MKSSS's
Cummins
COLLEGE OF ENGINEERING
FOR WOMEN

Maharshi Karve Stree Shikshan Sanstha's
Cummins College of Engineering for Women, Karve Nagar,
Pune - 411052
Department of Computer Engineering
Semester I
IoT Mini Project

"Exam Proctoring System "

Internet of Things Laboratory
CE4103 Class: Final Year

C22019221466 Vaishnavi Sinha 4465

Faculty Name: Prof. Ashwini Rajurkar

IOT Mini Project

Mini Project title : Examination Proctoring System

Introduction to the Concept :

A proctored exam is a timed exam taken while the student is monitored via a webcam using proctoring software. The video will be streamed live to the supervisor (Proctor). After the exam, audio recorded during the assessment test will be sent to the test proctor's email address for further analysis and may serve as evidence of breaches of academic integrity. Here this IOT based supervising application is implemented on Raspberry Pi.

Need of Concept :

Also known as online proctoring or remote proctoring, proctored exams use software and services to monitor and protect the academic integrity of students who take online exams. Prevent and deter violation of academic integrity. Assist and support students during exams.

Online education has developed rapidly in recent years. More and more students are taking advantage of Massive Open Online Courses (MOOCS) and other online certification courses. Universities are also moving online, providing more resources to students. The number of people offering courses is also growing rapidly. All of this gives students more opportunities to learn and improve.

System Requirements for the project Implementation :

- **Hardware Requirements :**

1. Two devices(eg. two Raspberry Pi 4b, and other combinations that can compile python)
2. Camera
3. Microphone

- **Software Requirements :**

1. Raspbian Operating System
2. Thonny Python IDE
3. Libraries:

a. opencv
b. tkinter
c. sounddevice
d. soundfile
e. smtplib
f. socket

Design Methodology :

1. Purpose and Requirements :

Purpose: The primary purpose of this system is to monitor candidates and prevent malpractices while testing by streaming live video of students and sharing recorded audio directly to the proctor.

Behaviour: The system only has automatic modes for sharing information via sockets and email. Many-to-many broadcast communication (for video).

System Management Requirement: The system must provide both live stream video and recorded audio to remote monitors and cloud storage.

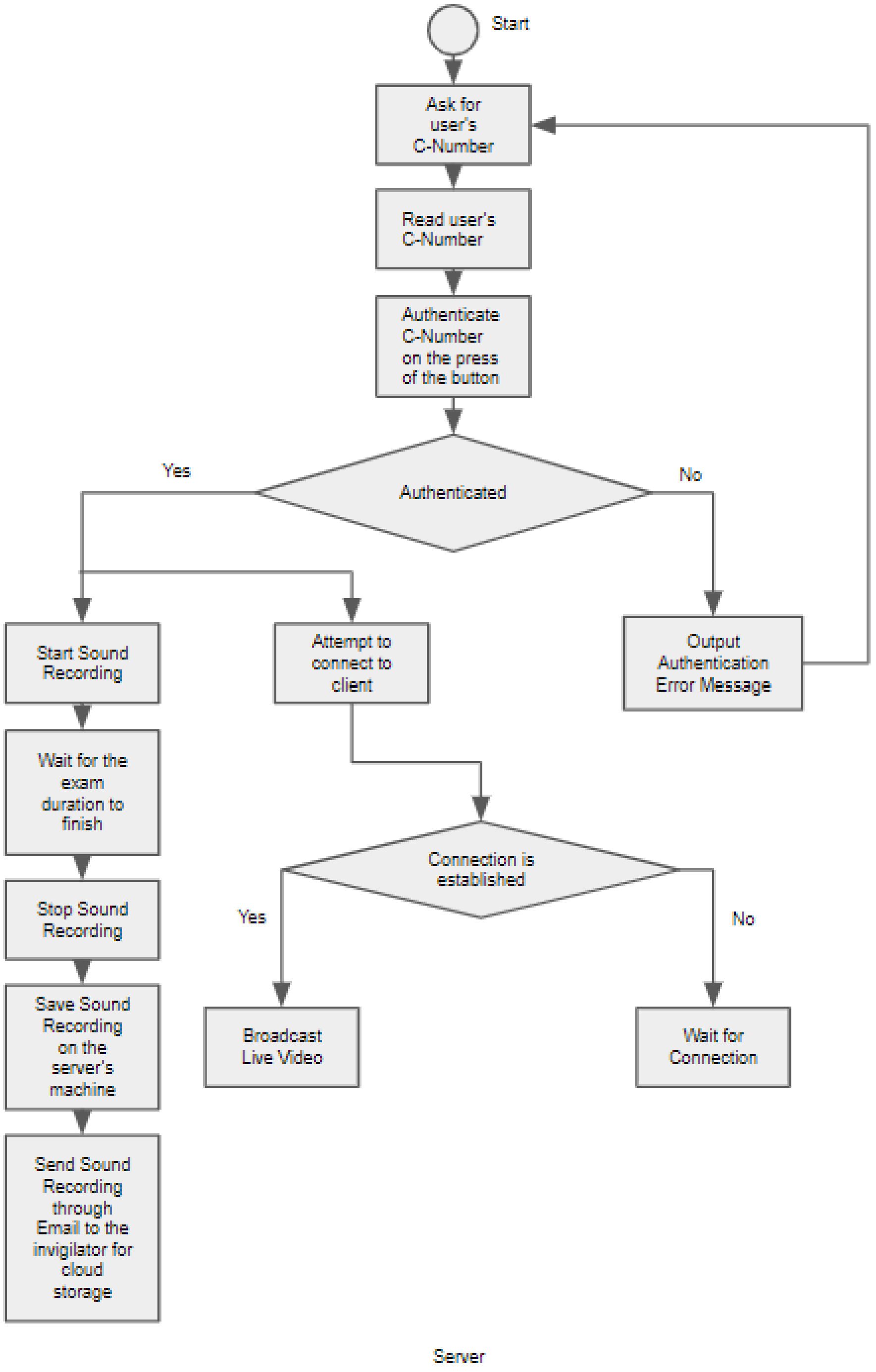
Application Deployment Requirements: Applications must be deployed locally on the device and accessed remotely on the device.

Data analysis requested: no analysis required.

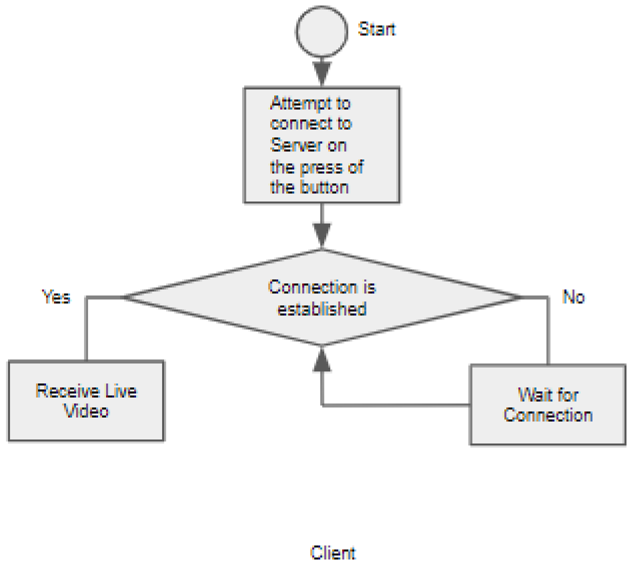
Security Requirement: The system must have basic user authentication capabilities.

2. Process Specification :

Process diagram of the test proctoring system. The system has two connection points for the client side (supervisor/supervisor) and the server side (student).

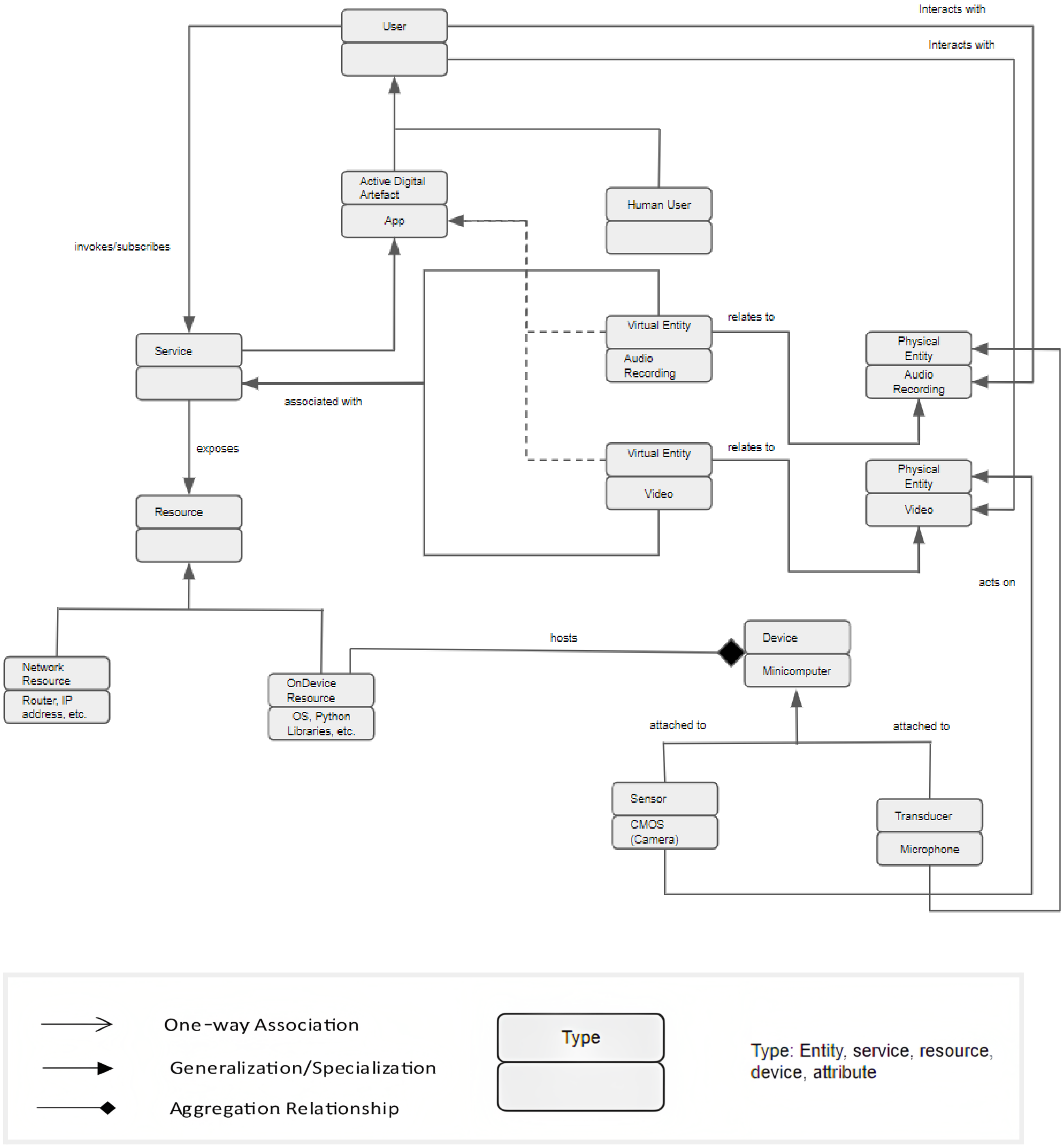

Start
Ask for user's C-Number
Read user's C-Number
Authenticate C-Number on the press of the button
Authenticated
Yes
No
Start Sound Recording
Wait for the exam duration to finish
Stop Sound Recording
Save Sound Recording on the server's machine
Send Sound Recording through Email to the invigilator for cloud storage
Attempt to connect to client
Output Authentication Error Message
Connection is established
Yes
No
Broadcast Live Video
Wait for Connection
Server

Program on Server's side is on the student's device


Start
Attempt to connect to Server on the press of the button
Connection is established
Yes
No
Receive Live Video
Wait for Connection
Client

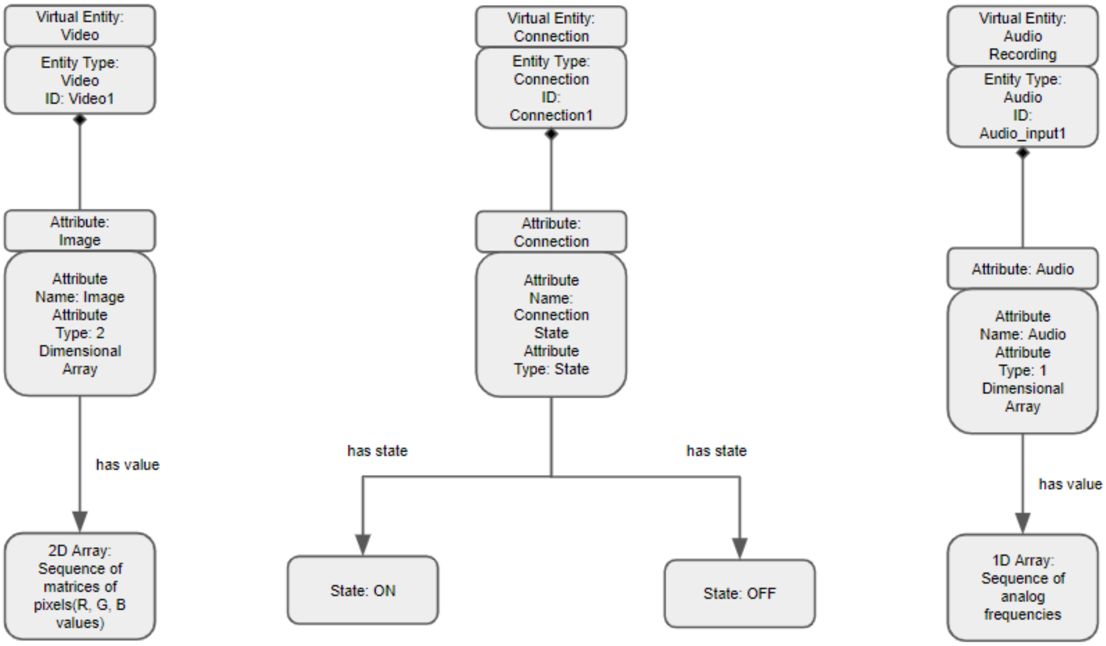
Program on Client’s side is on the proctor’s device

3. **Domain Model Specification :**
   The following diagram shows the domain model of the proctoring system. The domain model includes a device (such as a Raspberry Pi minicomputer, or two devices that can run the extension .exe), a camera, and a microphone. There are virtual entities for established connections as well as cameras and microphones. Resources are software components such as Thonny IDE (on-device resources) and smtplib (network resources) and so on. Services include a controller service that streams video live and sends notification emails containing student audio recordings to proctors.


Interacts with
Interacts with
User
Active Digital Artefact
App
Human User
invokes/subscribes
Service
associated with
exposes
Resource
Virtual Entity
Audio Recording
relates to
Virtual Entity
Video
relates to
Physical Entity
Audio Recording
Physical Entity
Video
acts on
acts on
Network Resource
Router, IP address, etc.
OnDevice Resource
OS, Python Libraries, etc.
hosts
Device
Minicomputer
attached to
attached to
Sensor
CMOS (Camera)
Transducer
Microphone
One-way Association
Generalization/Specialization
Aggregation Relationship
Type
Type: Entity, service, resource, device, attribute

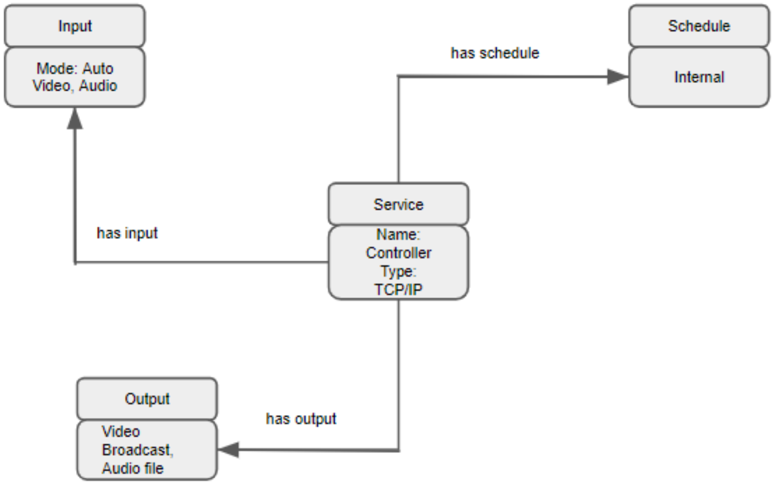
4. Information Model Specifications :

The proctoring system has virtual entities of recorded video, recorded audio, and connections made.


Virtual Entity: Video
Entity Type: Video ID: Video1
Attribute: Image
Attribute Name: Image Attribute Type: 2 Dimensional Array
has value
2D Array: Sequence of matrices of pixels(R, G, B values)
Virtual Entity: Connection
Entity Type: Connection ID: Connection1
Attribute: Connection
Attribute Name: Connection State Attribute Type: State
has state
has state
State: ON
State: OFF
Virtual Entity: Audio Recording
Entity Type: Audio ID: Audio_input1
Attribute: Audio
Attribute Name: Audio Attribute Type: 1 Dimensional Array
has value
1D Array: Sequence of analog frequencies

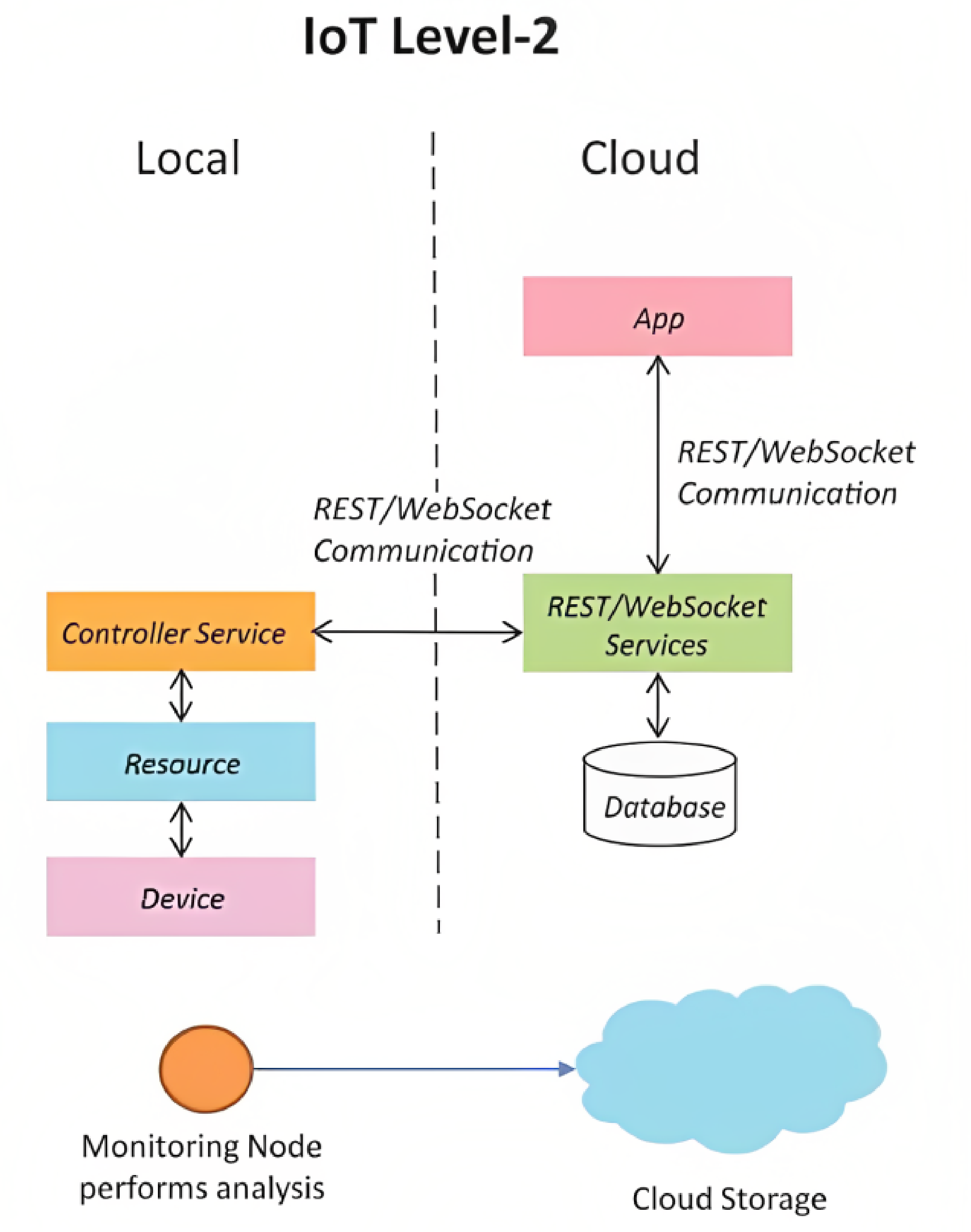
5. Service Specifications :

In auto mode, the controller service retrieves the recorded audio file, emails it to the exam leader after the exam is finished, and broadcasts live video on the server side.

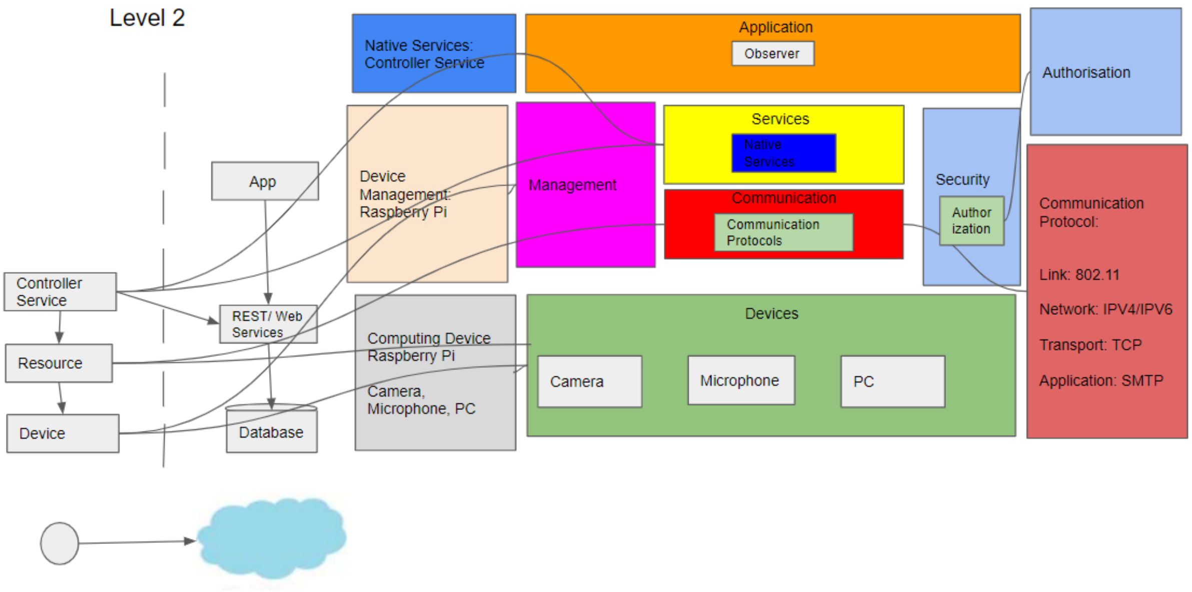


6. **IOT Level Specification :**

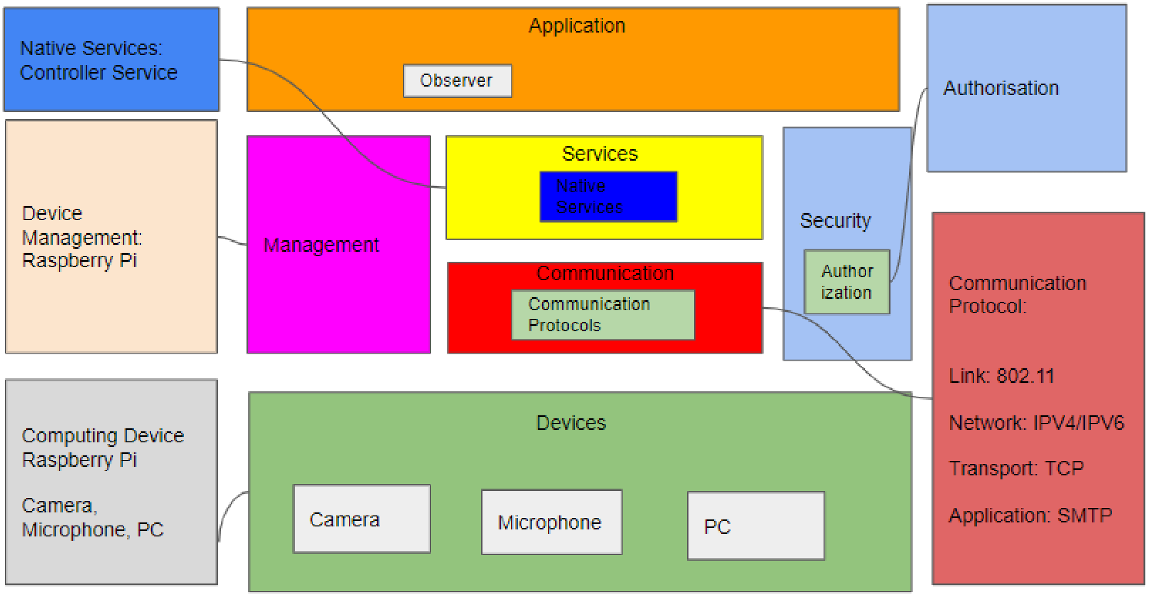
The IOT level of this system is level 2. This is because we have a node/device which is a microphone that records and stores audio data and sends the audio file to the Invigilator. A second device, a CMOS sensor camera, is used to capture a series of images and send them out a port when the connection between client and server is established. Both use the TCP/IP protocol. This system is IOT Level 2 as the audio files are stored on Google Drive (cloud).



7. Functional View Specification :

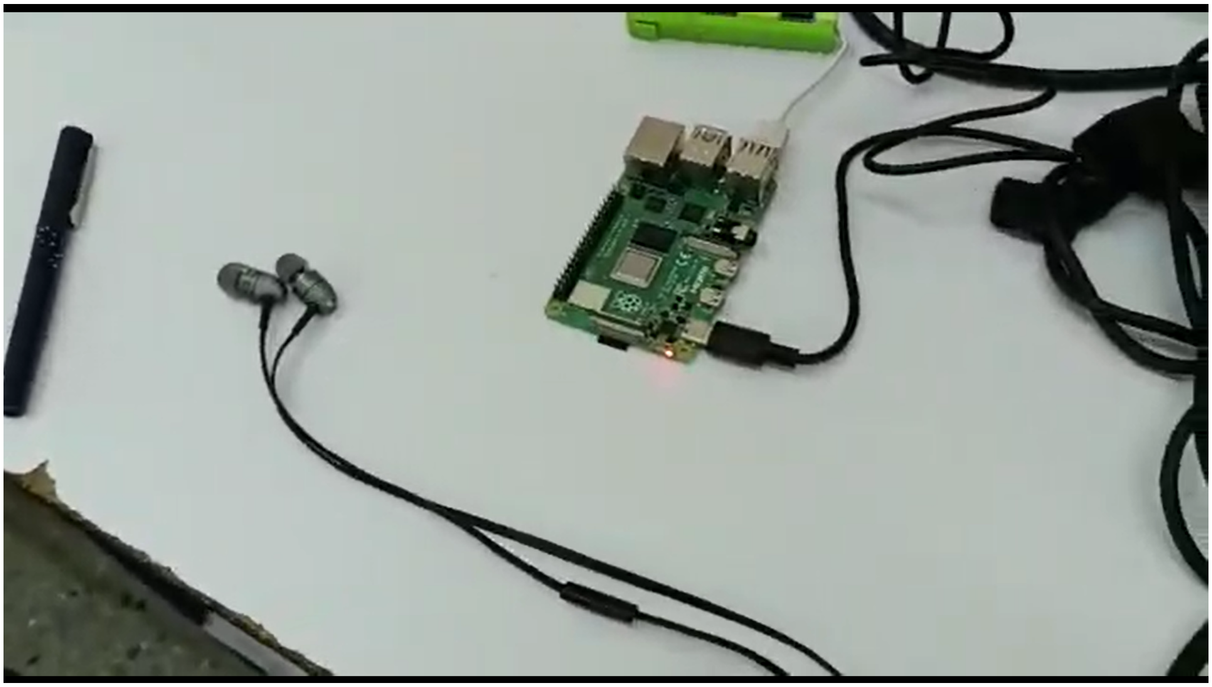


8. Operational View Specification :



9. Device and Component Integration

A schematic of a prototype test proctoring system using a Raspberry Pi.



Earphones used for microphone

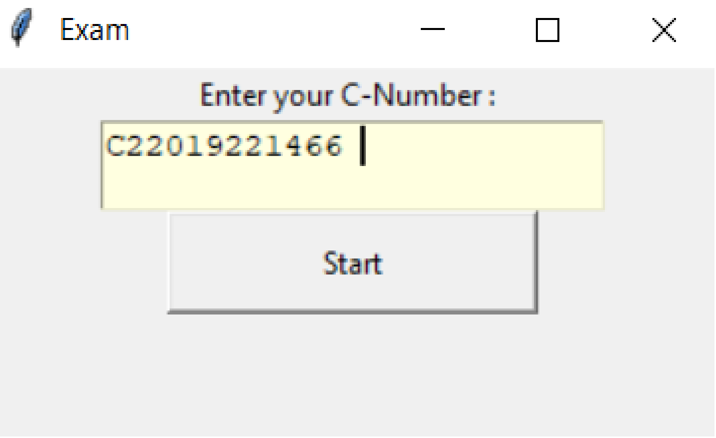
Webcam used(CMOS sensor)

10. Application Development

a. **Run the .exe file for the server**

fse.exe Application 57,266 KB

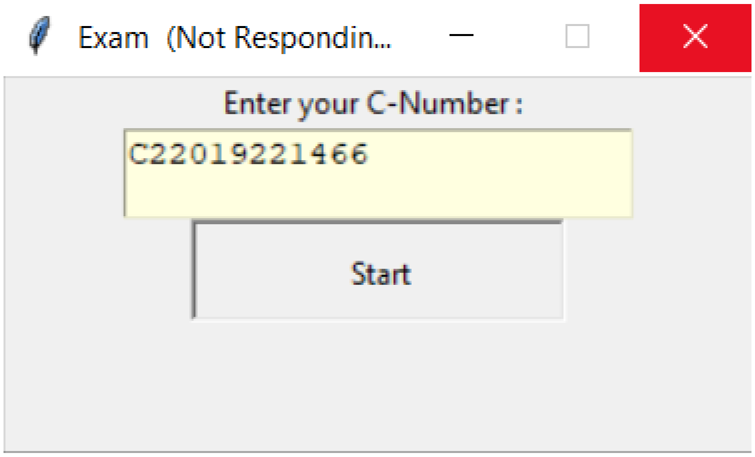
b. **You'll see a GUI**



Enter your C-Number :

C22019221466

c. **Enter your C-Number and click on the "Start" button**

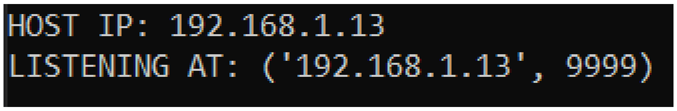


Enter your C-Number :

C22019221466

d. **It'll start listening at the assigned port to connect to a client, as well as recording the audio on the student's(server's) side**

Here 192.168.1.13 is the IP address of the student's(server's) device

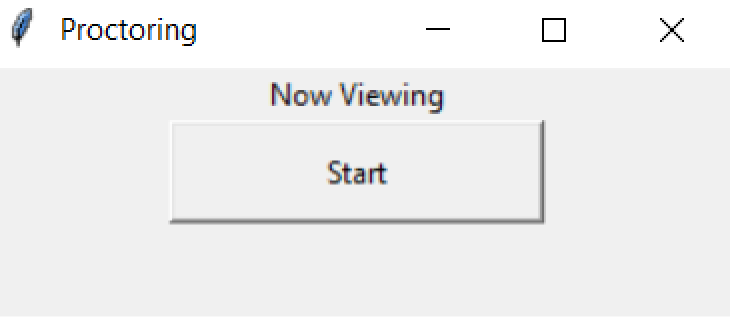


```
HOST IP: 192.168.1.13
LISTENING AT: ('192.168.1.13', 9999)
```

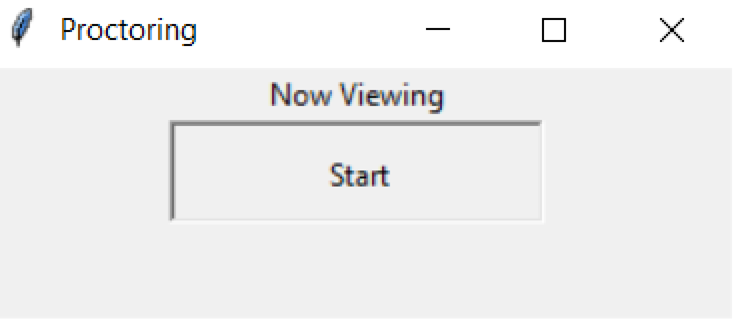
e. **Run the .exe file for the client on another device**

fcl.exe Application 56,007 KB

f. **You’ll see a GUI**

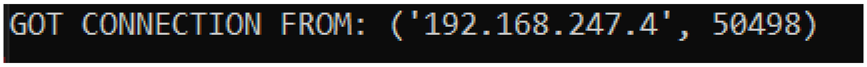


g. **Click on the “Start” button to connect to the server and start viewing**



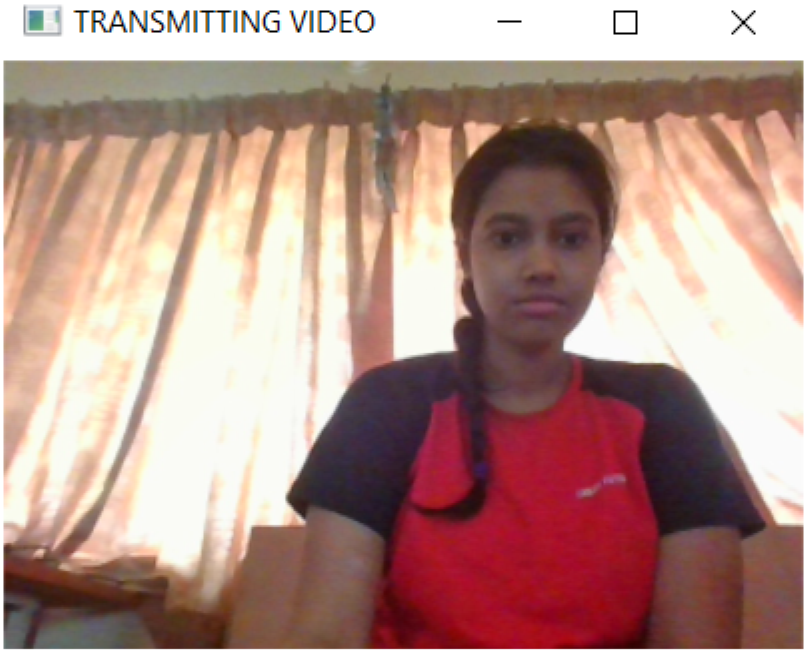
h. **When the client connects to the server, you’ll see the following which indicates successful connection of client and server**

Here 192.168.247.4 is the IP address of the examiner/proctor/invigilator/client’s device

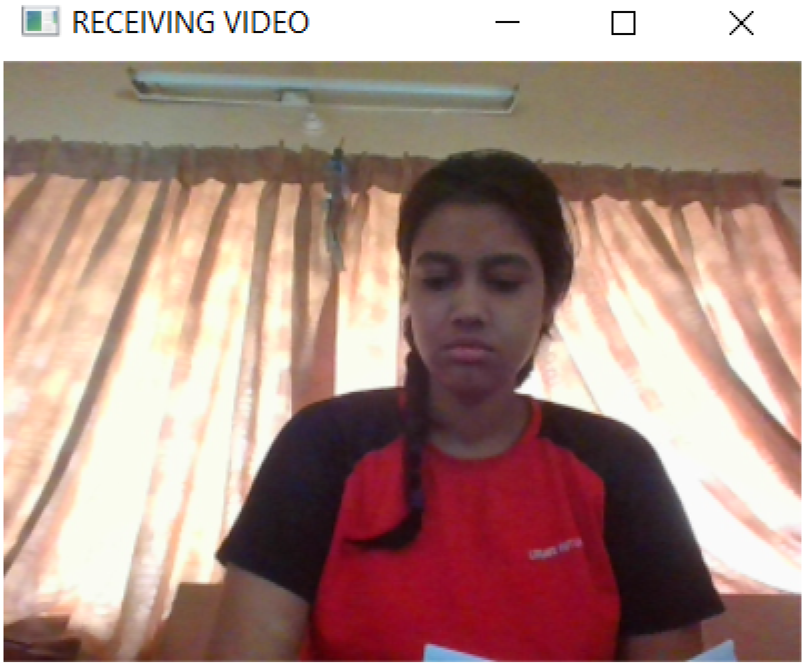


```
GOT CONNECTION FROM: ('192.168.247.4', 50498)
```

i. **On the server's side you'll be able to view the video that's being transmitted**

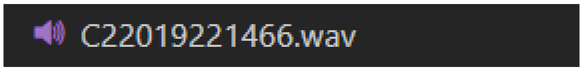


j. **On the client's side you'll be able to receive the live video of the student which will help prevent any violations to academic integrity**

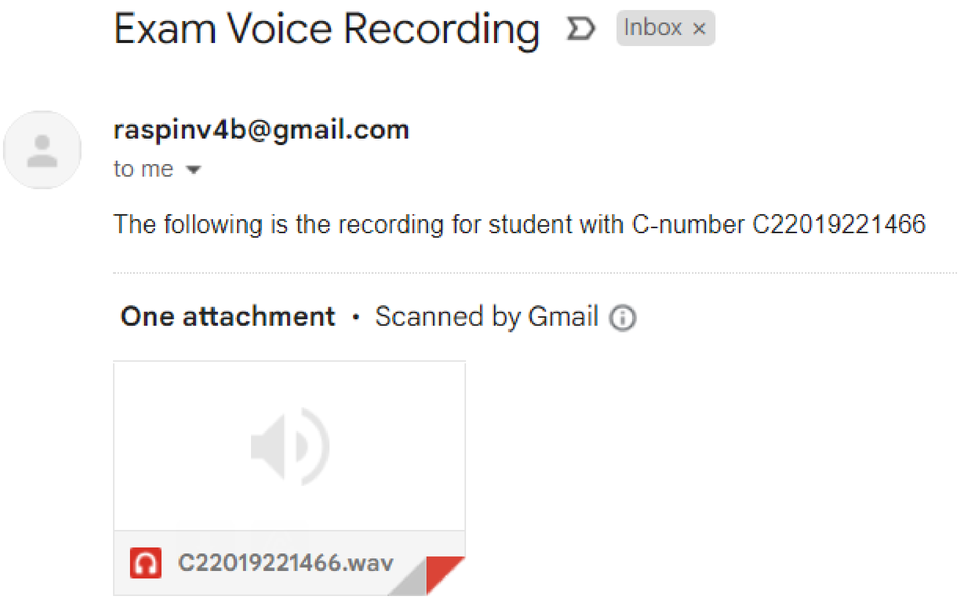


k. **After the examination duration is over**

   i. **The recorded audio will be saved on the student's device**


C22019221466.wav

ii. **Then sent to the invigilator's email address which is now stored on their drive with the following format**



Exam Voice Recording

raspinv4b@gmail.com

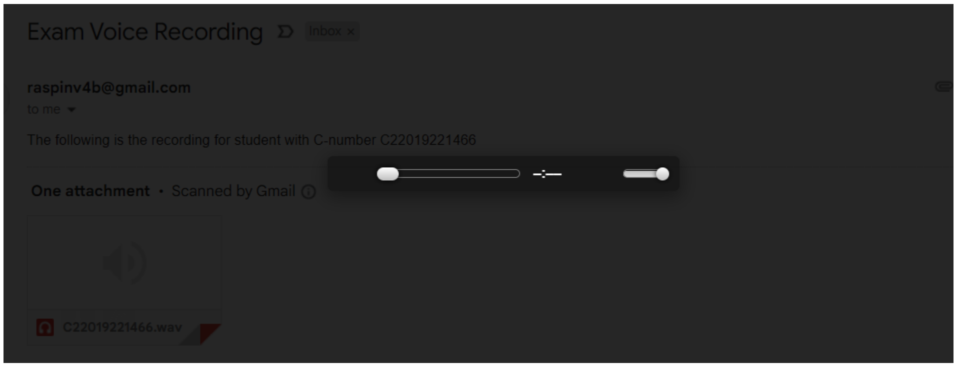
to me

The following is the recording for student with C-number C22019221466

One attachment • Scanned by Gmail

C22019221466.wav

l. **Which can then be played on GMail itself**



Exam Voice Recording

raspinv4b@gmail.com

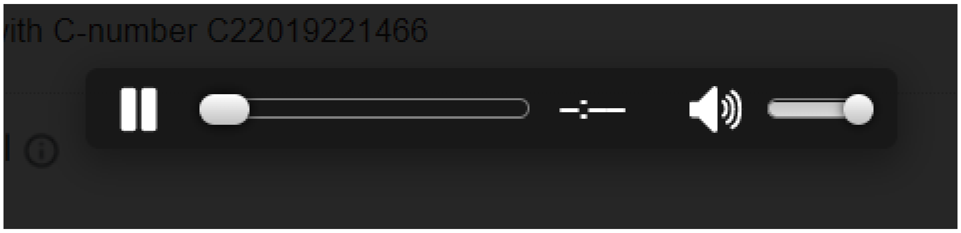
to me

The following is the recording for student with C-number C22019221466

One attachment • Scanned by Gmail

C22019221466.wav

or it can be downloaded to play

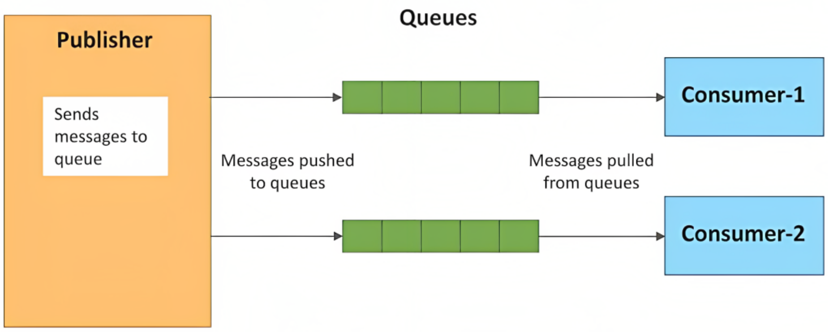


ith C-number C22019221466

which can act as evidence in case of academic dishonesty

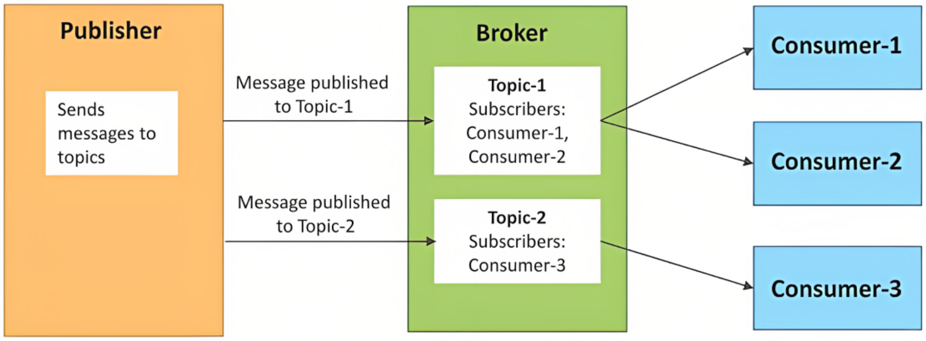
Communication Models :

- **Communication Model : Push-Pull(Live Video Broadcasting)**
  Push-pull is a communication model where data producers push data into queues and consumers pull data from queues. Producers do not need to be conscious of consumers.
  queues help separate messaging between producers and consumers. The queue also acts as a buffer. This is useful when there is a discrepancy between the speed at which producers push data and the speed at which consumers retrieve data.



- **Communication Model : Publish-Subscribe(Audio file sharing through mail)**

  Publish-subscribe is a communication model involving publishers, brokers and consumers.
  Publisher is a data source. Publishers send data to topics managed by brokers. Publishers are unaware of consumers.
  consumers subscribe to topics managed by brokers.
  When the broker receives data for a topic from a publisher, it broadcasts the data to all subscribing consumers.


Publisher
Sends messages to topics
Message published to Topic-1
Message published to Topic-2
Broker
Topic-1 Subscribers: Consumer-1, Consumer-2
Topic-2 Subscribers: Consumer-3
Consumer-1
Consumer-2
Consumer-3

Conclusion :

Online testing is the next wave of adoption following online learning, which has seen a significant increase in demand as a result of the ongoing COVID-19 Pandemic. Although this many-to-many exam proctoring system has its own set of advantages, it does not claim to be completely foolproof. To summarise, it is unclear whether the advantages of these Online Proctoring technologies outweigh the risks. The most reasonable conclusion we can reach at the moment is that the ethical justification of these technologies and their various capabilities necessitates us rigorously ensuring that a balance is struck between the concerns with the potential benefits to the best of our abilities.

References :

https://ip-webcam.appspot.com/static/doc.html

https://docs.python.org/3/library/socket.html

https://docs.python.org/3/howto/sockets.html

https://realpython.com/python-sockets/

https://python-sounddevice.readthedocs.io/en/0.4.5/

https://pysoundfile.readthedocs.io/en/latest/

https://pyinstaller.org/en/stable/

https://www.techtarget.com/searchnetworking/definition/port-number

https://www.geeksforgeeks.org/difference-between-socket-and-port/

https://pypi.org/project/vidstream/

https://github.com/scivision/PyLivestream