

EE 629 – IoT using Raspberry Pi

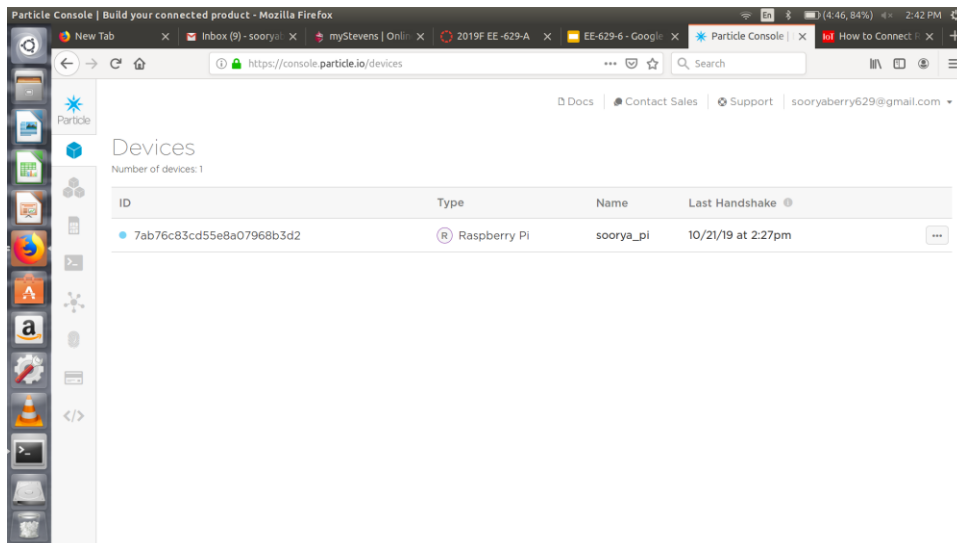
Lesson 6 – Exercises - **Done**

Particle cloud:

The Particle Cloud is the powerful centerpiece of the Particle platform, handling many of the most complex pieces of creating an IoT product.

Lab A: Particle Cloud

- Created a Particle account at <https://www.particle.io>
- Installed the Particle-Agent on Raspberry Pi, log in the Particle account, and registered Raspberry Pi to the Particle account
- Toggled the green activity LED on Raspberry pi HIGH or LOW



```
pi@soorya: ~
Get started by running "particle login"
If that doesn't work, open a new terminal and make sure /home/pi/bin is in your
shell path.
If you previously installed the CLI with npm, run "npm uninstall -g particle-cli"

pi@soorya:~ $ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/local/games:/u
sr/games
pi@soorya:~ $ export PATH="/home/pi/bin:$PATH"
pi@soorya:~ $ particle login
? Please enter your email address sooryaberry629@gmail.com
? Please enter your password [hidden]
> Successfully completed login!
pi@soorya:~ $ particle call <device_name> digitalwrite D7=HIGH
-bash: device_name: No such file or directory
pi@soorya:~ $ particle call soorya_pi digitalwrite D7=HIGH
1
pi@soorya:~ $ particle call soorya_pi digitalwrite D7=LOW
1
pi@soorya:~ $ particle logout
? Would you like to keep the current authentication token? Yes
> Leaving your token intact.
> You have been logged out from sooryaberry629@gmail.com.
pi@soorya:~ $
```

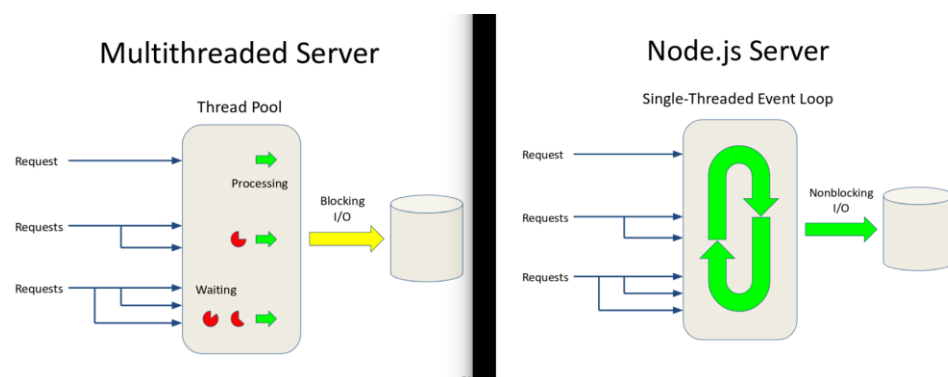
Node.js

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js = Runtime Environment + JavaScript Library

Features:

- Asynchronous and Event Driven
- Very fast and no buffering
- Single Threaded but Highly Scalable



Lab B: Node.js

Ran Node.js server at <http://127.0.0.1:8080> on Raspberry Pi