Proposed Solution Description

Hardware

- ESP8266_01: We will use the ESP8266 version 1 wifi chip in all work. http://tronixlabs.com/iot/wifi-serial-transceiver-module-with-esp8266/
- ESP8266_12 with Breakout: Additional we will verify and include, if necessary, alternate instructions when using the version 12 chip. http://tronixlabs.com/wireless/esp8266/esp8266-esp-12-full-evaluation-board/
- USB to serial 3.3 volt: Used to connect the chip to pc for flashing and programing, and also to supply power. http://tronixlabs.com/usb-serial/ftdi-basic-breakout-5v-3-3v-micro-usb/
- Breadboard
- ___ Placeholder (sensors like led etc.…)

Software

- ESPTool: a python utility to communicate between esp8266 rom bootloader and a pc. https://github.com/themadinventor/esptool
- NodeMCU-firmware: A firmware for the esp8266 that uses lua to more simple issue commands and takes advantage of features. https://github.com/nodemcu/nodemcu-firmware
- Python 2.7
- ___ Placeholder

Features

- ESP8266 Introduction manual.
    - Hardware overview, describing the features of the chip.
    - Pin layout, detailing how to power and the usage of all pins.
    - Usage instructions.
    - Useful knowledge, such as knowledge gained from use like different ways to power the chip.
    - Links to communities and good git repositories.
- Windows installer for setting up an ESP8266 usage environment. This installer will setup all dependencies like python; install the IDE and interactive tutorials.
- Lua crash course tutorial that is basic enough to be quick, but covers all lua used in the tutorials.
- ESP8266 wiring and power up guide.
- ESP8266 Flash guide.
- Several practical tutorials using the ESP8266, of increasing complexity and covering as much of the chip features as possible. Possible projects include;
    - Flashing an LED over a web server (demonstrates flashing, injecting code, connecting and wiring gpio pins).
    - Detecting if a shower is running or not (demonstrates using analogue sensors, two way communication).

    - Mesh network using ESP8266's to track some items position. (demonstrates more advanced use of the chip for more novel ideas).
- A library / API of commonly used code.

Solution Architecture Model

TODO

Deliverables

- ESP8266 Introduction manual
- Windows installer for esp8266 environment
- ESP8266 tutorials and documentation for all code
- Common Usage API and documentation for all code

Development Plan

**see fortnightly plan for more accurate adaptive plan*

1. Research current information on ESP8266
2. Follow existing tutorials on setup and usage of ESP8266
3. Create installer for setup of all required software under windows
4. Write basic usage tutorial including connecting and recommended hardware
5. Develop, complete and review spikes for more advanced usage
6. Reflect gained advanced knowledge in more advanced tutorials
7. Repeat step 5 and 6 as time allows
8. Refine best projects/tutorials to highest level of quality
9. Encapsulate common code into an API
10. Deliver