SWE30011-IoT Programming

Individual Assignment (Practical)

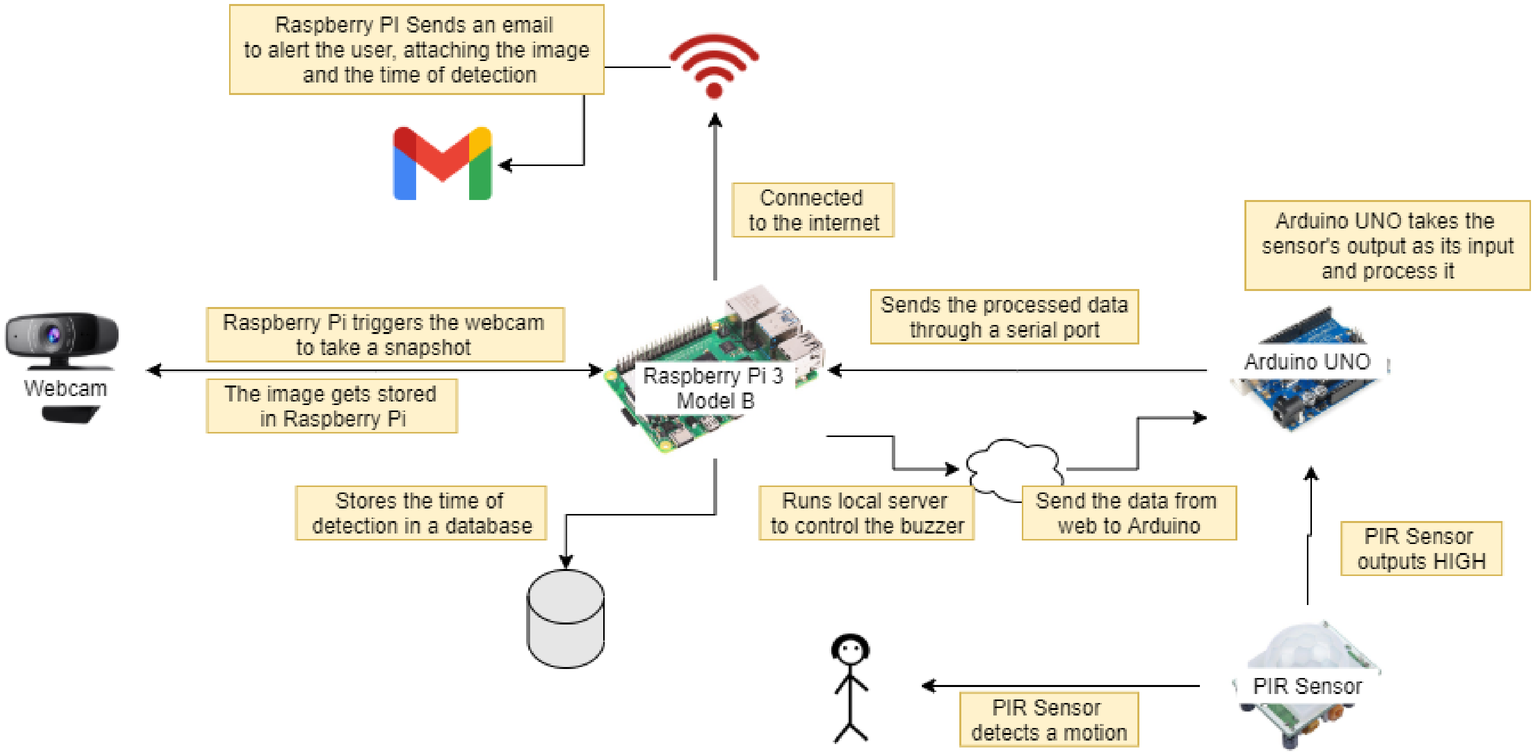
Motion Detecting System

Panvina Touch || 101928802

Introduction

As someone who lives with family members, maintaining privacy can be tricky as keeping my door locked every time could create questions about trust in the family. However, leaving my door unlocked without me home could potentially have my siblings take things without my permission, or have my mum finds things I do not want her to find. A system has been proposed to tackle this issue. The system detects motion when a person enters the room. It will take a snapshot of them and send an email address to notify about the detection with the attached image. The time of detection will also be stored on a database for future references. This allows me to monitor and trace who enters my room at certain time, and I will be able to act accordingly should I find something missing from my room. I could also access the local server to sound the alarm if I find that there is a detection. The system includes a raspberry pi, an Arduino, PIR, a webcam, and buzzer.

Conceptual Design



Raspberry PI Sends an email to alert the user, attaching the image and the time of detection
Connected to the internet
Arduino UNO takes the sensor's output as its input and process it
Raspberry Pi triggers the webcam to take a snapshot
Sends the processed data through a serial port
Webcam
Raspberry Pi 3 Model B
Arduino UNO
The image gets stored in Raspberry Pi
Stores the time of detection in a database
Runs local server to control the buzzer
Send the data from web to Arduino
PIR Sensor outputs HIGH
PIR Sensor
PIR Sensor detects a motion

Implementation

Components Used

- Arduino UNO – receive the output from the PIR sensor. It will forward the data to Raspberry pi for processing.
- Raspberry Pi – It receives the data from a serial port. If the data that it receives confirms that a PIR sensor has detected something, it will command the camera to capture an image of that instance, it processes a notification by emailing to alert with the image attached. Then, the time of detection will be stored in a database.
- Webcam – to take snapshots of the motion. The images are stored in the storage of Raspberry Pi as it will attach them in the alert email.
- PIR Sensor – consists of two halves to intercept the change of IR to detect whether an object is nearby. If one half detects more IR than the other, the sensor will output HIGH. Otherwise, if both halves detect the same IR, it will not react as there is no motion. The sensor will send its output to the Arduino.
- Buzzer – is controlled by a local web server in which data received from the web is passed through the raspberry pi and then back to the Arduino to process.
- Accessible Wifi

Pseudo Code of Proposed System

```
1. Read sensorData from the serial port
    a. If sensorData == ‘1’:
        i. Initialize the camera
            1. capture the image
            2. save the image
        ii. close the camera
        iii. send email(email address, subject, content + current time)
        iv. connect to the database (localhose, username, database name)
            1. insert the current time into the table
            2. close connection
2. serial.close()
```

Controlling the Buzzer in Arduino From the Website

```
1. Opens the serial port
2. Initialize web access
3. If the website toggles a ‘turn on’
    a. Write ‘1’ serial port
    b. Change the buzzer status on the website to currently ‘turned on’
4. Else
    a. write ‘0’ to serial port
    b. Change the buzzer status on the website to currently ‘turned off’
5. Start the web service
```

From Arduino

```
1. If the received data on Arduino == ‘0’or ’turn on’
    a. Turn on the buzzer
2. Else
    a. Turn it off
```

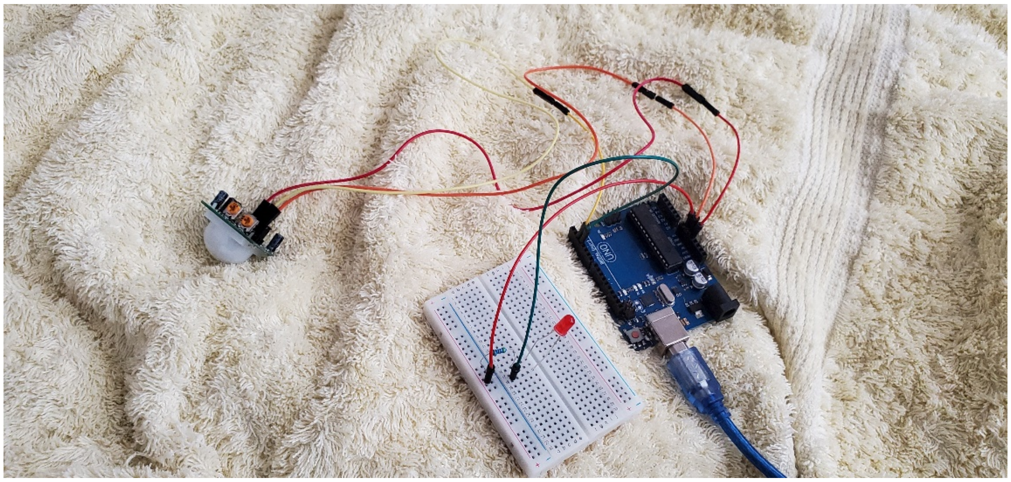


Fig 1. Initial setup: Led light connection is used to run tests on the PIR sensor

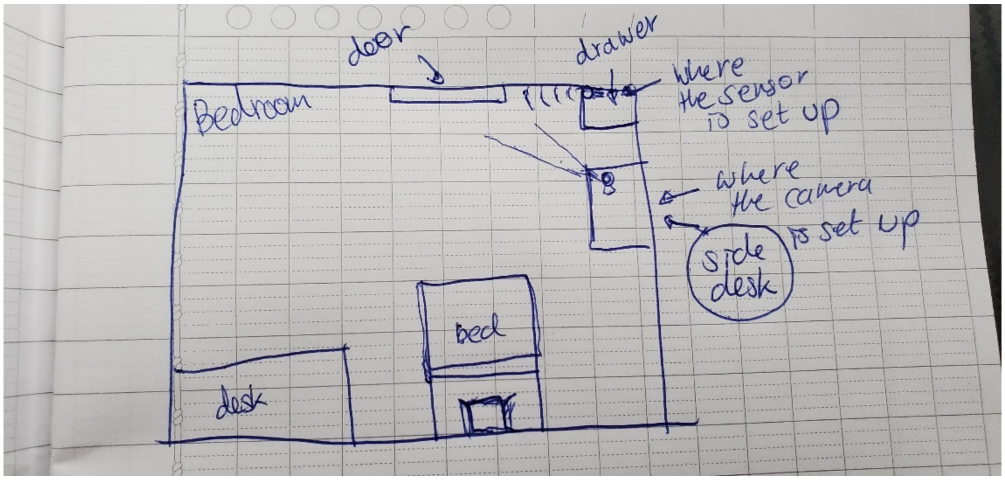


Fig 2. bird's-eye view.

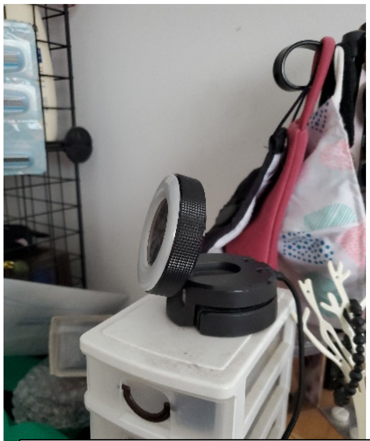


Fig 3. Camera is pointing towards the door.

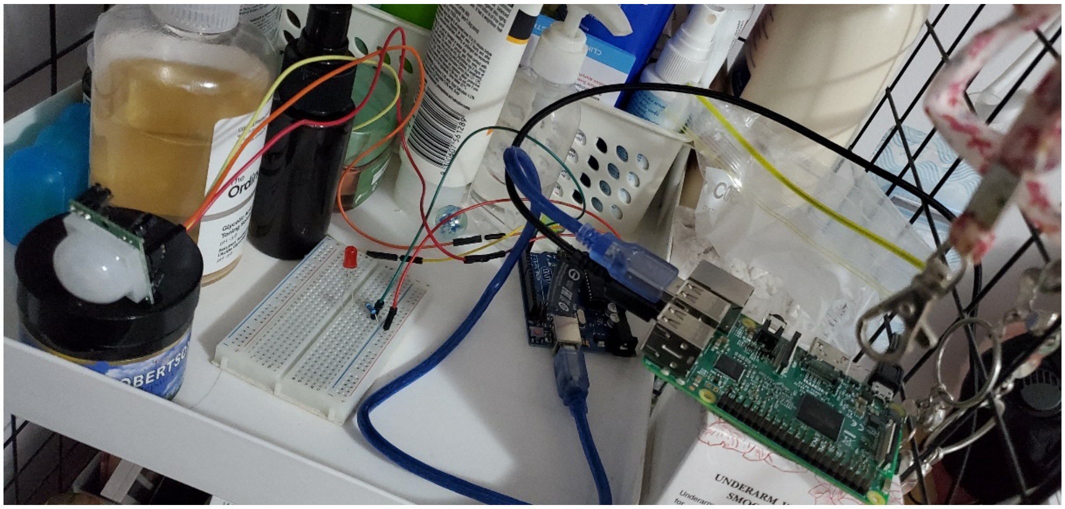


Fig 4. The sensor system is set up in the drawer to hide from view.

Fig 2, 3, 4. A sketch of where it is set up in the bedroom – in paper and in real life.

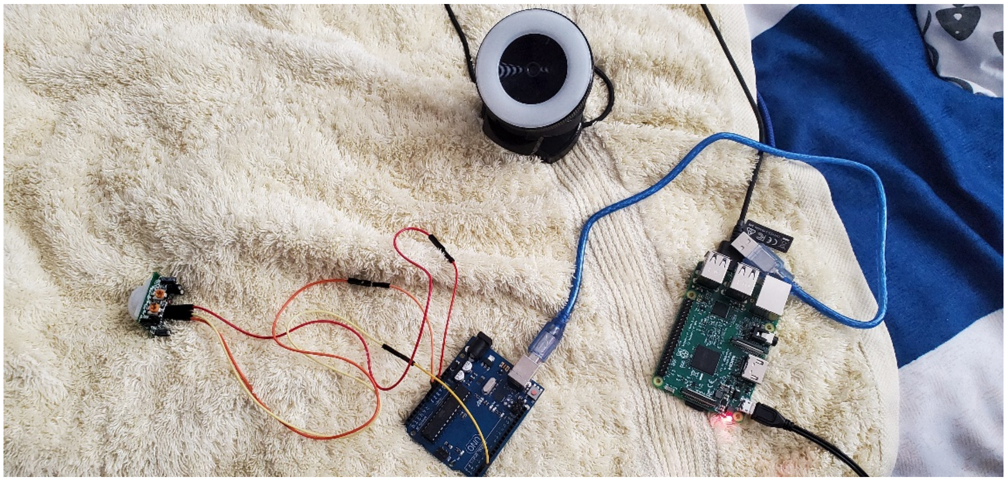


Fig 5. Hardware Setup Before Adding the Buzzer.

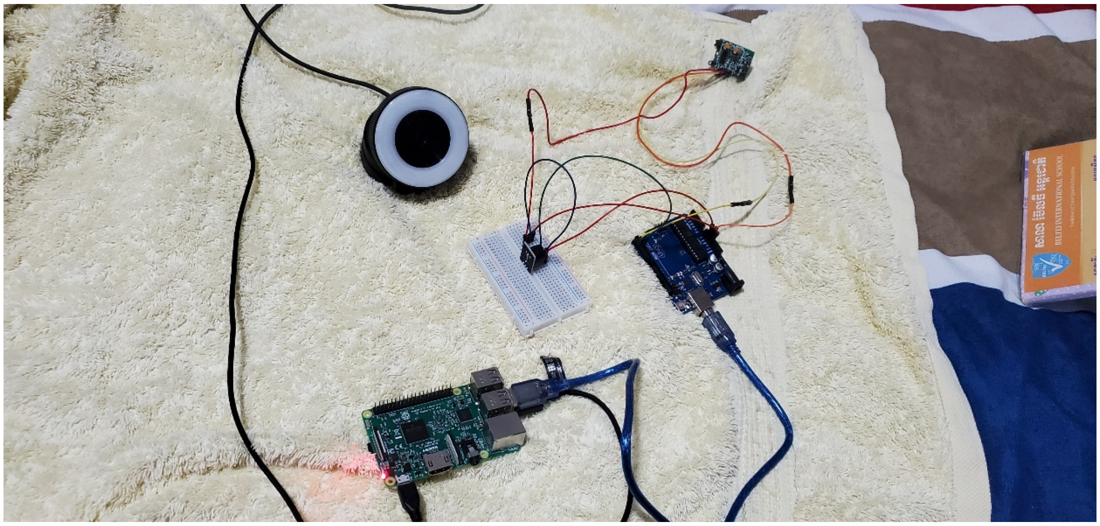
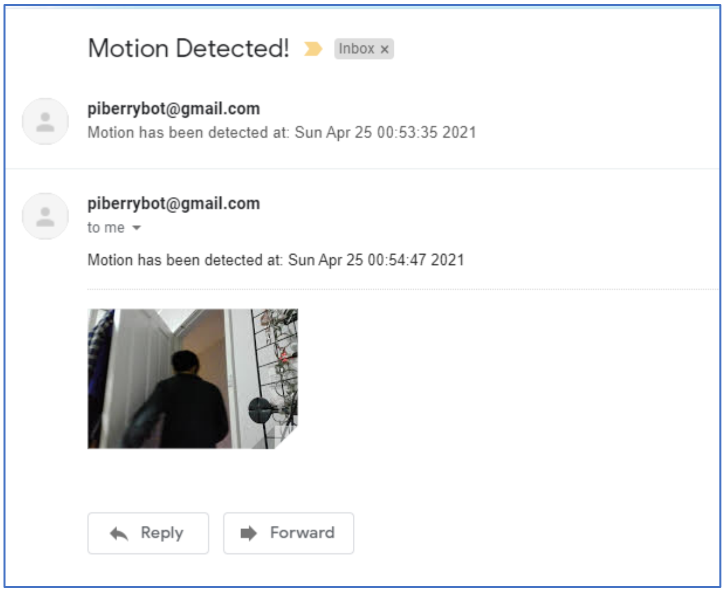


Fig 6. A final hardware Setup including the buzzer to be controlled by a website.



Motion Detected!

piberrybot@gmail.com

Motion has been detected at: Sun Apr 25 00:53:35 2021

piberrybot@gmail.com

to me

Motion has been detected at: Sun Apr 25 00:54:47 2021

Fig 7. The email being sent from raspberry PI when a motion is detected

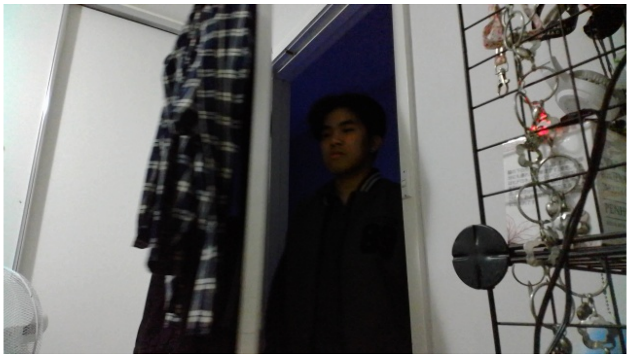
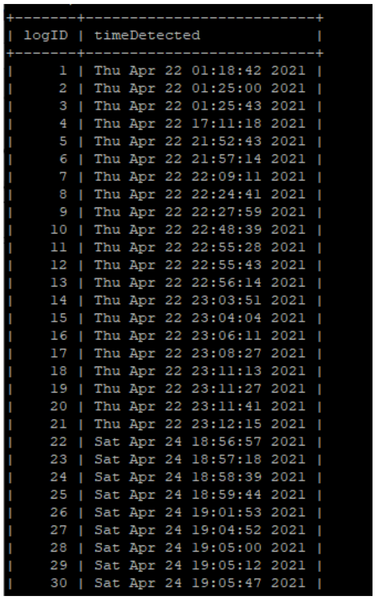
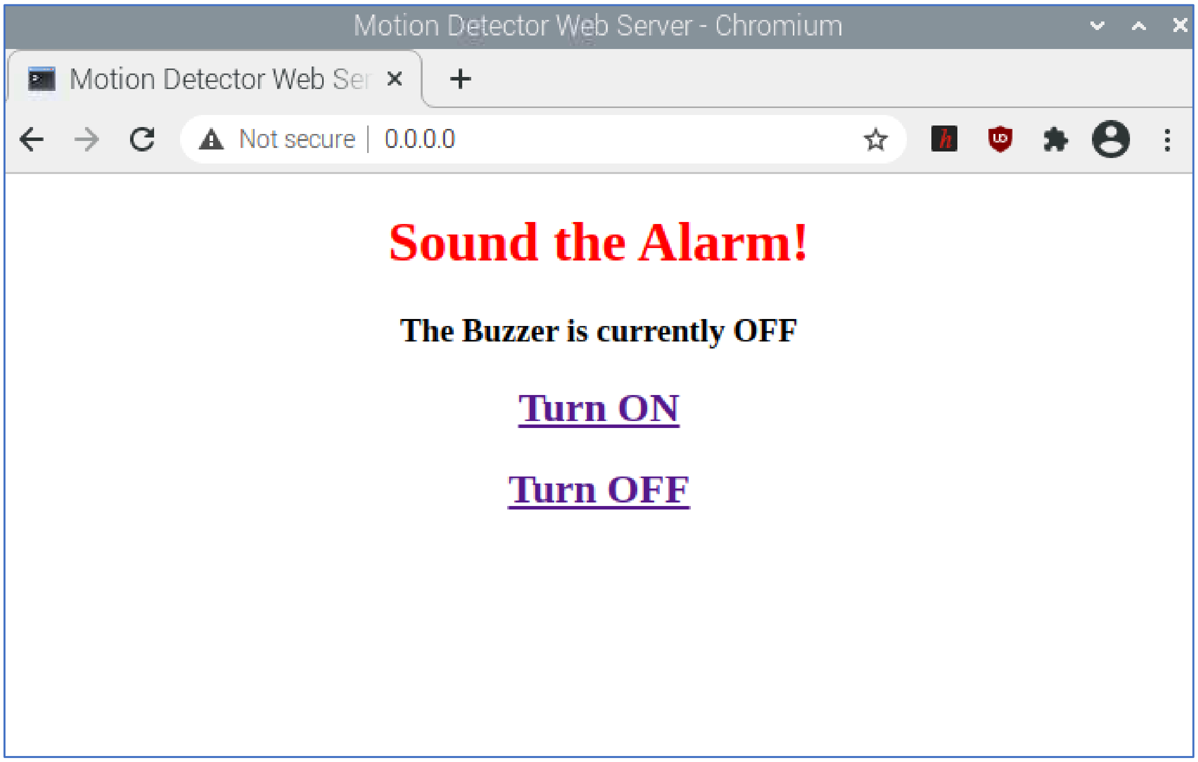


Fig 8. Some photos that are captured by the camera



```
+--------+--------------------------+
| logID  | timeDetected             |
+--------+--------------------------+
|      1 | Thu Apr 22 01:18:42 2021 |
|      2 | Thu Apr 22 01:25:00 2021 |
|      3 | Thu Apr 22 01:25:43 2021 |
|      4 | Thu Apr 22 17:11:18 2021 |
|      5 | Thu Apr 22 21:52:43 2021 |
|      6 | Thu Apr 22 21:57:14 2021 |
|      7 | Thu Apr 22 22:09:11 2021 |
|      8 | Thu Apr 22 22:24:41 2021 |
|      9 | Thu Apr 22 22:27:59 2021 |
|     10 | Thu Apr 22 22:48:39 2021 |
|     11 | Thu Apr 22 22:55:28 2021 |
|     12 | Thu Apr 22 22:55:43 2021 |
|     13 | Thu Apr 22 22:56:14 2021 |
|     14 | Thu Apr 22 23:03:51 2021 |
|     15 | Thu Apr 22 23:04:04 2021 |
|     16 | Thu Apr 22 23:06:11 2021 |
|     17 | Thu Apr 22 23:08:27 2021 |
|     18 | Thu Apr 22 23:11:13 2021 |
|     19 | Thu Apr 22 23:11:27 2021 |
|     20 | Thu Apr 22 23:11:41 2021 |
|     21 | Thu Apr 22 23:12:15 2021 |
|     22 | Sat Apr 24 18:56:57 2021 |
|     23 | Sat Apr 24 18:57:18 2021 |
|     24 | Sat Apr 24 18:58:39 2021 |
|     25 | Sat Apr 24 18:59:44 2021 |
|     26 | Sat Apr 24 19:01:53 2021 |
|     27 | Sat Apr 24 19:04:52 2021 |
|     28 | Sat Apr 24 19:05:00 2021 |
|     29 | Sat Apr 24 19:05:12 2021 |
|     30 | Sat Apr 24 19:05:47 2021 |
```

Fig 9. Times of detection being stored in the database.



Sound the Alarm!

The Buzzer is currently OFF

Fig 10. Layout of the website that controls the buzzer

Resources

- Lecture Slides
- https://stackoverflow.com/
- https://www.w3schools.com/python/python_classes.asp
- https://randomnerdtutorials.com/arduino-with-pir-motion-sensor/
- https://www.raspberrypi.org/forums/viewtopic.php?t=121724
- https://bc-robotics.com/tutorials/sending-email-attached-photo-using-python-raspberry-pi/
- https://subscription.packtpub.com/book/application_development/9781785283932/3/ch03lvl1sec28/accessing-the-webcam
- https://subscription.packtpub.com/book/hardware_and_creative/9781785285066/7/ch07lvl1sec41/working-with-webcam-using-opencv
- https://automaticaddison.com/2-way-communication-between-raspberry-pi-and-arduino/
- https://flask.palletsprojects.com/en/1.1.x/quickstart/
- https://www.arduino.cc/reference/en/language/functions/communication/serial/read/
- https://www.freeconvert.com/video-compressor
- https://clipchamp.com/en/video-editor/