Six things to look for in a Hydroponic sensor data logging system

Data is key. It will help you obtain high yields and improve with each additional crop cycle. Having sensor measurements not only allows you to diagnose your crop at any given point in time but also allows you to go back and figure out what might have happened if something went wrong. With all the commercial offerings now becoming available, it is starting to become harder and harder to evaluate which data logging system might be ideal for you. In this post, I seek to share with you 5 things that I always look for when evaluating data logging systems for a greenhouse or grow room. These are all things that will enable you to store sensor data adequately and take full advantage of it, ensuring you're not handy capped by a poor starting choice.

Sensor compatibility. One of the first things that I look for is which sensors I can add and what restrictions I might have on sensors that are added to the system. I like to have systems where I can connect any 3-5V analog sensor I want. I also want to be able to connect sensors that use common protocols, like i2c sensors. I also like to know that for things like pH and EC, the boards have standard plugs I can connect to, to make sure I can replace the electrodes given to me by the company with others if I wish to do so. Freedom in sensor compatibility and in the ability to replace sensors with sensors from outside the company are both a must for me.

Expandability. Many of the commercially available data logging platforms are very restricted and can often only accommodate a very small number of sensors. Whenever you're looking for a data logging solution that will need to be deployed on a medium/large scale, it is important to consider how this implementation can expand, and how painful it would be to make that expansion. Being able to easily add/remove sensors to a platform is key to having a flexible and robust data logging solution.

Not cloud reliant. It is very important for me to be able to use the system, regardless of whether the computers are online or not, and to have all the data that I register logged locally in some manner. Systems where an internet connection is needed for data logging or where data is not stored locally are both big show stoppers when it comes to evaluating a data logging system. There is nothing wrong with having data backed up to the cloud – this is indeed very desirable – but I want to ensure that I have a local copy of my data that can I always rely on and that logging of data won’t be stopped because there is some internet connection issue. Also bear in mind that if your sensors are cloud reliant you will be left without any sort of data logging system if the company goes under and those servers cease to exist.

Connectivity of sensors is robust. In many of the more trendier new systems sensor connectivity is wireless. This can be perfectly fine if it is built robustly enough, but it is often the case that connections based on WiFi will tend to fail under environments that are filled with electromagnetic noise, such as when you have a lot of HPS ballasts. It is therefore important to consider that if you have such an environment, having most of your sensors connected using cables, or using a wireless implementation robust to this type of noise is necessary.

Have a robust API to directly access your data. Since I do a lot of data analyses using the data from hydroponics crops, I find it very crippling to be limited by some web interface that only allows me to look at data in some very limited ways. I want any data logging system I use to allow me to use an API to get direct access to the data so that I can implement a data structure and analysis the way I see fit. Having your data available through a robust API will allow you to expand the usage of your data significantly and it will also ensure you can backup your data or structure the database in whatever way you see fit. An example of this is sensor calibration logging and comparisons, while commercial platforms almost never have this functionality, having an API allows me to download the data and compare sensor readings between each other to figure out if some sensors have lost calibration or make sure to schedule their calibration if they haven’t been calibrated for a long time.

Ability to repair. When making a data logging choice, we are making a bet on a particular company to continue existing and supporting their products in the long term. However, this is often not the case and we do not want to be left with a completely obsolete system if a company goes under and ceases to support the product they made. I always like to ensure that the systems that are being bought can continue working if the company goes under and that there is a realistic ability to find parts and replace sections of those products that might fail in the future if this were to be the case. *Open source products are the most ideal because of this fact.*

These are some of my top six priorities whenever I evaluate a commercial data logging solution for deployment. From the above, not being cloud reliant and having a robust API are the most important, while sensor compatibility can be ignored to an extent if the system is only being deployed for a very specific need (for which the sensors provided/available are just fine). Which of the above you give the most priority to depends on how much money you’re going to be investing and how big and robust you want the implementation to be.