**Rpi Credentials(Host Node):** **OctoPrint Login (Admin):**

**User: Will** **User: Makerspace**

**Pass: 0ctopr!nt** **Pass: 0ctopr!nt**

**Control Node:**

* **(OptiPlex 7050 w/16GB RAM running Ubuntu)**
  + **OctoFarm installed to allow control of all Host Node OctoPi instances**
    - DNS/ Static IP: 10.1.1.2
  + Upstream of all Host Nodes via Cisco 48-Port POE+ Ethernet Switch, connected via Ethernet Cable.
  + VLAN?
  + **Admin/User Login Support**
    - Admin Login hidden to edit preferences & parameters
    - Student Login
      * Is this where students will be able to check out printer?
      * Upload Print / Check-in on status of print
      * Display anticipated cost of print:
        + Based on pre-determined filament type
        + Amount of filament needed for print can be obtained from gcode (double check this—plugin available?)
        + Total Cost calculated by with above variables in mind

Need to be able to account for and include any re-prints to capture the true amount of filament used per upload.

**Host Node(s):**

* **Rasperry Pi 4B running Raspberry Pi OS**
  + **Powered by POE+ HAT** via Network Switch (30W Max, 25.5W Typ)
  + **Running 2x OctoPi Instances**, connected to 2x 3D Printers.
    - OctoPi instances will be mapped to coinciding 3D Printer & Webcam.
      * Serial Port & Webcam stream tied via TBD naming convention.
        + Need to make sure this approach can support a hot-swap scenario in the future if the node needs to be replaced in a pinch. Minimal amount of customization per setup is ultimate goal.
      * Each Instance outputs to CSV
    - *Physical Connections:* USB 2.0 (low bandwidth, ~0.4 Mbps)
    - *USB 2.0 Spec Reference:*Well below max practical throughput of 480 Mbps
  + **Running 2x Webcam Streams**, connected to 2x Logitech C270 or PiCam
    - Using *MJPEG encoding*, controlling webcams via UVC (USB Video Class)
      * MJPEG typical bitrate: ~0.07–0.15 bits/pixel
      * Bandwidth ≈ Resolution × FPS × Bits per pixel
        + Low compression estimate (more quality): 0.15 bpp
        + Moderate compression estimate: 0.10 bpp
    - Lower Resolution: 480p (640x480) @ 15 FPS à ~92-138 Mbps
      * Slightly higher framerate, but resolution quality may be lacking for future AI interpretation.
    - Higher Resolution: 720p (1280x720) @ 10 FPS à ~184-276 Mbps
      * Slightly lower framerate, but higher resolution would lend itself better for future AI interpretation, and 10 FPS should still be enough granularity to still allow for a decent timelapse if desired.
    - Resolution Considerations:
      * Depending on what you’re hoping to use it for, 720p may better.
      * AI models are all spatial feature-driven — especially those used in vision tasks like:
        + Detecting spaghetti extrusion
        + Identifying first-layer problems
        + Spotting detachment/lifting
        + Monitoring nozzle tracking
      * **720p gives ~3× more data per frame than 480p**
        + This gives you better accuracy in detecting Spaghettification, Curling Layers, Nozzle Misalignment and Blob Formation.
        + Higher Resolution would also reduce false negatives.
    - *Physical Connetions:* USB 3.0 (higher bandwidth, ~276 Mbps)
    - *Reference:*Well below max practical throughput of 2,500 Mbps

**Network Switch:**

* **Cisco Catalyst 2960 X-Series 48-Port Ethernet Switch (WS-C2960X-48LPS-L V04)**
  + POE+ (IEEE 802.3at, 370W total, 30W Max, 25.5W typ. per port)
  + Will Power each Host Node via POE+ hat
    - Max cable length of 100m/328ft as per spec
    - Guaranteed 5V @ 4A (20W) after regulation at each Host Node

Plugins:

* Octolapse (Needs printer specific information to work properly – this might be a non-starter)
* Mjpeg-streamer
* Obico (AI Detection)