THE GOAL - 目標

The goal of this project is to create a timelapse video of a plant growing, using a raspberry pi, cameras, and a temperature/humidity sensor.

このプロジェクトの目標は、ラズベリーパイ、カメラ、温度/湿度センサーを使用して、植物が成長するタイムラプスビデオを作成することです。

COMPONENTS NEEDED - 必要なコンポーネント

Pot - 壺

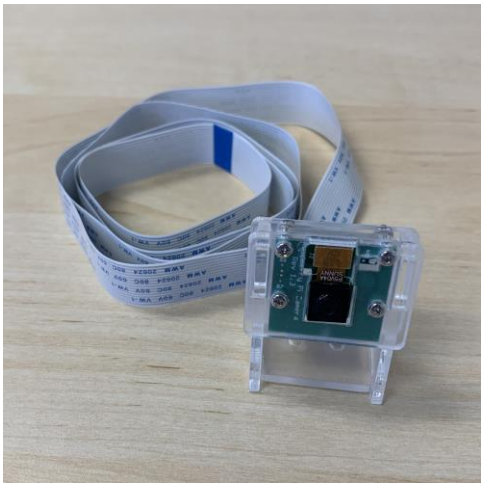
Soil - 土壌

Seeds - 種

Small Table - 物置台

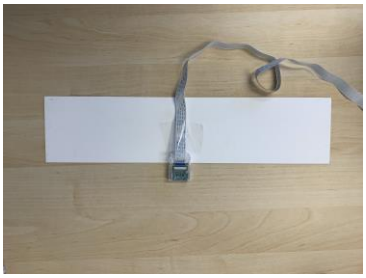
USB Webcam - USB ウェブカム

PiCamera - パイ・カメラ



Masking Tape - 養成テープ

Poster Board - マットボード

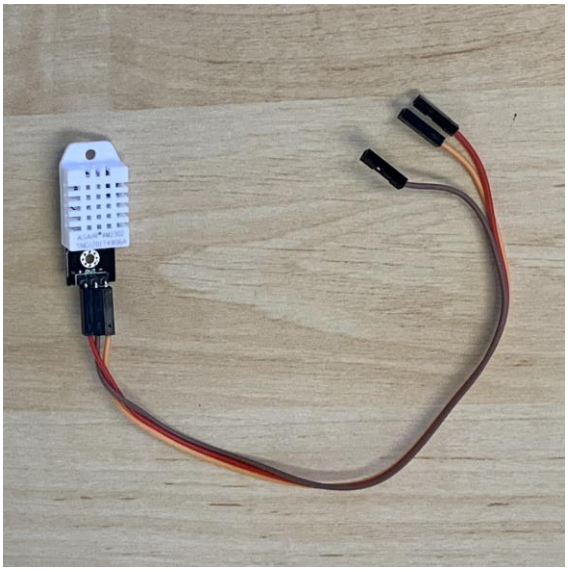


Tripod - 三脚

Saucer- 受け皿

Raspberry Pi - ラズベリーパイ

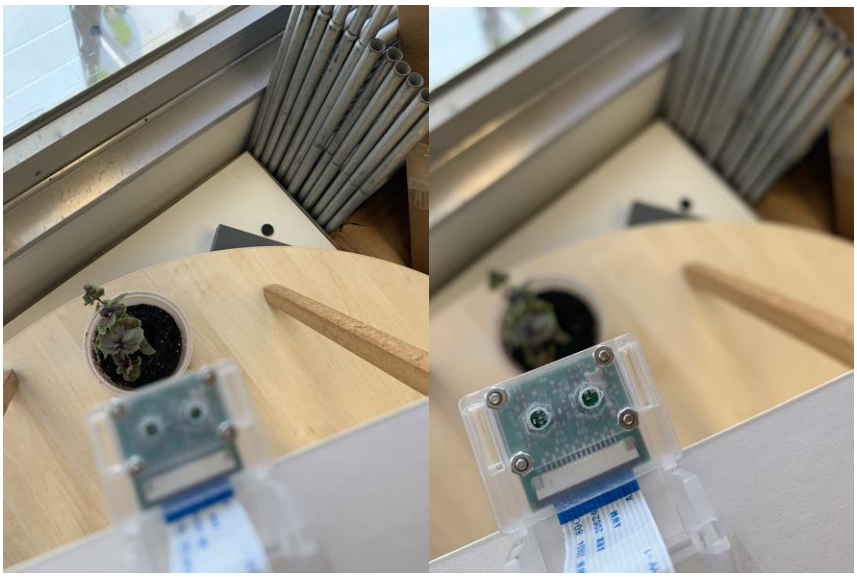
DHT-22 Sensor - DHT-22 温度/湿度センサー



Sunlight - 太陽

PHYSICAL SETUP - 物理的なセットアップ

The main challenge was to set up the equipment and cameras in a way that I could get two clear camera angles without cables getting in the way. I positioned the raspberry pi and DHT-22 sensor on a small table with cross legs on top of a larger table. Between the cross legs I created a platform made out of matboard for the PiCamera to take nadir shots. I used masking tape to afix the camera to the board. I put the portrait view camera on the main table on top of a tripod, again using masking tape as the webcam had no screw mounts. I had to take several test shots to get the cameras to center the shot.



CODE: TAKING PICS AND TEMP READINGS - コード：写真と温度の読み取りを行う

Picam Nadir Shot

USB Webcam Shot

I created a single python script to handle taking pictures and temperature/humidity data. It also takes the two images, blends them into one image, and overlays the current number of days

that have past since the start of the project. The code is contained in the snapshot.py file.

CODE: BACKUP DATA OFFSITE - コード: オフサイトデータバックアップ

In the beginning, I feared that the images I was taking would soon fill the relatively small 8GB micro SDHC card, so I decided to back up the images at the end of the day on a NAS that I keep at home. For this, I used Python's ftplib library. See file dailysweep.py

CODE: VIDEO CREATION - コード: ビデオ作製

I intended to create a video each day and upload it to YouTube before deleting it, but ran into difficulties. I decided to handle this process manually so I wouldn't fill the disk with 240MB video files. The code for this file is found in processimages.py

Final video: https://youtu.be/kBFOXsLvERw

SCHEDULING - ジョブスケジューラ

Use the command: **sudo crontab –e**

sudo crontab -e　のコマンドを使って以下の行を入力する。

First, make the picture-taking script execute every ten minutes.

まず写真と温度の読み取りを行うコードを１０分に一回実行するように設定する。

***/10 * * * * /usr/bin/python3 /home/pi/refactored/snapshot.py**

Then, make the image backup script execute just after midnight every day.

次、毎日夜中の１２時に写真をバックアップするコードを実行するように設定する。

0 0 * * * /usr/bin/python3 /home/pi/refactored/dailysweep.py

LESSONS LEARNED - 学んだ教訓

I used this project to learn things that I hadn't learned to do in Python. I learned:

1. How to hook up and capture data from a sensor
2. How to use the PILlow library
3. How to use the ftplib for FTP'ing files
4. How to control a PiCam via Python
5. How to make system calls to capture images with a USB webcam
6. Learned that FPS rates were inversely proportional to length of project in days in that the longer the project ran, the faster we would need to display the images in order to prevent sleep.

FUTURE IMPROVEMENTS - 将来の改善

I would like to make several changes in the future:

1. Refactor code and abstract directory and camera data into JSON objects
2. Use a smart switch to turn a dumb light on and off right before and after taking the picture, so I don't waste electricity at night.
3. Hook up an arduino to do the actual watering process for me.
4. Generate video automatically via a cron job
5. Upload the video automatically to Youtube on a regular basis
6. Email Youtube link to video as well as text message it to me.
7. Experiment with flowers and other plants that sprout more impressively.
8. Overlay temperature information in a graphical form on each frame.