ECE 331

Project 2 - Temperature Logger

See course web site for due date

Grade reduced 1% per minute after the due data, before the start of class, rounded up to the next minute.

Items	Details	Value	Score
Deliverables **25%**	All code, scripts, and documented configuration		
	Printed using enscript	5%	
	4 space tabs	5%	
	Name in header	5%	
	Git		
	All code, scripts, and Makefile in git	5%	
	Reasonable commit comments	5%	
Design **55%**	SQLite3 operation	7%	
	Perl/Python/C (or combination) code to query sensors	8%	
	JS operation	8%	
	ChartJS plot	8%	
	lighttpd setup	6%	
	Comments	6%	
	Errors checked and handled correctly	6%	
	Organized data representation	6%	
Demo **20%**	Works as described - page loads, renders, and is correct at the time of grading. Depends on public internet access.	20%	
	TOTAL		

Description

For this project, create a temperature, humidity, pressure, and light data logger and plotter on the web. The data source is the ECE331 expansion HAT. Data values can be read from the RPi built-in serial port on the GPIO header. The serial ports appears as `/dev/ttyAMA0` at 38400 8N1. On the RPi 3, bluetooth must be disabled and the built in serial port enabled using `raspi-config`. Sending any single character to the ECE331 expansion HAT will cause the ECE331 expansion HAT to print all sensor values. Example sensor values are shown below

`29.937500`	High resolution temperature, degrees Celsius (`29.937500` ℃)
`7082`	IR Intensity, unsigned 16 bit raw value
`22650`	Full Spectrum Intensity, unsigned 16 bit raw value
`15568`	Visible Light Intensity, unsigned 16 bit raw value
`582`	Illuminance, Lux (582 Lux)

3031	Temperature, degrees centiCelsius (30.31 ºC)
9997816	Pressure, millibars * 10^4 (999.7816 mb)
22223	Relative Humidity, millipercent (22.223%)

Write a program in C, perl, python, C++, java, ... or any combination except command line, to obtain the sensor values. Insert the obtained sensor values, and the date and time of the reading into a SQLite3 database using C, perl, python, C++, java, ... or any combination except command line. Add new data every minute. Setup and run `lighttpd`. Be sure to have it start at boot time. Write a javascript script in ChartJS that graphs the data for the last 24 hours worth of data. Setup your RPi so the page is accessible from the public internet. Secure your RPi so that it is not disabled from the network.

Comment all of your code/scripts. Use git to manage revisions for all of your code, scripts, documentation, Makefiles, and any other electronic document. Include all relevant configuration including commands within the documentation. Include the instructor of the course as a "Developer" member. Name the repository "`datalogger`".

Deliverables

At the due date, turn in your
- Call code and script committed and commented in gitlab
- Emailed URL of the project

Demo

Email the URL of your system to the instructor before the due date. URL must be accessible on the public internet.

Grading

Grading will be done by the rubric given on the first page.