Sensor System Research

Arduino UNO Overview:
-$22 on arduino store
-14 digital I/O pins (6 can provide PWM output)
-6 analog I/O pins
-O2 sensor would be analog input
-I/O pins allow 20mA current each (input or output) and operate at 5V
-powered by USB (type B, 5V) or external 9V power supply (through VIN pin, 7-20V, or DC power jack, 7-20V)
-accepted power supply is 7-20V
-3.3V pin allows 50mA
-32kB flash mem, 0.5kB of which allocated to bootloader
-2kB SRAM
-16MHz clock

Pin specifics:
-digital pin 13 drives an LED on the board (active high)
-VIN pin, alternate input voltage for external power supply
-5V pin outputs a constant 5V from the board's regulator, this can be used to power the board but avoid doing so as it bypasses the regulator and can damage the board
-3.3V pin outputs a constant 3.3V from the board's regulator, max current draw is 50mA
-GND pins are ground
-IOREF pin outputs the MCU's ref voltage, useful mainly for shields to choose which voltage to operate at
-RESET, similar to IOREF, useful for shields that need their own reset/block the Arduino's reset
-AREF is the reference voltage for analog inputs
-Each I/O pin can be configured as an input or output using *pinMode()*, *digitalWrite()*, and *digitalRead()* functions
-Each I/O pin has an internal 20-50kOhm pull-up resistor (disabled by default)
-Each I/O pin has an absolute max operating current of 40mA
-The 6 analog input pins each have a resolution of 10 bits (2^10 = 1024 values)
    -they can each measure from ground to 5V
        -the upper limit of this range can be modified using the AREF pin and *analogReference()* function

-Pins 0 and 1 are for UART RX and TX data (serial), respectively
-Pins 2 and 3 can be used to trigger on external interrupts, and are configurable to trigger on high, low, rising edge, etc.
-Pins 3, 5, 6, 9, 10, 11 can provide 8-bit PWM output using the *analogWrite()* function
-Pins 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK) are for SPI (serial peripheral interface) communication (not sure what this is)
    -SPI library

-Pins A4 or SDA and A5 or SCL are for TWI (two-wire interface) communication (not sure what this is either)

-Wire library

Refs:

https://en.wikipedia.org/wiki/Arduino_Uno

https://store.arduino.cc/usa/arduino-uno-rev3

O2 sensor:

The only option available on Homecenter:
https://www.homecenter.com.co/homecenter-co/product/348515/Medidor-Oxigeno-Disuelto-Temp/348515

https://www.google.com/search?q=water+oxygen+sensor&tbs=vw:l,ss:44&tbm=shop&ei=A8ioXafRCMWt5wK32begDg&start=20&sa=N&ved=0ahUKEwjn3seviqTlAhXF1IkKHbfsDeQQ8NMDCN8E&biw=1366&bih=625

Marcos:
-potentially make this more mobile, less setup
-ideally would lower cost, but this would likely necessitate arch changes that aren't really wanted
-the overall system itself is in a good state, only really issues seem to be the usability/mobility of the system and its overall cost
-cheaper/better sensors are likely outside of scope, something that a dedicated science + engineering hybrid project would need to solve (not solely a product focused project like this one, as there are many other applications for sensors)
-want to iterate, not redesign
-USER INTERFACE, CELL PHONES?
-timer since last cycle
-suggested dosages… algorithm problem?

Initial thoughts are that the main areas of improvement are related to the power source (finding a more mobile solution with less setup) and the mobile app. Improvement to the app can be the addition of more information (time since last system cycle, etc.) and suggested adjustments (does the system need to be cycled, dosage of chemicals to add, etc.).

POWER STUFF:

From https://edge.rit.edu/edge/P19762/public/FinalDocuments/TechnicalPaper.pdf . . .

The entire system requires 9V and 0.15A, total of 1.35W. Power per day required is 32.4Wh.

Current power supply provides 7.4V and contains 2.6Ah for a total of 19.24Wh.

Nine-Volt battery (lithium ion, rechargeable) nominal voltage is 7.4V, provides 0.62Ah for a total of 4.588Wh.

D batteries (potentially not rechargeable) nominal voltage is 1.5V, and provide 4.5Ah for a total of 6.75Wh.

Potentially just use a rechargeable power brick (like for phones), and power the Arduino over USB???

Even if flow rates don't work out (system is unfeasible) if we can at least prove that we can achieve correct ammonia, oxygen etc. levels we will have learned something useful