sample as the time that the data arrives in the platform.

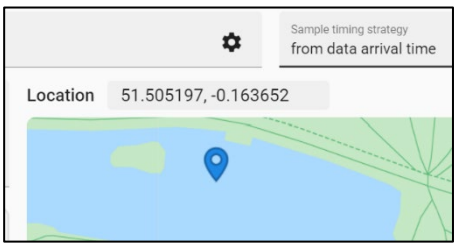
    - "**from field in incoming data**." This is for sensors that may transmit data samples sometime after they are taken. For example, a sensor that takes a reading every 15 minutes and transmits data every hour. If this setting is selected, the system will register the time of a data sample as the time recorded in the API.

- The **Share With** field is only visible if this Account 'owns' the sensor. It enables the User to share this sensor with one or more other Accounts within the company or organisation. The User must know the name of the Account they wish to share with. **Note: DAVE subscription costs are calculated "per sensor per account per year," so sharing a sensor with another Account will result in additional subscription charges**.

- There are two ways to set the **Location** of the sensor:

    - The User can enter the GPS coordinates in DD format (decimal degrees) in the **Location** box, or;

    - The User simply can drag the sensor icon to the right location on the map.

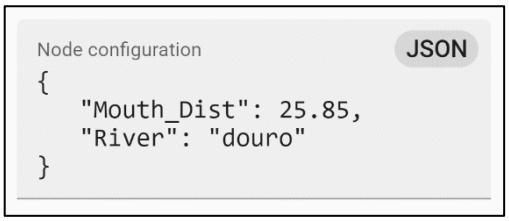


- Below the Device ID name is a field labelled **Node Configuration**. This is used to tell the system which Neural Networks to use for this sensor, based on geography and water type.

Short cut: return to start of document

- **For the purpose of the Tutorial with the Test Sensor, regardless of where you choose to 'locate' the sensor on the map, please write or cut/paste the following into this field**:

```
{
  "Mouth_Dist": 25.85,
  "River": "douro"
}
```

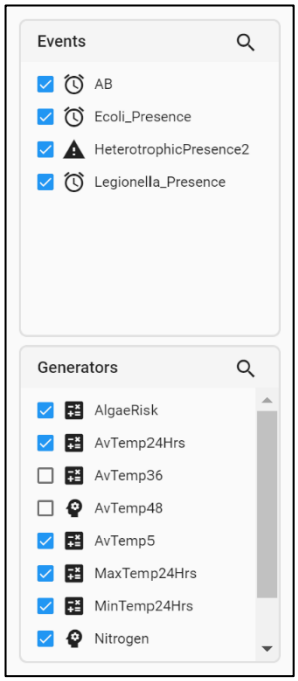


- When you are setting up a live sensor, you will need to speak to your Administrator or to UnifAI who will tell you what to put here.

- Currently, Node Configuration may require either the *river name* and *mouth distance*, or require the correct *Zone,* depending on which Neural Networks are being accessed and used. In future versions of DAVE, Node Configuration will only require the GPS coordinates of the sensor, and the selection of the correct Neural Network will be done automatically.

- If the User has already set up the **Direct Parameters** with the correct parameter names, the Direct Parameter data will automatically appear in the dashboard for this sensor when the next sensor reading is transmitted.

- If the User has already set up **Indirect Parameters** and **Events**, then they can select which Events and Generators the new sensor can access.



- Events or Generators which are not selected will not produce any outputs for this sensor, and any Indirect Parameters in the dashboard which require those Generators will remain blank (these can be hidden in the dashboard).

- For example, a User may have water quality sensors deployed in buildings and in rivers on the same dashboard, and may wish to switch off *Legionella Presence* for the river sensors, and switch off *Nitrogen* for the building sensors.

- This prevents unnecessary Events or Generators from consuming compute power.

- Select **Apply**.

- The new sensor is now set up. Returning to the Dashboard, the User will see the gauges and charts for this sensor. These will be empty until the next data transmission is received from the sensor.

Short cut: return to start of document

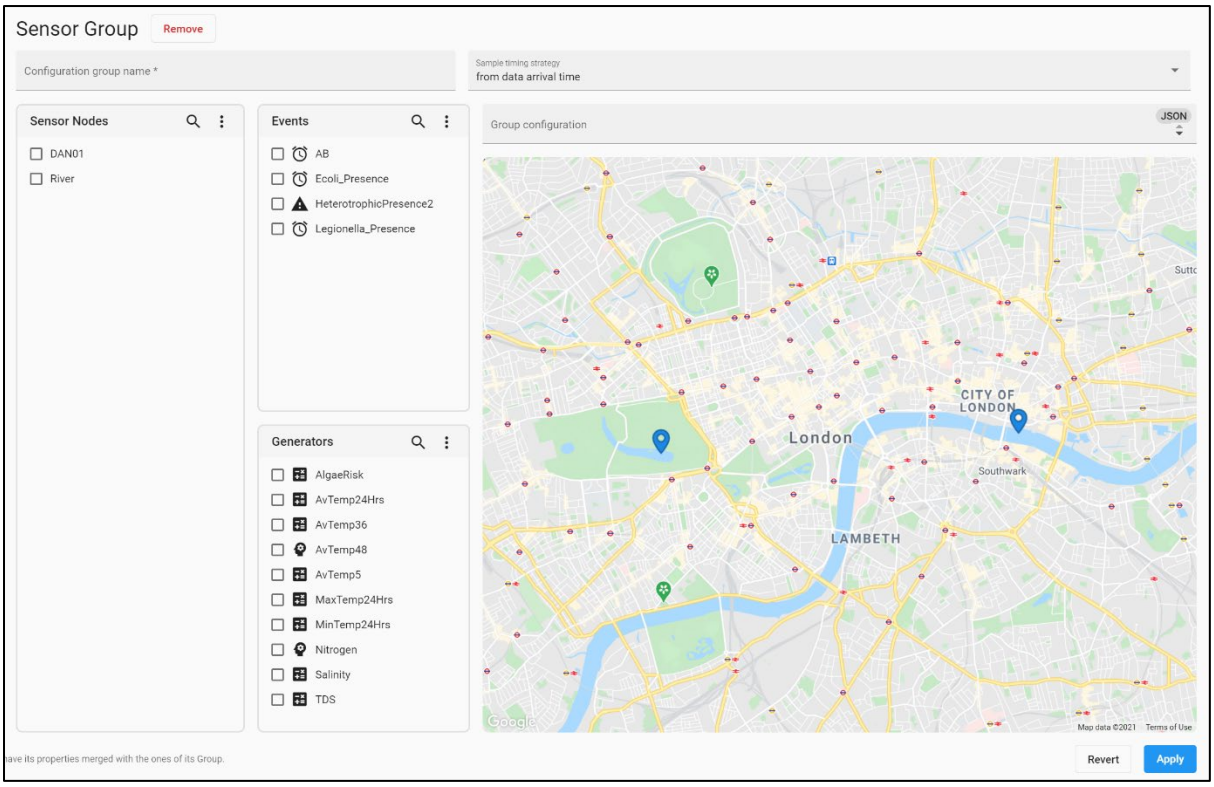
5.2 Setting up a Sensor Configuration Group

- If the User is setting up multiple sensors, they can set up a **Sensor Configuration Group** in order to configure multiple sensors at once, in a batch.

- In the menu at the top of the page, click on the **Group** tab.

- In the box on the left of the page, click on the **Plus**  sign.

- This will open the **Sensor Group** page.



- Set a **Configuration Group Name**.

- Select which sensors to add to this Sensor Group.

- The new Sensor Group can be configured in the same way as individual sensors:

    - **Sample Timing Strategy**.
    - Group **Configuration** (similar to Node Configuration but for a whole sensor group).

Short cut: return to start of document

- Which **Indirect Parameters** and **Events** sensors in this Sensor Group can access.

- As with **Node Configuration** for an individual sensor, **Group Configuration** is used to tell the system which Neural Networks to use for these sensors, based on geography and water type. When you are setting up a live sensors, you will need to speak to your Administrator or to UnifAI who will tell you what to put here. For the purpose of the **Tutorial** if you wish to practice setting up a Sensor Group using the Test Sensor, regardless of where you choose to 'locate' it, please write or cut and paste the following into this field:

```
{
  "Mouth_Dist": 25.85,
  "River": "douro"
}
```

- Select **Apply**.

- The options selected will be applied to all sensors in the group.

- For sensors allocated to a Sensor Group, some options on the individual **Sensor Registration** will be greyed out to prevent conflicting configuration choices.

Short cut: return to start of document

6 User Setup

The User setup section enables the Administrator of the Sub Account to invite additional Users, and to set some limits and controls on those Users' actions.

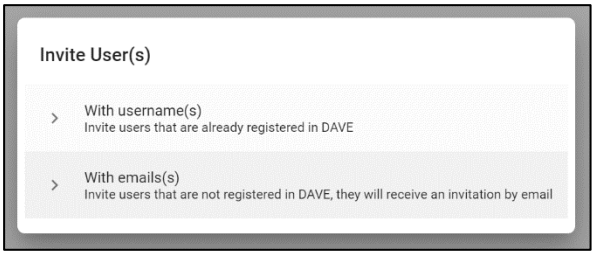
6.1 Adding Users

- In the menu at the top of the page, click on the **Users** tab.

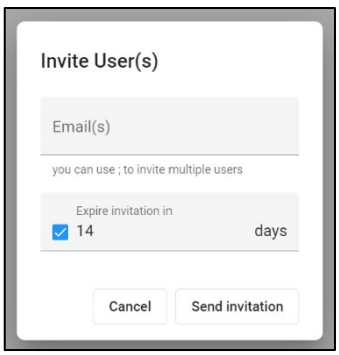
6.1.1 Inviting a new User

- In the box on the left of the page, click on the **Plus**  sign.
- A small drop down box will open.
- Select Invite User(s) from the menu.

- This will open the **Invite User (s)** box.
- If the new User is already a signed-up Dave User, select **With username**.
- This will open a new box which will enable you to send an invite directly to one or more Users. You must know their Usernames to do this.



- The invitees will receive an email with instructions, and will see the invitation the next time they log into Dave.
- If the new User is not yet signed-up with Dave, or if you do not know their Usernames, select **With email**. This will open a open a new box which will enable you to send an invite directly to one or more Users via email.
- The invitees will receive an email with instructions and a unique Sign Up Key to enable them to sign-up for a Username with Dave.



Short cut: return to start of document

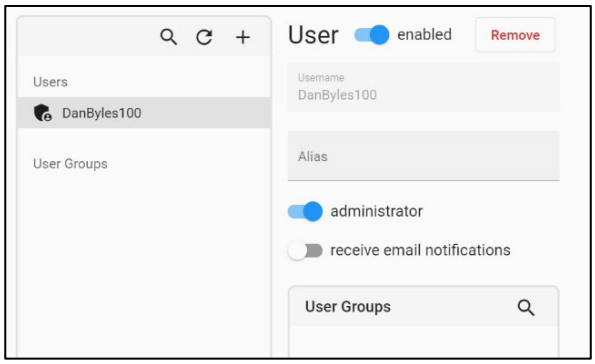
- The Sign Up Key can only be used once, and for added security you can select an expiry period for the invitation.

- The new User will have to accept the invitation before any further User configuration can be completed.

- Until then, if invited by Username they will appear in the User list on the left hand side as *Pending Invitations*.

- The number of Users a company or organisation can have may be limited by their licence agreement. Additional User seats can be purchased if required.

6.1.2 Configuring a new User



- Once a new User has accepted the invitation, their Username will appear in the list of Users on the left hand side of the User page.

- Select a User, and a number of options become available.

- Important note: **the new User will not be able to see any sensors** in this Account unless they are either:

    - Made an Administrator for the Account, or;
    - Connected to a User Group which has sensors enabled for it (see below).

- Toggling **enabled** off allows the Administrator to temporarily restrict access to this Sub Account by this User.

- The Administrator has the option of adding an **Alias** to the Username to help manage and keep track of Users. Such as the person's actual name, or a job title or operational function ("COO, "Facilities Manager" etc)

- Toggling the **administrator** on will make this User an Administrator for this account. This will give them full access to the setup pages and the right to make changes. Toggling this off will make them a basic User for this account. This will allow them to view the data but not make changes to the setup. Note: basic User's must be part of a User Group to see sensors and data.

- Toggling the **receive email notifications** on will determine whether this User receives email alerts if the Account Owner has authorised email notifications to Users when the system detects an Alarm, Warning or Parameter Alert.

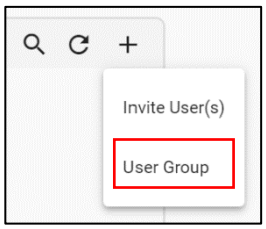
Short cut: return to start of document

6.2 Adding User Groups

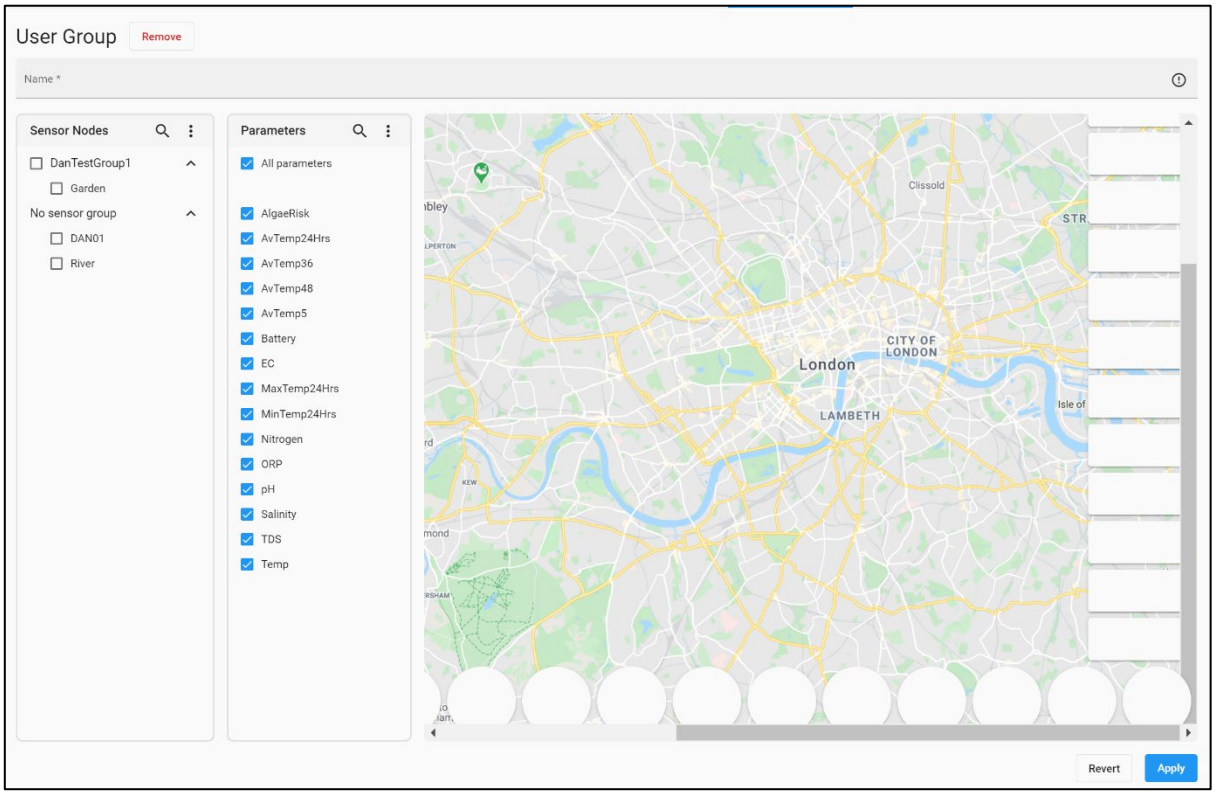
- In the menu at the top of the page, click on the **Users** tab.

6.2.1 Creating a new User Group

- In the box on the left of the page, click on the **Plus**  sign.
- A small drop down box will open.
- Select User Group from the menu.



- This will open the **User Group** page.

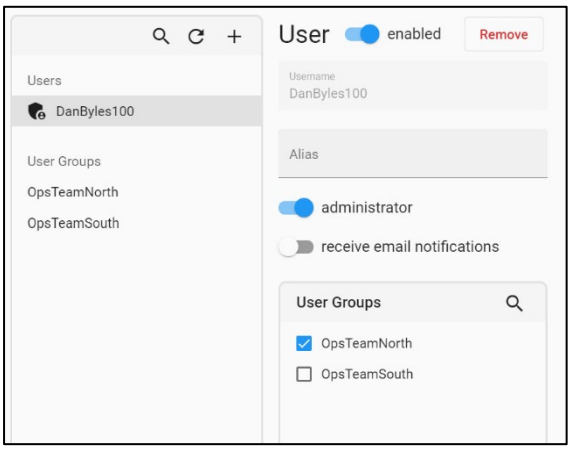


- Set a **Group Name**.

Short cut: return to start of document

- Select which sensors to add to allow this User Group to see. You can add Sensor Groups or individual sensors, or both.
- Select which Parameters to allow this User Group to see.
- Select **Apply**.
- The User Group is now set up.
- If you do not set up a User Group, any basic Users will not be able to see any sensors or data.

6.2.2 Assigning Users to User Groups



- Select a User from the **Users** list on the left.
- Any User Groups you have created will be listed.
- Setting up more than one User Group enables you to allow different basic Users to see different groups of sensors and/or parameters.
- Select which User Groups to assign this User to. Each User can have multiple User Groups, and each User Group can have multiple Users.
- User Groups give the Sub Account administrator fine-grained control over what sensors each User can see, and what data from each sensor each User can see.

Short cut: return to start of document

APPENDICES

Appendix 1 – Glossary of Terms

Account: may describe a **Master Account** or a **Sub Account**.

Administrator: a **User** who can view the dashboard and access all visualisation and analytic functions, and who can access their **Account** Setup pages to setup and configure the account details, parameters, events, sensors, and additional users' access and permissions for that Account.

Alarm: when **DAVE** identifies an **Event**, it can trigger and Alarm or a **Warning**, which is shown visually on the dashboard and can trigger an email notification to **Users**.

ANNA: (Advanced Neural Networks and AI). ANNA is UnifAI Technology's suite of optimised artificial intelligence tools. ANNA is integrated with DAVE, and simplifies the use of powerful AI and analytic tools to unlock new insights and to improve outcomes from the data managed by DAVE.

API: (Application Programming Interface). Software that allows two applications to talk to each other. An API is a mechanism to transfer data into **DAVE**.

Chart: a configurable line graph showing **Parameter** readings for a **Sensor** over time. Charts can show data for a specified day, a specified week, or a customised time period. Charts visually show when a parameter is or has been within certain pre-set thresholds according to a red, yellow and green 'traffic light system.' Charts also show when readings have coincided with an **Event**.

Claim Key: a 'password' unique to each device or sensor and defined in the sensor API, used when connecting a sensor to DAVE. The Claim Key is a security feature that ensures no third party can connect to a sensor and see data without permission.

Data Import Feature: a powerful feature that enables **Account Owners** and **Administrators** to manage the bulk import of existing datasets via CSV files. The feature can quickly organise and import large datasets of up to 100 Gigabytes at a time.

DAVE: (Data, Analytics and Visualisation Engine). DAVE is a proprietary platform built and operated by UnifAI Technology. It enables **Users** to integrate and manage data from multiple sources, including existing datasets and data from third-party platforms, **Sensors** or devices. DAVE is a data platform, user interface and dashboard.

Direct Parameter: a **Parameter** which shows data received directly from a sensor. This may be visualised directly in the dashboard, and/or may be used as the input data for an **Indirect Parameter**.

Device: in the context of **DAVE**, typically a **Sensor**.

Device ID: the unique identifier for a **Sensor**. Device ID is defined in the sensor API, and when registering a new sensor in DAVE, in order to link the sensor configuration in the dashboard with the correct physical sensor.

Event: when DAVE identifies a pre-defined set of conditions or pattern in the data, it creates an Event. This can include complex correlations or patterns of behaviour and change between multiple parameters over time or at a moment in time. An Event triggers an **Alarm** or a **Warning**.

EVA: (Enhanced Visual Analytics). EVA is a dashboard page which allows the **User** visualise correlations between multiple sensor outputs across parameters and sensors over time.

Short cut: return to start of document

Gauge: a visual 'dial' on the dashboard that shows the most recent **Parameter** data for a **Sensor**. Gauges can be configured to show a red, yellow and green 'traffic light system' when a parameter falls into certain pre-set thresholds. Gauges can also show a visual **Parameter Alert** in the form of a red exclamation mark when a parameter is currently 'in the red.'

Generator: generates an **Indirect Parameter** from one or more **Direct Parameters** using **Neural Networks** or mathematical functions to derive additional information from the initial data inputs.

Indirect Parameter: a **Parameter** which is created in **DAVE** from **Direct Parameters** through the use of a **Generator**,

Master Account: one of two types of **Account** (the other being **Sub Account**), and the top tier in the Account hierarchy. Master Accounts can only be created or deleted by UnifAI. The **Owner** of a Master Account can create and delete connected Sub Accounts. A company or organisation may have one Master Account and will typically use it to manage Sub Accounts.

Neural Network: a set of algorithms that learns (or is trained) to recognise underlying patterns and relationships in a set of data through a process that mimics the way the human brain operates. Neural networks are essentially a part of Deep Learning, which is a subset of Machine Learning, which is in turn a subset of Artificial Intelligence (AI). UnifAI Technology uses different types of neural networks and other AI tools as part of UnifAI's AI engine, **ANNA**.

Owner: for every **Account**, one **Administrator** is the Account **Owner**. The Owner is the only person who can authorise the Account to send email notifications. The Owner can pass on ownership of the Account to any other User for that Account. Only Owners of **Master Accounts** can create or delete **Sub Accounts**.

Parameter: a numerical or other measurable factor, representing a physical, chemical or biological variable in the physical world. Parameters in **DAVE** are typically the outputs from sensors, from mathematical functions or from **Neural Networks** which are visualised through **Gauges** and **Charts**, and which can be exported to CSV files or via API to other systems. Parameters in DAVE may be **Direct Parameters** or **Indirect Parameters**.

Parameter Alert: a visual alert in the form of a red exclamation mark which shows both on a relevant **Gauge** and on the Sensor List on the Map Layout and Grid Layout screens when a **Parameter** falls into certain pre-set critical threshold, and is currently 'in the red.'

Quality Index: a powerful and convenient means of summarizing complex data and facilitating its communication. It incorporates three elements: scope - the number of variables not meeting quality objectives; frequency - the number of times these objectives are not met; and amplitude - the amount by which the objectives are not met. The index produces a number between 0 (worst quality) and 100 (best quality). These numbers are divided into 5 descriptive categories to simplify presentation.

Sensor: a device which converts physical parameters into a signal which can be measured electronically. DAVE enables the outputs from sensors to be visualised, analysed, used as inputs for **Neural Networks** and other artificial intelligence tools, and/or exported to other systems.

Sub Account: one of two types of **Account** (the other being **Master Account**), and the bottom tier in the Account hierarchy. Sub Accounts can be created or deleted by the **Owner** of the connected Master Account. The Owner of a Sub Account cannot create or delete Sub Accounts. A company or organisation may have more than one Sub Account, and will typically use them to manage and view sensors and data.

Tag: optional keywords assigned by **Administrators** to **Parameters** and to **Sensors**. A Tag is a piece of information thars adds helpful descriptors and can aid in device management and differentiation. For example, a water quality sensor could be given the tags: water, quality, river, UK, floating. These could differentiate it

Short cut: return to start of document

from air quality sensors, from water depth sensors, from water sensors in swimming pools, from sensors outside the UK and from non-floating fixed sensors. Parameters and Sensors can each have multiple Tags.

Time Machine: a visualisation tool in the dashboard which allows a **User** to compare data over time for multiple parameters and multiple sensors on the same screen. Time Machine enables multiple **Charts** to be viewed simultaneously, for multiple **Parameters** and multiple **Sensors**, and in some cases overlayed onto the same axis.

User: can access **Accounts** to which they have been invited by an **Administrator**. Users can view the dashboard and access all visualisation and analytic functions.

Warning: when **DAVE** identifies an **Event**, it can trigger and **Alarm** or a Warning, which is shown visually on the dashboard and can trigger an email notification to **Users**.

Short cut: return to start of document

Appendix 2 – Formula Examples

- DAVE provides Users with flexible and powerful tools for manipulating and analysing data.
- Administrators can easily create new parameters and new events using mathematical formulas.
- The examples in the table below do not represent an exhaustive list, but show you how formulas work in DAVE.
- Only numerical values can be applied.
- Conditional statements: "if" may be used as a conditional statement.
    - For example: if(P > 1000, 1,2) where if P is greater than 1000 the result returned will be 1, otherwise it will be 2.
- Nested if statements can also be used.
    - For example if (P > 1000, 1, if (P1 < 500, 2,3)
- This can be a very useful tool to fine tune outputs and add validation to limit false positive and false negative alerts that may be triggered by sensor readings. It enables the user to add multi-parameter and conditional intelligence to their data.
- DAVE's JSON function in the parameter setup section will enable the user to convert the numbers into text, and therefore convert sensor readings/multi-sensor readings from numbers into meaningful outputs.
- When referring to a range of parameters which includes the lowest or the highest potential value, use the word "null" instead of a number.
- Additional functions include:
    - acos(); asin(); atan(); cos(); sin(); tan(); exp(); log(x); log(x,base);
- When looking at readings over time:
    - "0" always represents the most recent reading.
    - "-1" represents the previous reading.
    - "-2" represents the reading before that, etc.
- For the examples on the next page, P, P1, P2 and P3 are four illustrative Parameters:
    - P = The last five readings for P are: 5, 8, 13, 9, 10 (with 10 being the most recent reading)
    - P1 = 5
    - P2 = -10
    - P3 = 2.7

Short cut: return to start of document

Function	Purpose	How to use in DAVE
: or ,	**:** (using a colon) means all points between a range. **,** (using a comma) means that only the specified points. This works between [] parenthesis but not when () applies. Therefore either the **:** or **,** may be used with the functions in blue below.	**average(P[-3:0])** will return **10**, which is the average of the most recent reading and the previous three readings. ("-3" = 8, "-2" = 13, "-1" = 9 and "0" = 10) Whereas **average(P[-3,0])** will return **9**, which is the average of the most recent reading the reading three readings before that. ("-3" = 8 and "0" = 10)
+ - * / ^	Returns the product of two parameters. Add (+) Subtract (-) Multiply (*) Divide (/) Power of (^)	**P + P1 = 15** **P – P1 = 5** **P * P1 = 50** **P / P1 = 2** **P ^ P1 = 100,000**
abs()	Returns an absolute number without its sign For example, the absolute number of 10 and -10 is 10	**abs(P2) = 10**
average([])	Returns the average of a parameter over a selected period [-3:0] = 10, 9, 13, 8 [-3,0] = 10, 8	**average(P[-3:0]) = 10** ("-3" = 8 "-2" = 13, "-1" = 9 and "0" = 10). The total of these four numbers is **40** and the average is **10**. Whereas, **average(P[-3,0]) = 9** **:** (using a colon) means all points between a range. **,** (using a comma) means that only the specified points.

Short cut: return to start of document

ceil()	Rounds a number up to the next largest integer	**ceil(P3) = 3** P3 is 2.7 and the smallest integer not less than 2.7 is 3
floor()	The integer or whole number not greater than	**floor(P3) = 2** P3 is 2.7 and the largest integer not greater than 2.7 is 2
round()	Round to the nearest integer	**round(P3) = 3**
count[]	Return the number of elements in a list [-3:0] = 10, 9, 13, 8 [-3,0] = 10, 8	**count[-3:0] = 4** **count[-3,0] = 2**
max([:])	Returns the maximum number from a range	**max(P[-4:0]) = 13**
median([:])	Returns the median number from a range	**median(P[-4:0]) = 9**
min([:])	Returns the minimum number from a range	**min(P[-4:0]) = 5**
::	This function is used to define an interval in time, which may be hours, minutes or seconds. It returns a list of values from the starting point to the end point. In this example all readings from 10h ago to 4 hours ago will be returned. This function is most commonly used with other functions such as .prev, .next, count, .near, average, sum etc.	P[hh(-10) :: hh(-4)] Example: **average (P[hh(-10) :: hh(-4)])**
.prev	Returns the number immediately prior to the interval selected, so in this case prior to 4.5 hours ago	**pH[hh(-4.5)].prev**
.next	Returns the number immediately after the interval selected, so in this case after to 270 minutes ago	**pH[mm(-270)].next**
.near	Returns the closest number to a defined timeframe. In this example it is obtaining the nearest sample to 3600 seconds ago. If hh or mm isn't defined then seconds will be assumed to be the timeframe	**P[-3600].near**

Short cut: return to start of document

sum([])	Returns the sum from a range or from a list [-4:0] = 10, 9, 13, 8, 5 [-4,0] = 10, 5	**Sum(P[-4:0]) = 45** **Sum(P[-4,0]) = 10 + 5 = 15**
[]	Used to get previous samples or range of samples P[-1] = 9 P[-3] = 8	last five samples is **P[-4:0]**

Short cut: return to start of document

Appendix 3 – Data Import Feature

- Details to follow shortly

Short cut: return to start of document

Appendix 4 – DAVE API

How do I send data?

UnifAI Technology has two ways for you to send data to its Data, analytics and Visualisation Engine (DAVE). Both methods can be operated by users with administration access.

The first is using UnifAI Technology's API.

An API is an Application Programming Interface. It is software that allows two applications to talk to each other. An API is a mechanism to transfer data into DAVE. A user can invoke UnifAI Technology's API to send data to DAVE so that it can be visualised and made available for AI work UnifAI Technology's Advanced Neural Network and AI tool, ANNA.

UnifAI Technology's API is a REST API based on HTTPS protocols using JSON strings. The API requires four pieces of mandatory data: format, device name, password and UTC Unix time. The format defines the data structure that follows and is defined by UnifAI. The format shown should cater for almost all scenarios but if there is a reason a different format UnifAI can easily create an alternative format for use within the existing API. The device name can be anything ranging from the mac address to a specific name you choose to call your sensors, which may be numerical or letters that are randomly chosen or have some form of meaning; and similarly, the password can be anything you choose. The naming convention requires _ instead of spaces. The UTC Unix time can be the time of transmission or another relevant time e.g. if the sample was collected prior to transmission, this time can be provided. DAVE will record the time a message is received as well as the time you send and a user can choose to display data based on the sensor sample time, or the time the data is received, depending on use case.

An API will typically be used when the data is generated by a sensor. To get data from sensors into DAVE using the API there is a two-step process:

- First you register your sensors in DAVE using UnifAI Technology's simple Sensor Management Portal (SMP). You need to register your sensors before you send data so that data from those sensors can be accepted.

- Secondly, and once you have registered your sensors, you can begin to send sensor data to DAVE using our fully configurable API.

Data can be sent from any source that can invoke our API, including sensors, databases, and proxy servers. You just need to embed our API into the software that is sending data. Sensors that are approved by UnifAI Technology and bought through our partners will have the sensor credentials registered in DAVE and you will be provided with the sensors.

Short cut: return to start of document

The API has a simple structure, as follows:

https://dave2.unifaitechnology.net:5001/v2.0/sensing/sample: POST {
"unifai_format": "Unifai_20_2",
"device_name": "SENSOR01",
"device_password": "password",
"utc_unixtime": 1611164083,
"values": {
"EC25": 1176,
"pH": 6.5,
"ORP": 169.2,
"Temperature": 26.3
}
}

In the values, you can send as many parameters as you wish. If the parameters you send are set-up in in DAVE, the data posted will automatically show up in your dashboard. If the parameter hasn't been set-up it won't be recognised and will be rejected. If the format is incorrect the whole message will be rejected.

Once an message has been sent to DAVE via the API a message will be returned as follows:

201 "x samples were created"
400 "Bad request"
400 "Unable to process request"
404 "No sensor nodes match incoming data"
500 "Internal server error"

The second method to provide data is to use DAVE's data import tool. This tool enables users to upload data using CSV files, and has a three-step process.

- The first is to define the format of the data by confirming if the data is stored vertically or horizontally; and, which rows or columns contain sensor names, time, location and which contain parameters. E.g. after selecting horizontal, the user may have rows 1 for time, 2 for longitude, 3 for latitude, 4 for sensor name and 5 to 10 for parameters.

- The second step is to validate the data. DAVE will automatically compare the parameter headings in the CSV file to see if they are already defined in DAVE. Any that aren't already defined will be presented back to the user so they can either select an existing parameter definition e.g. change Electroconductivity to EC, or create a new parameter, as required.

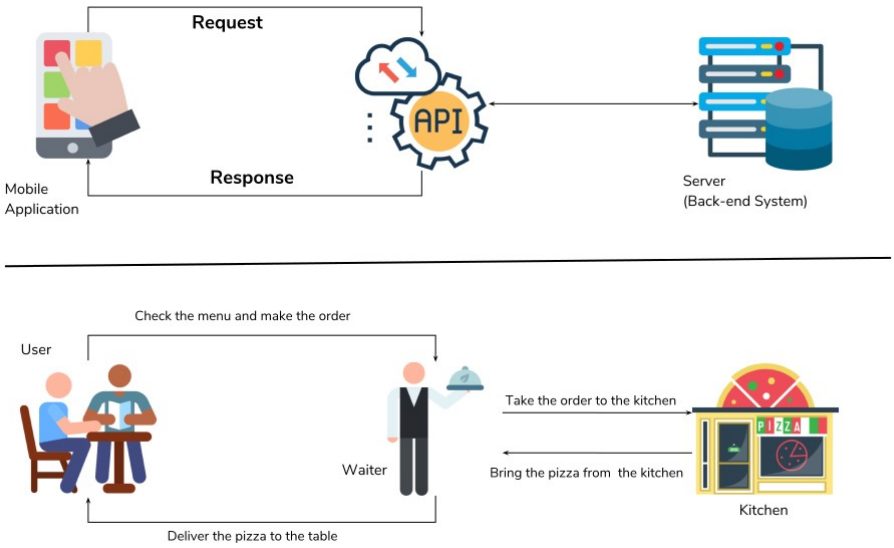
- The final step is to then import the data at which point it automatically becomes available to view in DAVE.

Users may wish to both import historic sensor data as well as direct new data into DAVE through the API. To do this the user will set-up the sensors in the SMP first, and then import the historic data and invoke the API.

Short cut: return to start of document

Data Security

UnifAI Technology takes the security of data very seriously. Data is transmitted using HTTPS encryption and the level of security on sensors and devices has been enhanced with a capability to add fully configurable sensor credentials to your sensor data. You simply register your devices with UnifAI Technology through the SMP, provide your chosen naming convention and passwords. Data can then only be received or sent with the credentials you have created to ensure bogus data isn't posted to your account, or data isn't extracted from your account without using those credentials. You can easily apply your own security policy by changing device passwords for your sensors using the UnifAI SMP tool.



Server

General considerations on API's endpoints.

url: dave2.unifaitechnology.net Port: 5003/5001 SSH Content-Type: application/json

Outputs

JSON

```
{
    ...
    "error": "error message, in case of error"
    "status_code": HTTP_STATUS_CODE,
    "status_message": "status message",
}
```


Short cut: return to start of document

Authentication
Bearer Token (JWT), in HTTP Headers

```
{
    ...
    Authorization: "Bearer " + TOKEN
}
```

Errors (all routes)

```
403, "Not allowed"
```

API: Login Existing User

The login endpoint returns a token to be used in other endpoint calls.
In order to use the endpoint for getting samples, there must be a login to the server. Login tokens have a lifespan of about 24 hours, so, for continuous use, a new user login should be issued regularly (ex: every 12h), in order to renew existing token.

Port: 5003

Route:

```
/dave/v2.0/auth/login    [POST]
```

Fields (json):

```
username
password
```

Output: 200

```
token
first_name
last_name
```

Errors:

```
400, "Invalid request"
400, "No username"
400, "Invalid username/password"
400, "No password"
401, "Invalid username/password"
```

Example:

```
curl -X POST "https://dave2.unifaitechnology.net:5003/dave/v2.0/auth/login" -H
"accept: application/json" -H "Content-Type: application/json" -d "{
\"username\": \"usr\", \"password\": \"pwd\"}"
```


Short cut: return to start of document

API: Get Samples from DAVE

This endpoint allows fetching all the samples from a given account 's sensor nodes, within a given period. This should be issued with "min_timestamp" with the date from the last time since the endpoint call has been issued, for a regular cumulative use. Be careful not to override the server with too wide periods or too much samples as depending on the account type there may be safety limits on this call.

Port: 5003

Route:
```
/dave/v2.0/samples      [GET]
```

Auth:
```
token
```

Fields (params):
```
account_id
(sensor_nodes) = List[id,]
min_timestamp
(max_timestamp)
```

Output:
```
samples = List[ ... ]   // Same as v1.1
```

Errors:
```
400, "Invalid request"
400, "No account"
400, "Must have 'min_timestamp'"
404, "Invalid contract"
401, "Permission denied"
```

Example:
```
curl -X GET
"https://dave2.unifaitechnology.net:5003/dave/v2.0/samples?min_timestamp=2020-07-
21%2009%3A00&account_id=2" -H "accept: application/json" -H "Authorization:
Bearer <TOKEN DATA>"
```

Parameters:

For continuous use, new samples should be fetched regularly from the last sample's date (samples are ordered by timestamp). Data quotas are to be applied to ensure there is no misuse of this API endpoint (ex: fetching all samples frequently).

Request parameters are the following:

- "account_id": account from which samples are to be obtained.
- “sensor_nodes”: optional list of sensor node id's.
- “minTimestamp”: UTC start time from which samples are obtained. (ex: “2019-01-18 16:59:02”).
- “maxTimestamp”: optional parameter with UTC end time up to which samples are obtained.

Output:

This request will return a JSON string like the following:

Short cut: return to start of document

```
{
  "samples": [
    {
      "alarms": [],
      "sample_id": 285717,
      "snode": "C2C441",
      "snode_id": 65,
      "timestamp": "2021-02-01 00:29:31",
      "utc_offset": 0,
      "values": {
        "Arsenic": 1.76851785182953,
        "BaseEC": 14363.003018,
        "Battery": 2,
        "EC": 25457.29,
        ...
        "Salinity": 14.6545073539949,
        "TDS": 12728.645,
        "Temp": 3.21,
        "WQI": 100,
        "Zinc": 10.0928239822388,
        "pH": 7.35
      }
    },
    {
      "sample_id": 285801,
      "snode": "C2C441",
      "snode_id": 65,
      "timestamp": "2021-02-01 03:31:07",
      "utc_offset": 0,
      "values": {
        "Arsenic": 1.76928961277008,
        "BaseEC": 14341.510896,
        "Battery": 2,
        "EC": 25555.08,
        ...
        "Salinity": 14.6989914308753,
        "TDS": 12777.54,
        "Temp": 3.06,
        "WQI": 100,
        "Zinc": 10.1353912353516,
        "pH": 7.34
      },
      "alarms": [
            {“Event”: “Effluent_Discharge”, “Message”: “Acidic Cond.”},
            {“Event”: “Effluent_Discharge”, “Message”: “Elevated Deriv.”}
        ]

    },
    "status_code": 200
    "status_message": "Samples returned"
}
```


The properties within this string are the following:

- “samples”: the list of returned samples.

Each sample has the following fields:

- “sample_id”: unique identifier of this sample.
- "snode_id": sensor node ID

Short cut: return to start of document

- “snode”: sensor node name.
- “timestamp”: UTC time when this sample was collected (ex: “2019-01-18 16:59:02”).
- “utc_offset”: sensor local time offset in minutes, in relation to UTC time.
- “alarms”: list of alarms that have been triggered by this sample. Check the example from the last sample above. Each alarm has properties “Event” (type of alarm event) and “Message” (specific warning message).
- "values": dictionary of {PARAMETER_NAME: VALUE}.

API: Send Sample to DAVE

This is the endpoint for sending a sample from a sensor node.

Port: 5001

https://dave2.unifaitechnology.net:5001/v2.0/sensing/sample: POST {

```
    "unifai_format": "Unifai_20_2",
    "device_id": "prefix:SENSOR01",
    "access_key": "password",
    "utc_unixtime": 1611164083,
    "values": {
        "EC25": 1176,
        "pH": 6.5,
        "ORP": 169.2,
        "Temperature": 26.3
    }
```

}

Short cut: return to start of document