



Secure and Remote 3D Printing

By: Tiffanie Petersen, Isaiah Thomas, Carl Mann, & Nick Cottrell

General Information:

Meeting Dates:

- 2/1/22
- 2/14/22

Faculty Sponsor

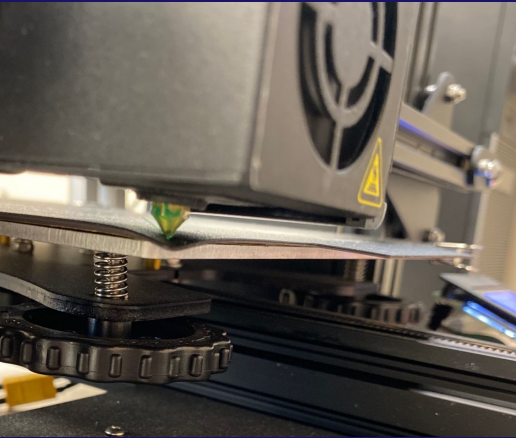
Dr. Sid Bhattacharyya

Client

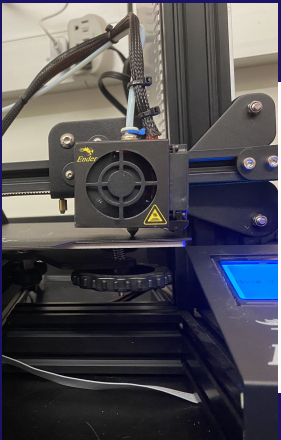
Mike C Newton

Task	Completion %	Tiffanie	Carl	Isaiah	Nick
1. Implement, test & demo use of the raspberry pie to print	100%	75%	0%	25%	0%
2. Implement, test & demo the website interfacing with octoprint's api	100%	0%	40%	60%	0%
3. Utilize the GreatFET to intercept and inject traffic	10%	0%	10%	0%	0%
4. Improve and expand on features of web application (gcode viewer and printer control)	80%	0%	50%	30%	0%
5. Update the Dockerfile to include Octoprint & Web application	100%	0%	0%	0%	100%
6. Fix the 3D printer from the last group	100%	100%	0%	0%	0%

Errors Encountered

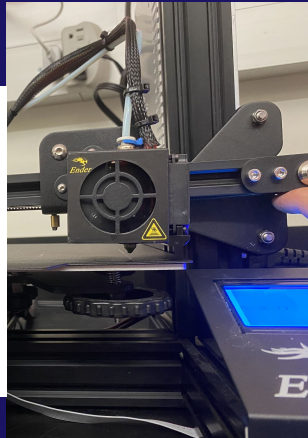


Extruder pushing on the bed causing it to bend



Extruder arm was not level causing it to rest on the bed.

(level shown on the right)



Connection

Serial Port

/dev/ttyACM0

Baudrate

AUTO

Printer Profile

Ender 3 Pro

☐ Save connection settings

☒ Auto-connect on server startup

Connect

State

State: Offline after error

Could not write to serial port

File:

Uploaded:

Timelapse: -

Approx. Total Print Time: -

Print Time: -

Print Time Left: -

Printed: -

Print

Pause

Cancel

Serial port was not being acknowledged

Versioning collisions

- Our repository failed with django 4
- We program with django 3, but requirements.txt wanted latest version of django
- Docker image couldnt build due to this
- Fixed by updating our code to django 4 and adding version markers to requirements.txt

```
1  django
2  octorest
3  |
```

```
1  asgiref==3.5.0
2  certifi==2021.10.8
3  charset-normalizer==2.0.11; python_version >= '3'
4  django==4.0.2
5  idna==3.3; python_version >= '3'
6  octorest==0.4
7  requests==2.27.1
8  sqlparse==0.4.2
9  urllib3==1.26.8
10 websocket-client==1.2.3
11 |
```

Raspberry Pi & Docker

- Raspberry pi is set up with Raspbian lite
- Uses docker containers for website and octoprint
- Working on compose file to connect website and octoprint together

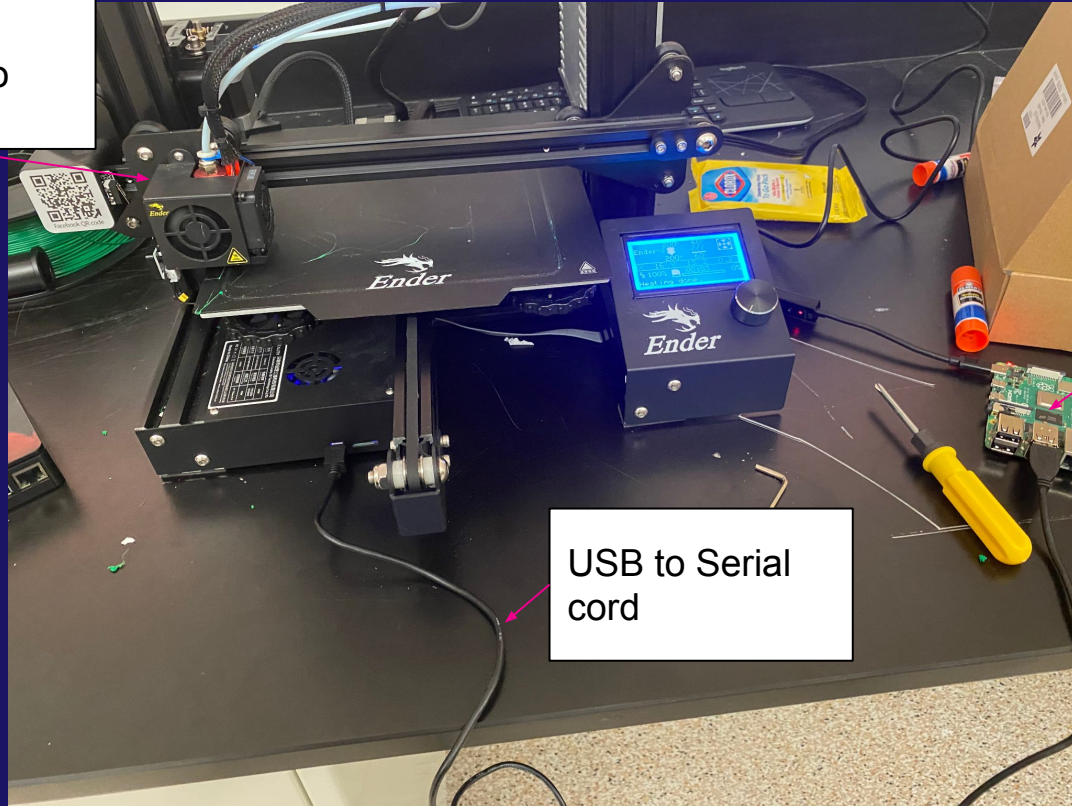
```
docker-compose.yml
version: '2.4'

services:
  octoprint:
    image: octoprint/octoprint
    restart: unless-stopped
    ports:
      - 80:80
    devices:
      - /dev/USB0:/dev/USB0
    volumes:
      - octoprint:/octoprint
  secureprinting:
    build: .
    image: secureprinting:local
    restart: unless-stopped
    ports:
      - 8080:8000
    volumes:
      - configs:/app/data
      - uploads:/app/uploads

volumes:
  octoprint: {}
  configs: {}
  uploads: {}
```

3D printer set up

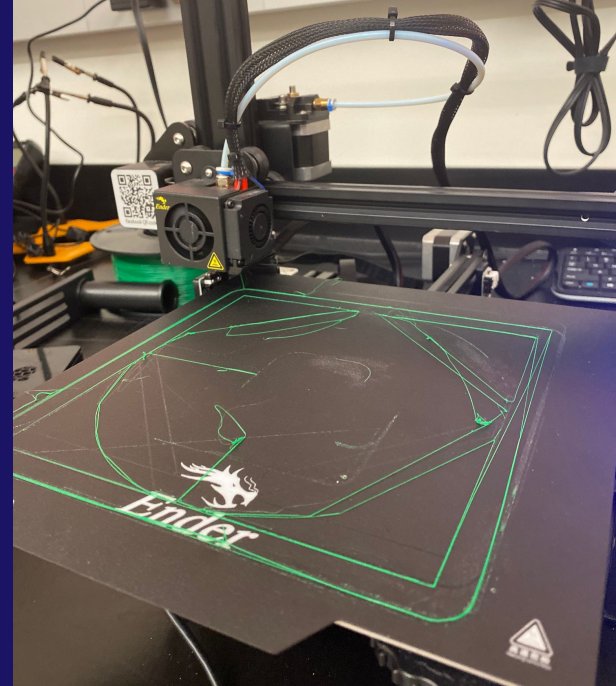
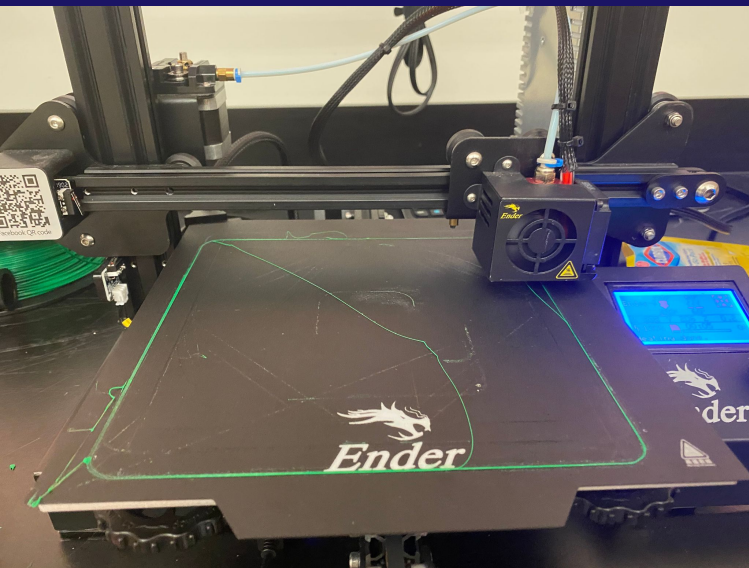
Ender 3 Pro



Raspberry Pi 4

USB to Serial
cord

Calibration Tests



API Implementation

- Added functionality:
 - Website establishes client connection to octoprint upon startup
 - Uploads files to printer upon approved upload
 - Deletes files from printer as well as queue
 - Starts printer operation
 - Pauses or Resumes operation
 - Queries printer for information regarding current state of operation

```
7 def make_client():
8     """
9     Creates and returns an instance of the OctoRest client.
10    Parameters:
11        url - the url to the octoprint server
12        apikey - the apikey from the octoprint server found in settings
13    """
14    url = "http://192.168.1.208"
15    apikey = "21BA190BCA9E49289245D9D0B36C9CE1"
16    try:
17        client = OctoRest(url=url, apikey=apikey)
18        print("\nSuccess -- connection to OctoprintAPI was established\n")
19        return client
20    except:
21        print("\nWarning -- connection to OctoprintAPI could not be established...\n")
22
```

OctoREST API

Query client for
printer state

Operational /
Not Operational

```
#Check printer status
if client:
    status = "Nominal."
    try:
        state = client.printer()['state']
        if state['flags']['printing']:
            status = str(state['text']) + " / Currently printing!"
        else:
            ready = state['flags']['ready']
            if ready:
                status = str(state['text']) + " / Ready to print!"
            else:
                status = str(state['text']) + " / Not ready to print!"
    except:
        status = "Client currently disconnected."
else:
    status = "Not so nominal."
```

Printer State

Name	Multiplicity	Type	Description
text	1	String	A textual representation of the current state of the printer, e.g. "Operational" or "Printing"
flags	1	Printer state flags	A few boolean printer state flags
flags.operational	1	Boolean	true if the printer is operational, false otherwise
flags.paused	1	Boolean	true if the printer is currently paused, false otherwise
flags.printing	1	Boolean	true if the printer is currently printing, false otherwise
flags.pausing	1	Boolean	true if the printer is currently printing and in the process of pausing, false otherwise
flags.cancelling	1	Boolean	true if the printer is currently printing and in the process of pausing, false otherwise
flags.sdReady	1	Boolean	true if the printer's SD card is available and initialized, false otherwise. This is redundant information to the SD State.
flags.error	1	Boolean	true if an unrecoverable error occurred, false otherwise
flags.ready	1	Boolean	true if the printer is operational and no data is currently being streamed to SD, so ready to receive instructions
flags.closedoreError	1	Boolean	true if the printer is disconnected (possibly due to an error), false otherwise

Currently
printing?

Ready state

OctoREST API

- Administrator Control:
 - Website displays the status of the printer and prevents operations without an established client connection
 - Operational
 - Printing
 - Ready to print
 - Not Operational
 - Client not established
 - Cancelling current print

Printer status: Not so nominal.

Show entries

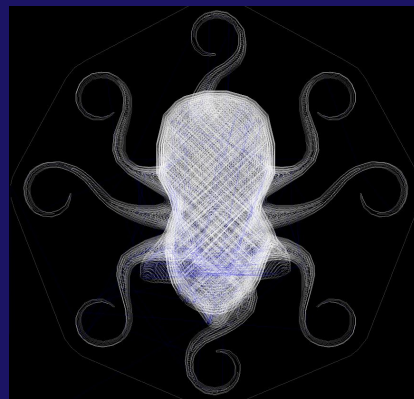
Search:

Upload Date	Uploaded By	Size(bytes)	Download File	Start Print	Pause	Process
Feb. 13, 2022, 9:36 p.m.	admin@admin.com	42658	Calibration	<input type="button" value="Start"/>	<input type="button" value="Pause"/>	<input type="button" value="Delete (for now)"/>
Feb. 14, 2022, 4:55 p.m.	admin@admin.com	154891	boom	<input type="button" value="Start"/>	<input type="button" value="Pause"/>	<input type="button" value="Delete (for now)"/>

Gcode Viewer and Printer Control

- Added modeling of GCODE files using Javascript's 3D modeling library
- Renderer allows full rotation and zooming on the model while showing line paths of print

Download File	Visualize
Octocat	<button>View</button>
Calibration	<button>View</button>
Cube	<button>View</button>



Life of a GCODE File

1 Connection

- Website creates a client to handle octoprint operations

2 Upload

- Website `"/data/"`
- Octoprint `"/api/files"`

3 Queue

- Administrator views the model and (if approved) starts the print

4 Print

- Printer processes the given GCODE file
- Client provides the user with time updates

5 Cleanup

- On print job completion files are deleted from the printer
- User is notified of completion

Milestone 5

Task	Tiffanie	Carl	Isaiah	Nick
Enhance website appearance	50%	0%	0%	50%
Final website functionality	0%	50%	50%	0%
Deploy website	25%	25%	25%	25%
Begin MitM of 3D printer	25%	25%	25%	25%
Create the poster for the project	75%	0%	25%	0%

**Any
Questions?**

Thank You!