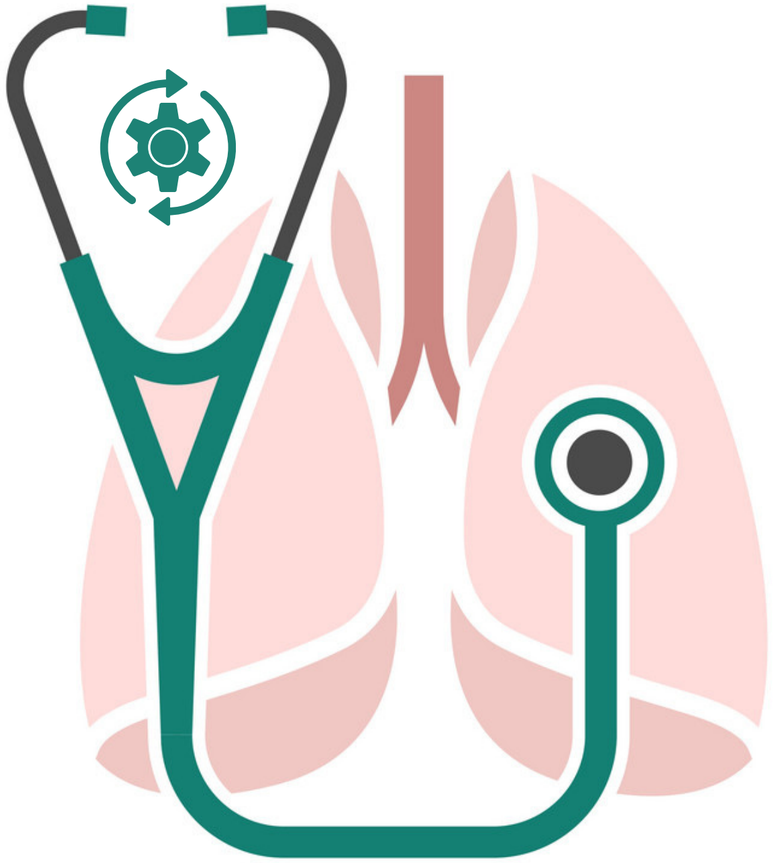


LUNG SOUND RECORDER

The Quad Chips

Team 38

Sai Praneeth - 2022101097

Jahnavi - 2022101118

Vaishnavi Priya - 2022101108

Sai Divya - 2022101090

Associated Professor - Abhishek Srivatsava

Associated TA - Santhoshini Thota

MICROPHONE ISSUES

Given that, our microphone doesn't work again, i.e, it is not feasible to solder the current one (due to being too small). Also considering Inputs from previous batches who had this project, we had to select a new microphone.

Also works for Raspi

- If you are short of GPIO pins then the MX9814 may be a good option
  - POSITIVE - it only needs one GPIO pin
  - POSITIVE - it has a built-in Automatic Gain Control (AGB).
  - NEGATIVE - it's quite noisy
  - NEGATIVE - you will need a good clean power supply
- If you can spare 3 GPIO pins then the INMP441 should be your choice
  - POSITIVE - much less noisy than the analog boards
  - POSITIVE - no need to power supply filter, handles the noisy power line without any issues
  - POSITIVE - very compact and small
  - NEGATIVE - no Automatic Gain Control (AGB)
  - NEGATIVE - no longer in production - replaced by the ICS4343 which doesn't have a large supply of break out boards at the moment.

Resource here: https://www.atomic14.com/2020/09/12/esp32-audio-input.html

VERIFICATION OF TEMP. SENSOR

Used temp. sensor and regular thermometer to get the readings from various subjects.

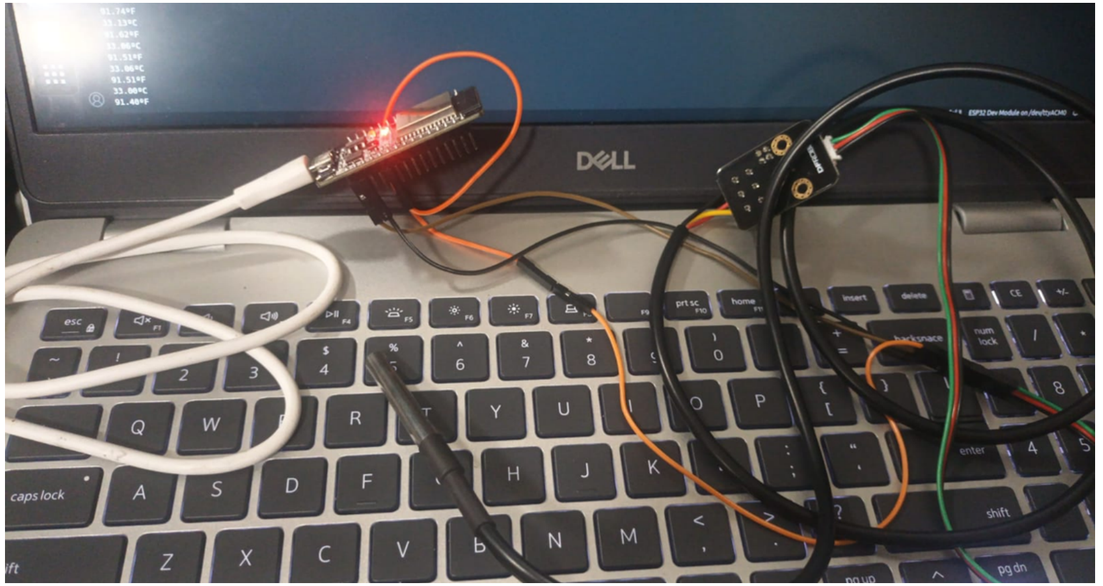
Reading from digital thermometer	Measured reading	%Error
96.5	95.5	1.03
95.4	94.2	1.27
96.9	95.7	1.23
95.9	94.1	1.87

Error % formula

(Thermometer reading - Measured reading)*100 ÷ Thermometer reading


96.5

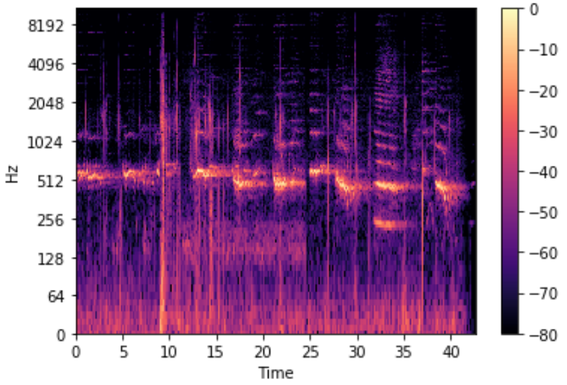

98.59


DELL

EXTRACTION OF THE BASIC FEATURE

To employ Machine learning techniques to extract the basic feature, we would need extensive datasets for each lung sound.

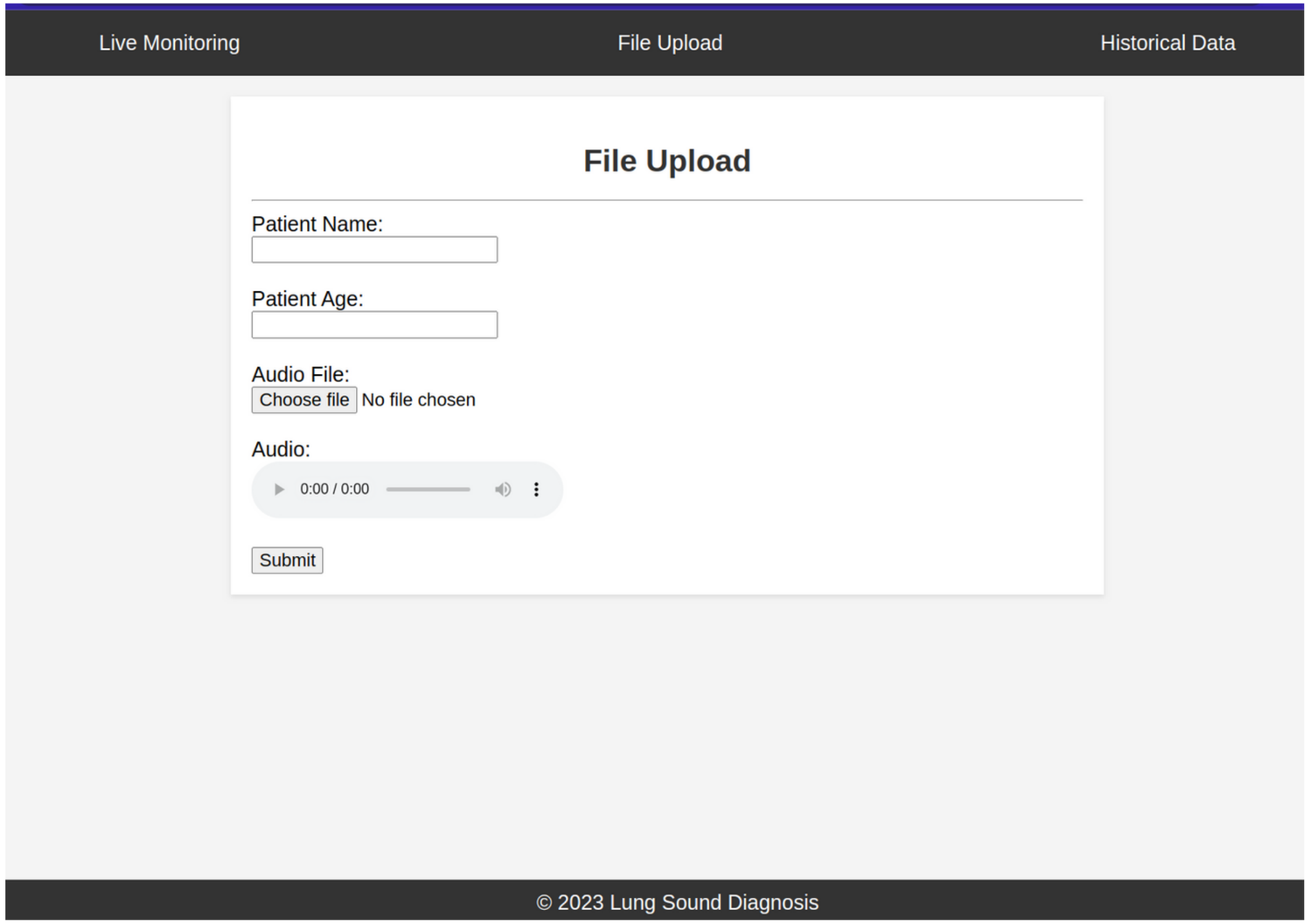
In the Signal Processing way, we would need to study and understand the basic features of each sound. Like frequency range, identifying sharp bands, ...



UI INTERFACE

We came up with a UI interface, for storing patients data
such as
1) Patients Name
2) Patients Age
3) Recorded Audio File (Lung Sounds)

Our UI takes Temperature data from ThingSpeak and the Lung sound from locally saved .wav files. Then, we run audio processing codes for the audio file and display the disease / class related to it.

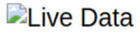


Here we will fill the patient details and upload the audio file

These details and audio file are saved in the local database

Live Monitoring File Upload Historical Data

Live Monitoring



Temperature

Temperature Reading: --°C

Here we are extracting temperature of the patient from thingspeak

Historical Data

- Patient Name: jo

  Patient Age: 333

  File Name: 4.wav

  0:00 / 0:00

- Patient Name: jo

  Patient Age: 43

  File Name: 4.wav

  0:00 / 0:00

- Patient Name: fhgjbn

  Patient Age: 45

  File Name: 4.wav

  0:00 / 0:00

- Patient Name: jo

Here we will display the data of the patient and the type of disease

FUTURE IMPLEMENTATIONS

We will further develop the user interface such that it will take the Lung sound and display the lung disease.

Coming up with better classification methods

Developing a front-end interface: Data visualization, admin controls - for setting the thresholds.

Real-time audio integration: Currently, audio is processed on some trigger. We will try to automate that by recording audios with some delay and processing it in between (alternating). And build an alerting system based on the real-time processed data.

Thank you!

Team 38, Group 4