



# MediSafe – Stay away and defeat diseases

2022 - 143

# Our Team



Supervisor

Mr. Ravi Supunya



Co- Supervisor

Mr. Samantha Rajapaksha



External Supervisor

Dr. Shashika  
Liyanage



IT18077698

Thennakoon  
T.M.B.C.K



IT19015422

Perera B.A.A.W.S



IT19015040

Rasuni Wageesha  
H.A



IT19011608

Senanayaka  
S.A.M.A.B.M

# Introduction

- ❑ What is MediSafe ?
- ❑ Why are we doing this research?
- ❑ What is the purpose of our research?



# Research problem

- There are some diseases that have arisen at present. (Heart attack, Pneumonia, Wheezing, Dengue, Covid'19)
- High cost for diagnosis.
- Informal lifestyle and busyness.
- Don't have enough idea about current situation of the country.

## ➤ Background

Table 22. Leading Causes of Hospital Deaths, 2010 - 2019

Disease and ICD (10 <sup>th</sup> Revision) Code	2019		2018		2017		2016		2015		2014		2013		2012		2011 <sup>a</sup>		2010 <sup>a</sup>	
	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%	Rank	%
Ischaemic heart disease I20 - I25	1	15.1	1	15.0	1	14.2	1	14.1	1	14.2	1	14.8	1	14.7	1	14.4	1	13.4	1	12.8
Zoonotic and other bacterial diseases A20 - A49	2	12.1	3	10.9	2	11.5	3	11.6	3	9.7	3	9.1	6	7.9	6	7.1	6	6.7	6	6.6
Neoplasms <sup>1</sup> C00 - D48	3	11.7	2	11.7	3	10.5	2	12.0	2	11.0	2	11.7	2	11.3	2	11.6	2	11.8	2	11.1
Diseases of the respiratory system excluding diseases of upper respiratory tract pneumonia I20 - I22, J40 - J98	4	10.7	4	9.9	4	9.8	5	8.3	4	9.2	6	8.0	5	7.9	5	7.2	5	6.9	5	7.0
Pneumonia J12 - J18	5	8.0	7	7.8	6	8.2	7	6.4	7	7.5	7	6.6	8	6.1	8	5.7	9	5.2	9	5.2
Pulmonary heart disease and diseases of the pulmonary circulation I26 - I51	6	7.6	6	7.9	5	8.5	4	8.7	5	8.3	4	8.6	4	8.4	3	9.0	4	8.7	3	8.7
Cerebrovascular disease I60 - I69	7	7.8	5	8.0	7	7.7	6	8.2	6	8.2	5	8.4	3	8.6	4	8.7	3	8.7	4	8.7
Diseases of the urinary system N00 - N99	8	5.8	8	5.8	8	5.9	8	6.3	8	6.2	8	6.8	7	6.2	7	6.3	7	5.7	8	5.7
Diseases of the gastro-intestinal tract K00 - K92	9	5.0	9	5.1	9	5.1	9	5.5	9	5.3	9	5.7	9	5.7	9	5.4	8	5.4	7	6.2
Traumatic injuries S00 - T32, W54	10	3.8	10	3.9	10	3.8	10	3.9	10	3.8	10	3.5	11	3.3	11	3.7	11	3.8	11	3.7
Disease of the nervous system G00 - G98	11	1.3	13	1.4	14	1.4	14	1.7	13	1.6	14	1.4	15	1.4	16	1.5	19	1.4	18	1.6
Symptoms, signs and abnormal clinical and labo R00 - R99	12	1.3	11	1.5	12	1.5	12	1.6	13	2.3	11	3.2	10	4.8	10	4.5	10	4.1	10	5.0
Diabetes mellitus E10 - E14	13	1.3	12	1.4	11	1.7	11	1.8	13	1.6	13	1.6	13	1.6	14	1.7	14	1.9	16	1.7

<sup>a</sup> Includes deaths reported from the Cancer Hospital (not analysed by site and type of neoplasm)

<sup>1</sup> Excludes Malignant Diseases

Source : Medical Statistics Unit



Source : Medical Statistics Unit Ministry of Health  
[http://www.health.gov.lk/moh\\_final/english/public/elfinder/files/publications/AHB/AHS%202019.pdf](http://www.health.gov.lk/moh_final/english/public/elfinder/files/publications/AHB/AHS%202019.pdf)

# Overall solutions – 50%

- Developed an Arduino-based device that detects certain types of symptoms to diagnose certain heart and lung related diseases.
- Use some machine learning based techniques to identify diseases. and clarify it.
- Show diseases spread rate to the user.
- Developing a mobile application and web application to facilitate patient usage.



# Research Objectives



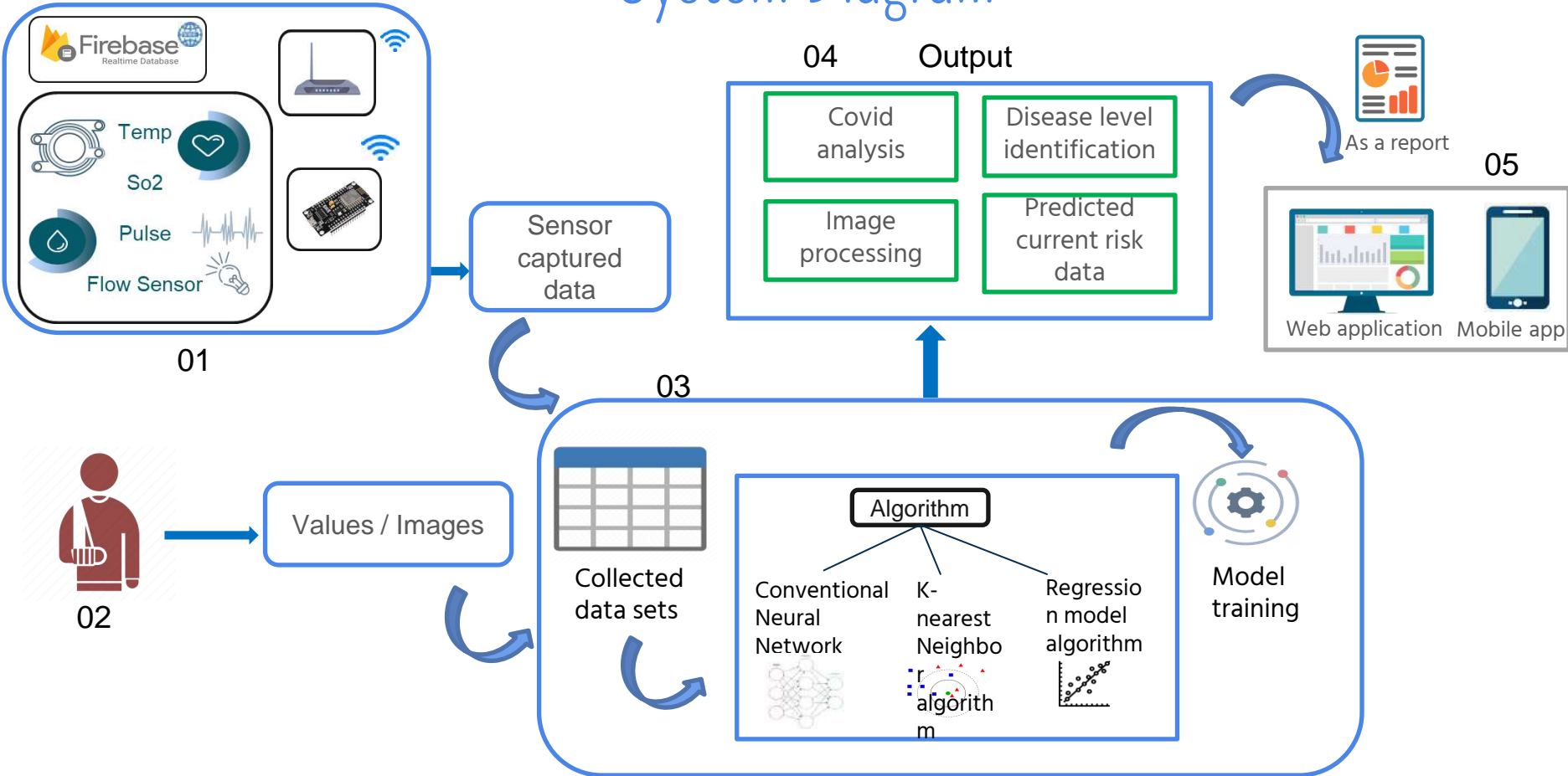
Implement a device to get parameters of the patient and identify Covid'19. (Possibility as a percentage)

Disease level wise identification and provide suggestions/ recommendations to reduce the risk level.

Identify the exact lung disease among other lung diseases.

Identify the three major diseases spread rate in Sri Lanka.

# System Diagram



# Focusing areas



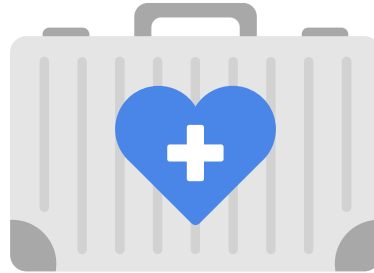
Hospital / Medical centers

Schools



Offices

Crowded areas



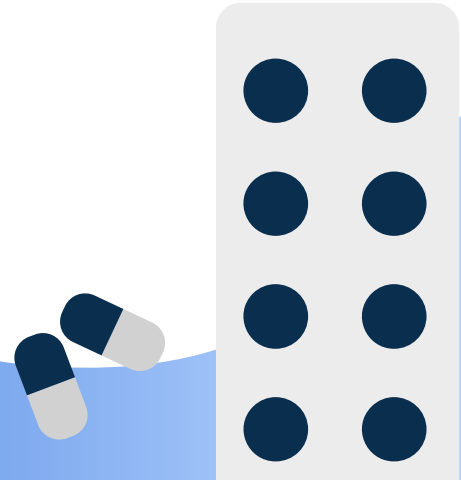




IT18077698

Thennakoon T.M.B.C.K

Specialization | Information Technology



# Table of Contents

01

Research question

02

Archived – 50%

03

System diagram

04

Progress

05

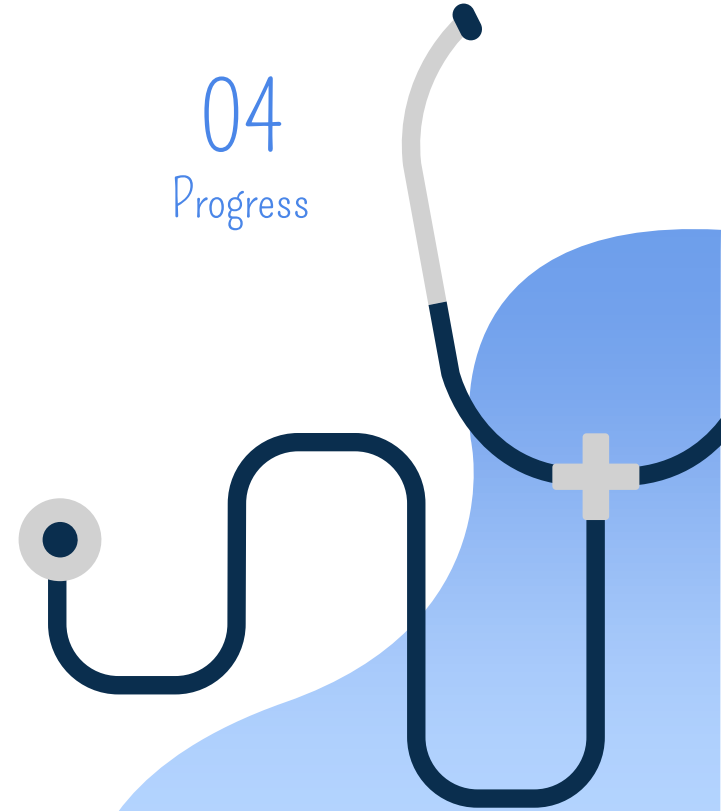
Technologies

06

Requirements

07

Risk mitigation



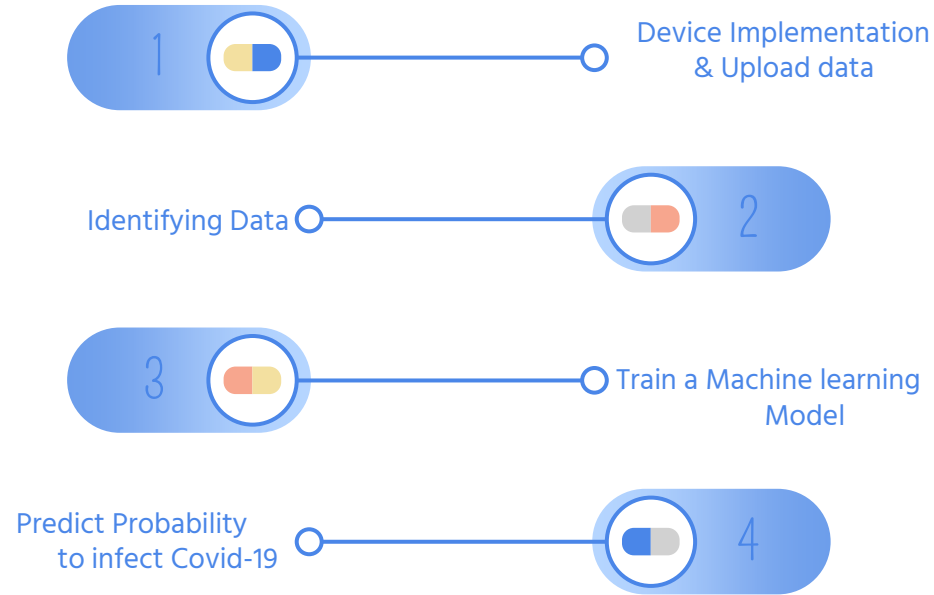
# Research question

- ❑ Identify all measurements using single device with few minutes.
- ❑ Simple and user-friendly web application and mobile application.

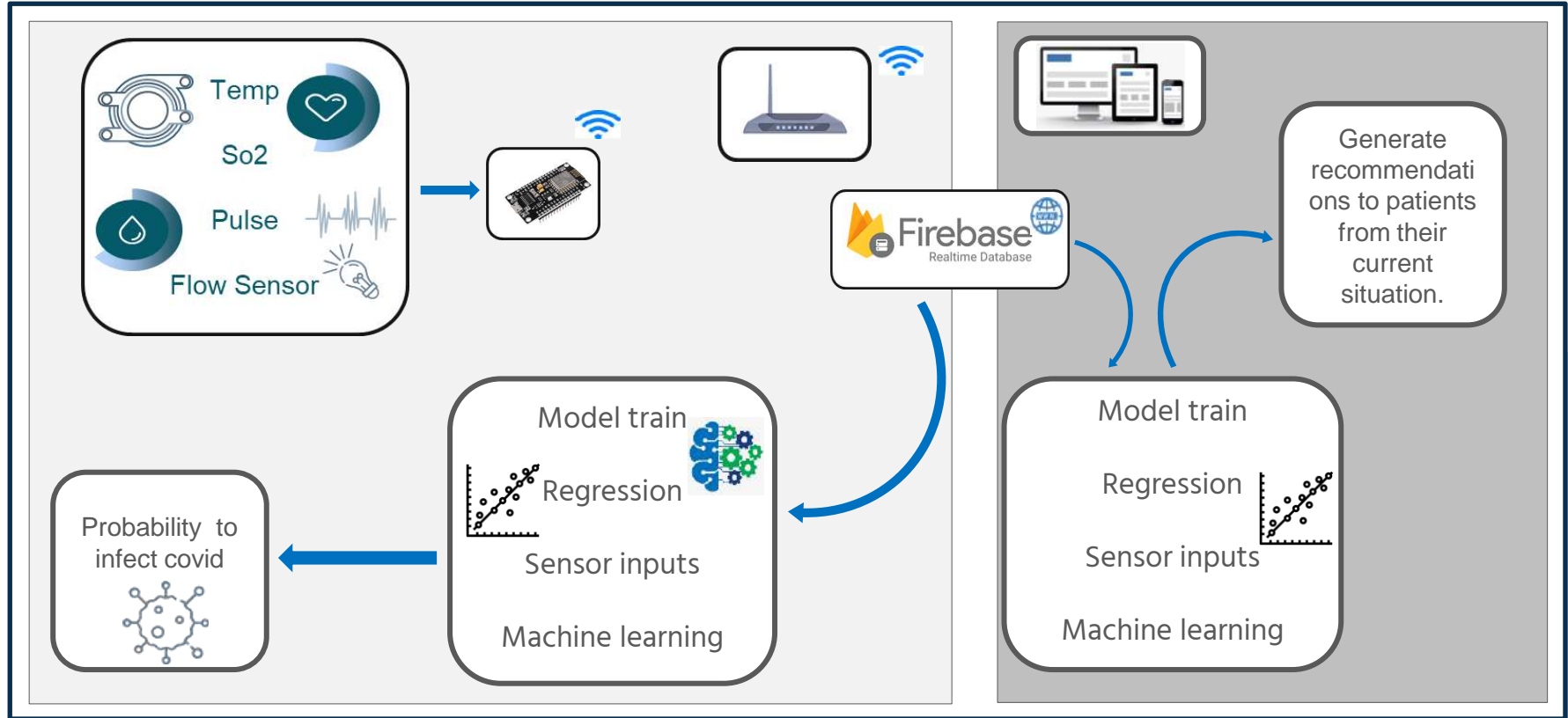


- ❑ Provide probability to infect Covid 19 & Give what are the necessary actions need to get by patient.
- ❑ Get necessary inputs and Generate healthy recommendations to day-to-day life.

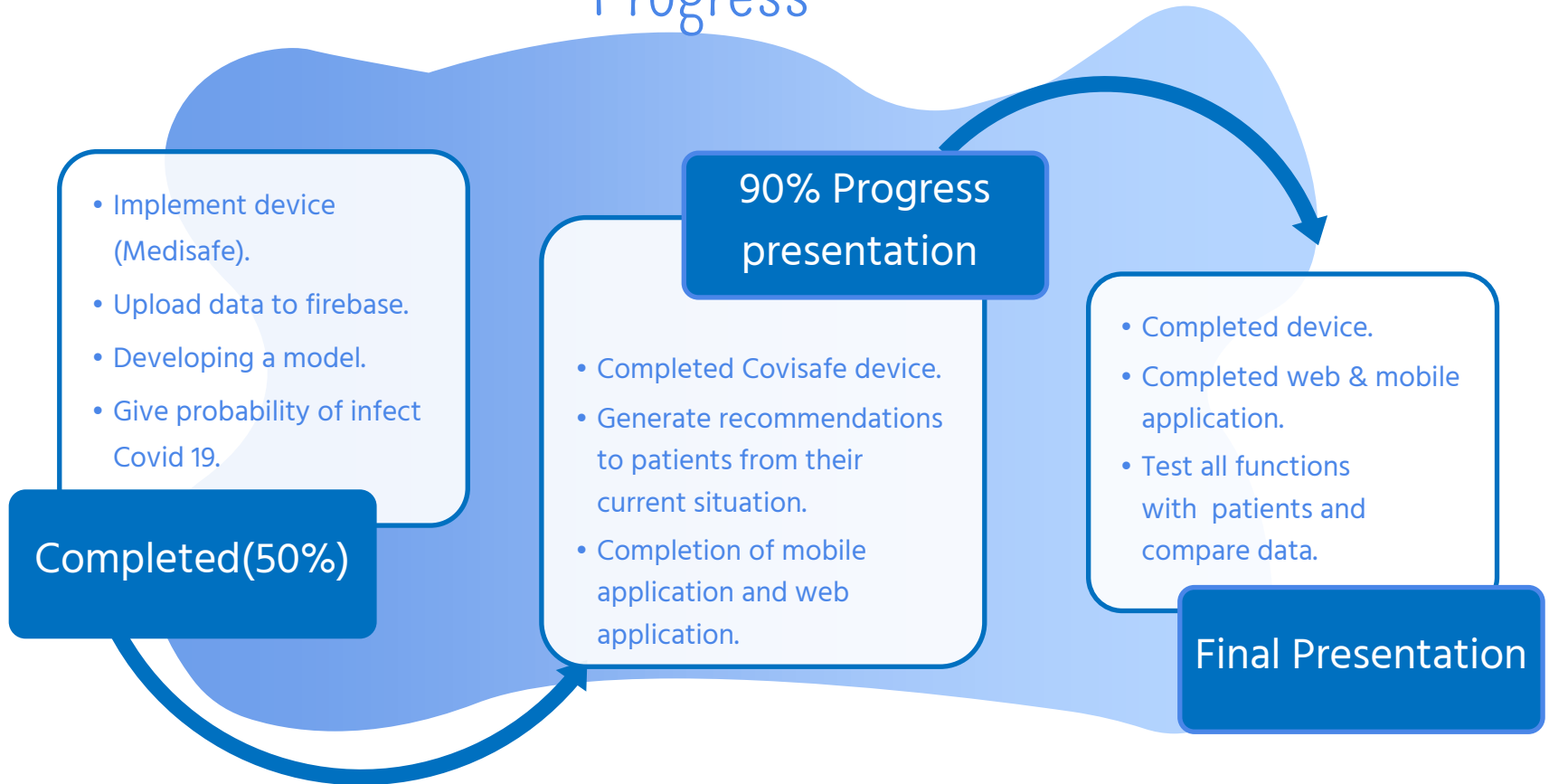
# Achieved – 50%



# System diagram



# Progress



# Latest technologies in MediSafe

## Medisafe device



ESP8266  
Max30105sens  
Flow meter  
Firebase

## Dataset collection and data training



Kaggle  
Regression  
Python

## IDE



Jupyter Notebook  
Arduino IDE  
Visual studio code  
Anaconda

# Requirements

## Functional

- Interoperability
- Accuracy
- Compliance

## Non – functional

- Maintainability
- Manageability
- Usability
- Integrity



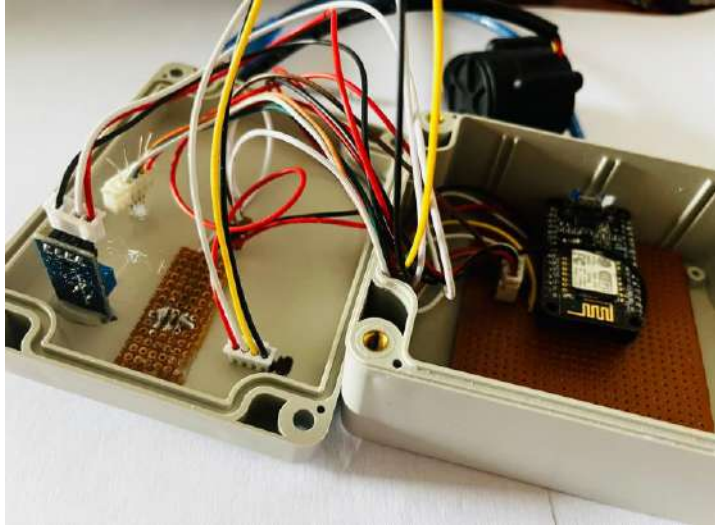


# Risk mitigation



- Checking the manual again with the doctor for data obtained from the tool and what is provided in the processed data output.
- Periodically check the accuracy of the data obtained by the device
- Identifying hazards and hazardous situations associated with a medical device.

# Completion of the project



Medisafe  
device

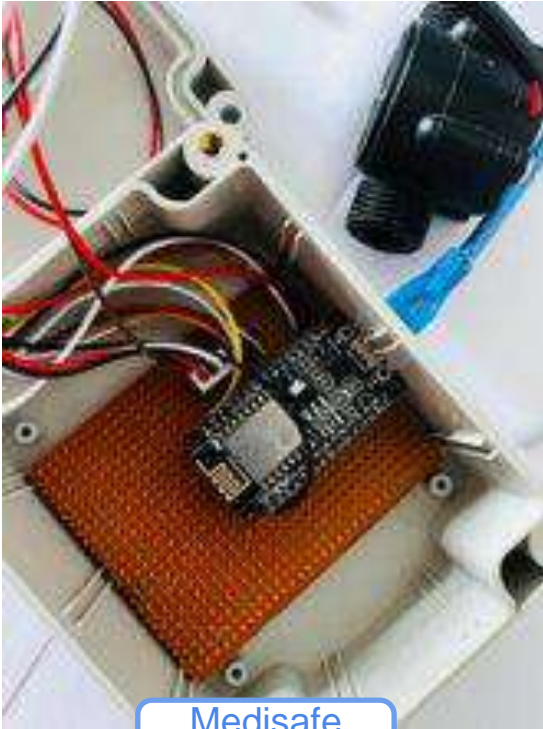
```
final2 | Arduino 1.8.19 (Windows Store 1.8.57.0)
File Edit Sketch Tools Help

final2$  Firebase  flow_meter  max30105  s02  temp

16 int count = 0;
17 int temp_cps;
18 //-----Firebase-----
19
20 #include <ArduinoJson.h>
21 #include "FirebaseESP8266.h"
22 #include <ESP8266WiFi.h>
23 // Set these to run example.
24 #define FIREBASE_HOST "medisafe-research-default-rtdb.firebaseio.com/unit_1"
25 #define FIREBASE_AUTH "qjnABtFp7TzCcENUpoxBQSe12lkghJolOPwzBB5"
26 #define WIFI_SSID "supun"
27 #define WIFI_PASSWORD "supun111191"
28 FirebaseData firebaseData;
29
30 #define SENSOR D4
31 long currentMillis = 0;
32 long previousMillis = 0;
33 int interval = 1000;
34 //boolean ledState = LOW;
35 float calibrationFactor = 4.5;
36 volatile byte pulseCount;
37 byte pulse1Sec = 0;
38 float flowRate;
39 unsigned int flowMillilitres;
40 unsigned long totalMillilitres;
41 // -----led
42
43 #define REDLED D5
44 // -----led
45
```

Arduino IDE

# Completion of the project



Medisafe  
device

```
C:\WINDOWS\system32\cmd.exe

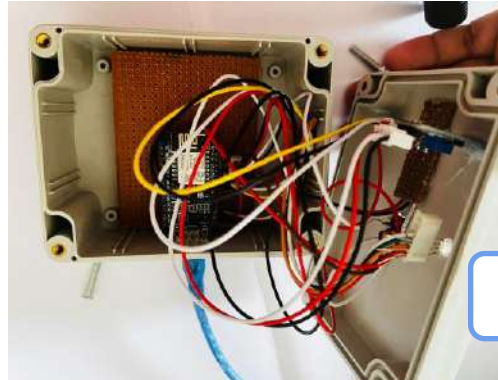
(covid) C:\Users\user>cd C:\Users\user\Desktop\24-04-2022\covid

(covid) C:\Users\user\Desktop\24-04-2022\covid>C:

(covid) C:\Users\user\Desktop\24-04-2022\covid>python Runcovid.py
type oxygen level : 90
type your pulse : 96
type your Temperature : 90
confidence : 100.0 %
The probability of having a covid infection is 35.360000000000004%
Traceback (most recent call last):
  File "Runcovid.py", line 1, in <module>
    from covid import predictc
ImportError: cannot import name 'predictc' from 'covid' (C:\Users\user\Desktop\24-04-2022\covid\covid.py)

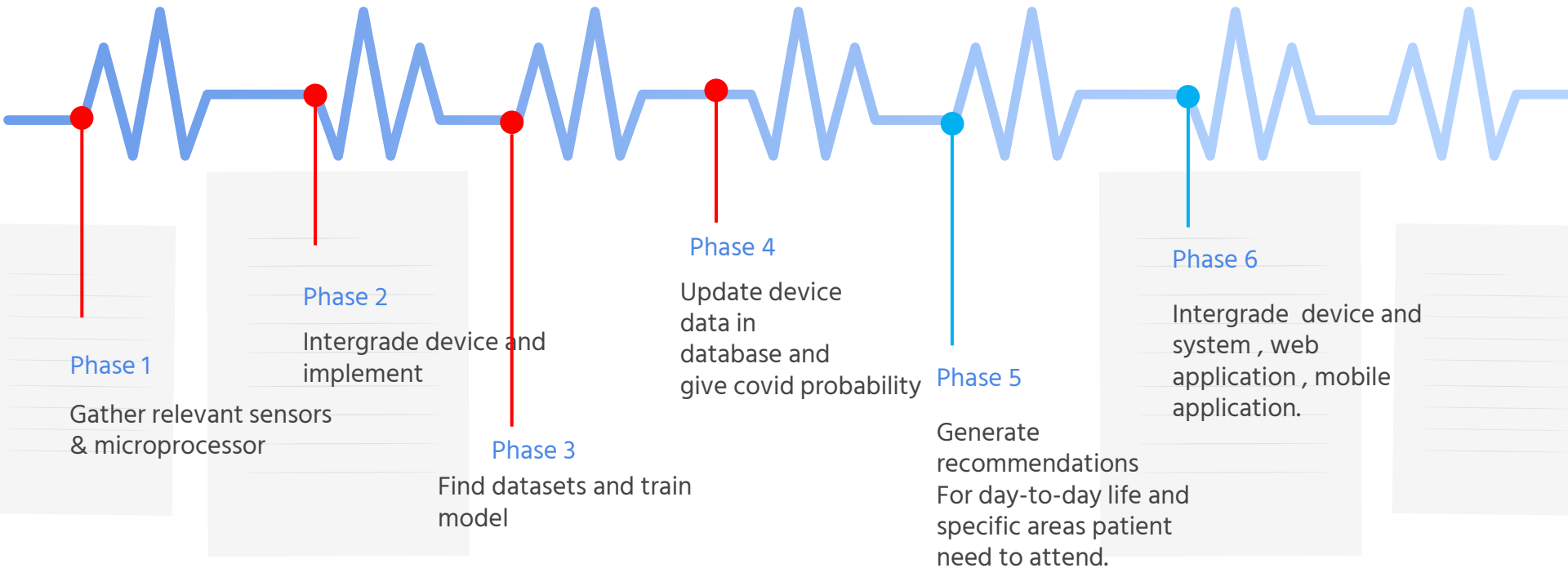
(covid) C:\Users\user\Desktop\24-04-2022\covid>
```

Covid prediction  
output



Medisafe  
device

# Self work breakdown structure



# Individual overleaf conference paper

The image shows a screenshot of the Overleaf LaTeX editor interface. The left pane displays the source code of a LaTeX document, and the right pane shows the rendered PDF output.

**Source Code (Left Pane):**

```
49 \and
50 \IEEEauthorblockN{6\textsuperscript{th} Senanayake S.A.M.B.M}
51 \IEEEauthorblockA{\textit{Faculty of computing, SLIIT (of Aff.)}} \
52 \textit{Sri Lanka Institute of Information Technology (of Aff.)}\
53 Sri Lanka \
54 it19011608@slit.my.lk}
55 }
56
57 \maketitle
58
59 \begin{abstract}
60 The world has altered for the worse as a result of the Covid 19 epidemic.
61 Many individuals used to avoid going to the hospital unless it was
62 absolutely necessary. During the epidemic, new concepts were produced
63 with the assistance of technology.
64 Many research on computerized pulse diagnosis have been conducted, based
65 on the background studies. Because of its non-invasive nature and ease of
66 use in determining people's health state, it is gaining popularity. To
67 get more precise pulse waves and contact pressure signals, many
68 acquisition devices have been suggested.
69 A gadget has been developed to identify the pulses of diabetes patients
70 from those of healthy people. The goal here is to normalize the signals
71 in normal circumstances before looking for differences in pulse
72 characteristics in fundamental pathological scenarios, such as diabetes
73 situations [4].
74 In addition, a remote patient diagnosis system based on IoT devices and
75 predictive machine learning models that operate on patient data on a
76 regular basis has been created [7].
77 Ayurvedic doctors are putting the system through its paces as a
78 computer-assisted diagnostic tool.
```

**Rendered PDF (Right Pane):**

**MEDISAFE - STAY AWAY AND DEFEAT DISEASE**

<b>1<sup>st</sup> N.H.P. Ravi Supunya Swarnakantha</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka ravis@slit.lk	<b>2<sup>nd</sup> U.U.Samantha Rajapaksha</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka samantha.r@slit.lk
<b>3<sup>rd</sup> T M B C K Thennakoon</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka it18077698@slit.my.lk	<b>4<sup>th</sup> Perera B.A.A.W.S</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka it19015422@slit.my.lk
<b>5<sup>th</sup> Rasuni Wagesha H.A</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka it19015040@slit.my.lk	<b>6<sup>th</sup> Senanayake S.A.M.B.M</b> <i>Faculty of computing, SLIIT (of Aff.)</i> <i>Sri Lanka Institute of Information Technology (of Aff.)</i> Sri Lanka it19011608@slit.my.lk

**Abstract:** The world has altered for the worse as a result of the Covid 19 epidemic. Many individuals used to avoid going to the hospital unless it was absolutely necessary. During the epidemic, new concepts were produced with the assistance of technology. Many research on computerized pulse diagnosis have been conducted, based on the background studies. Because of its non-invasive nature and ease of use in determining people's health state, it is gaining popularity. To get more precise pulse waves and contact pressure signals, many acquisition devices have been suggested. A gadget has been developed to identify the pulses of diabetes patients from those of healthy people. The goal here is to normalize the signals in normal circumstances before looking for differences in pulse characteristics in fundamental pathological scenarios, such as diabetes situations [4]. In addition, a remote patient diagnosis system based on IoT devices and predictive machine learning models that operate on patient data on a regular basis has been created [7]. Ayurvedic doctors are putting the system through its paces as a computer-assisted diagnostic tool.

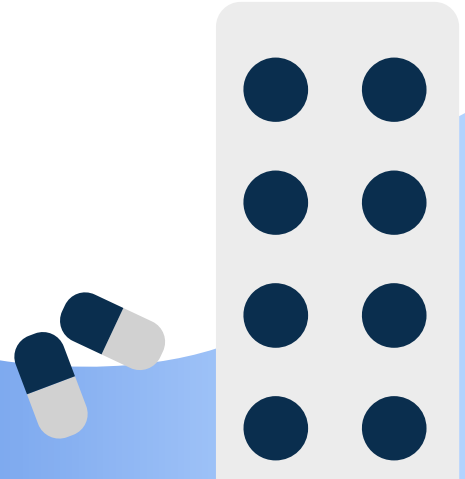
**Designed for vulnerable groups rather than COVID-19 patients,** and it depends on simple yes/no and categorical questions rather than NLU-based analysis of a patient's input, such as symptom descriptions. Following the provision of responses, the user experience was promptly assessed by requesting feedback. Another chatbot developed by University of California professors was established solely for the purpose of screening health-care employees and is not intended for general usage. [9]. The pre-trained Google BERT language model was employed in an early COVID-19 chatbot paradigm to address the COVID-19 question-answering task. The COVID-19 sample question-answers dataset was used to test the suggested chatbot. According to early results, the proposed technique delivered appropriate responses to the supplied question. More datasets will be added in the future to increase the model's accuracy and robustness [10]. In the health sector, the proposed IoT-based solution has the potential to save lives and deliver essential services. In the fight against infection, it might be a useful tool for medical professionals and law enforcement. This system can also give crucial healthcare and effective monitoring to infected and suspected individuals. Because physical distance may be maintained with the usage of the system while giving therapy, the risk of healthcare service providers being infected from treating any patient can be



IT19015422

Perera B.A.A.W.S

Specialization | Information Technology



# Table of Contents

01

Research question

02

Archived – 50%

03

System diagram

04

Progress

05

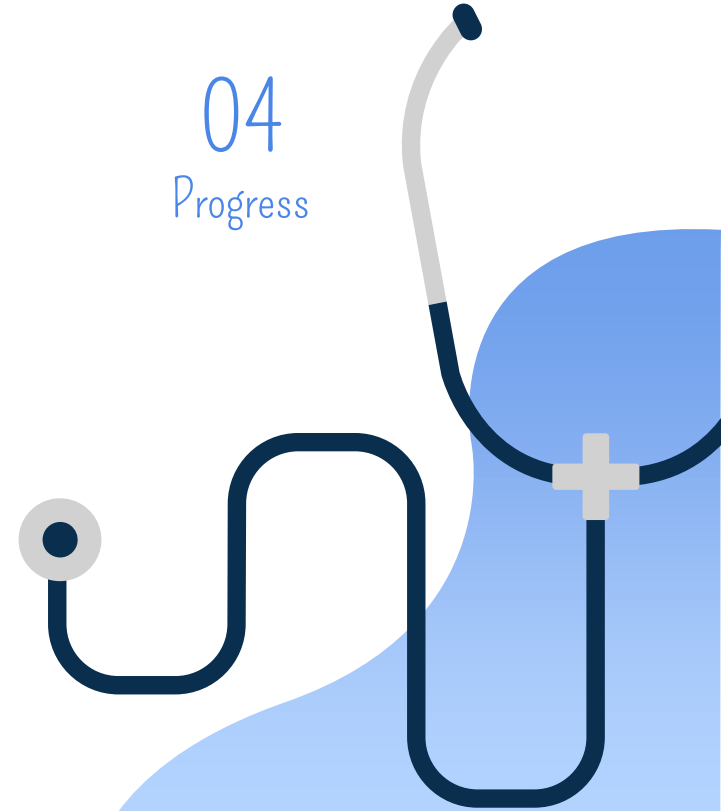
Technologies

06

Requirements

07

Risk mitigation



# Research question

- How to identify the people who are suffering in such lung and heart diseases (level wise)

- How to check current situation in cost effectively

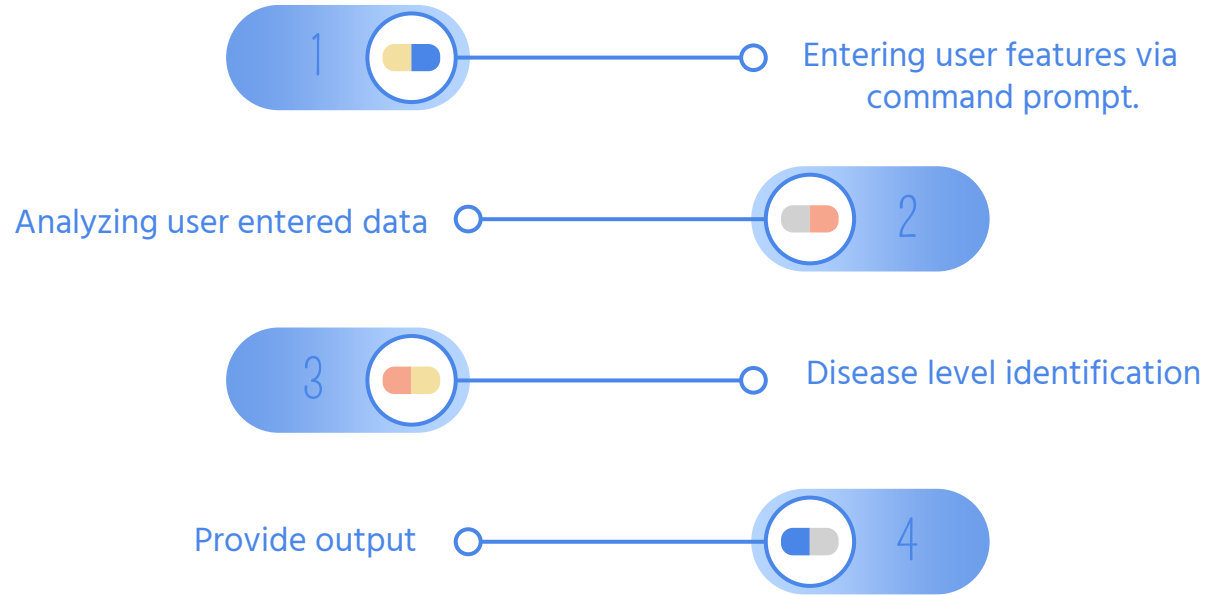
- How to provide easily recommendations via web/mobile application to the user

- What are the solutions we can give due to shortage of medicines

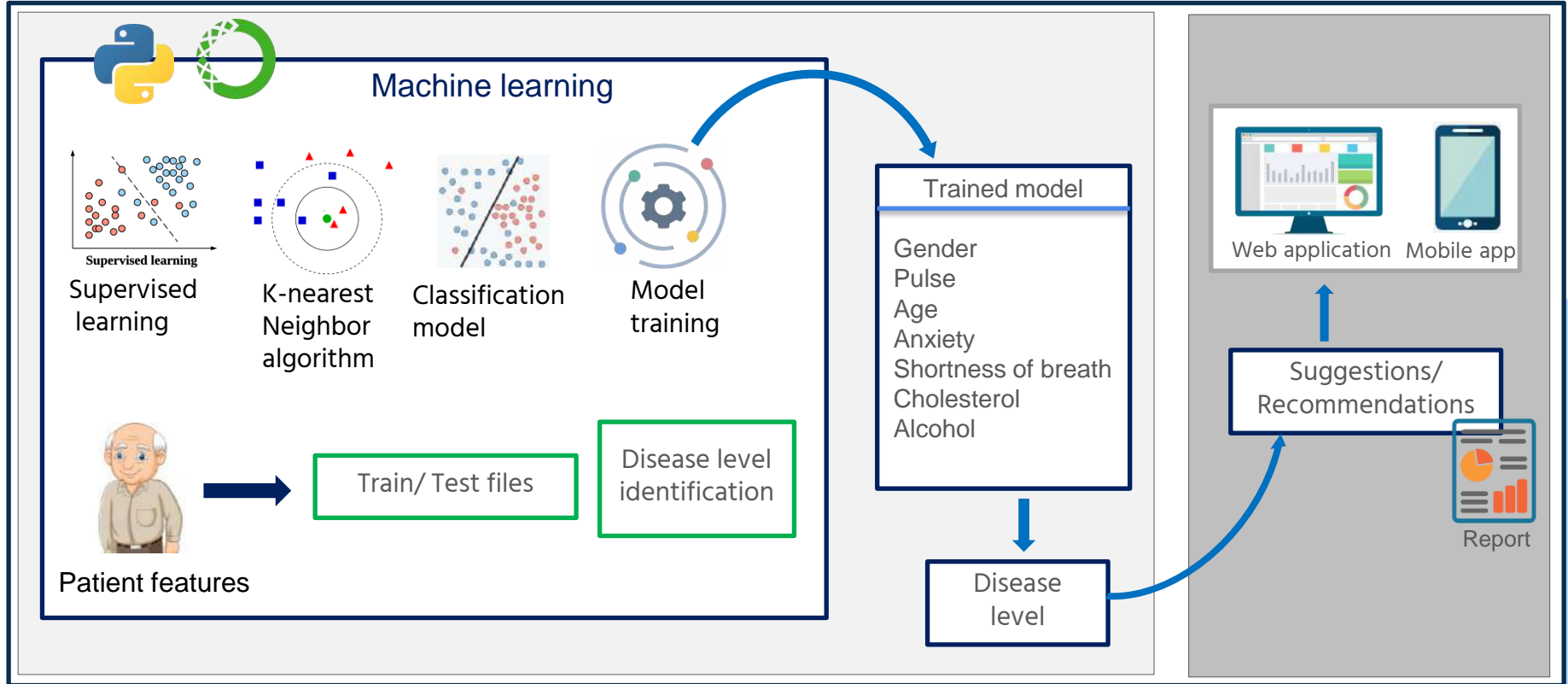




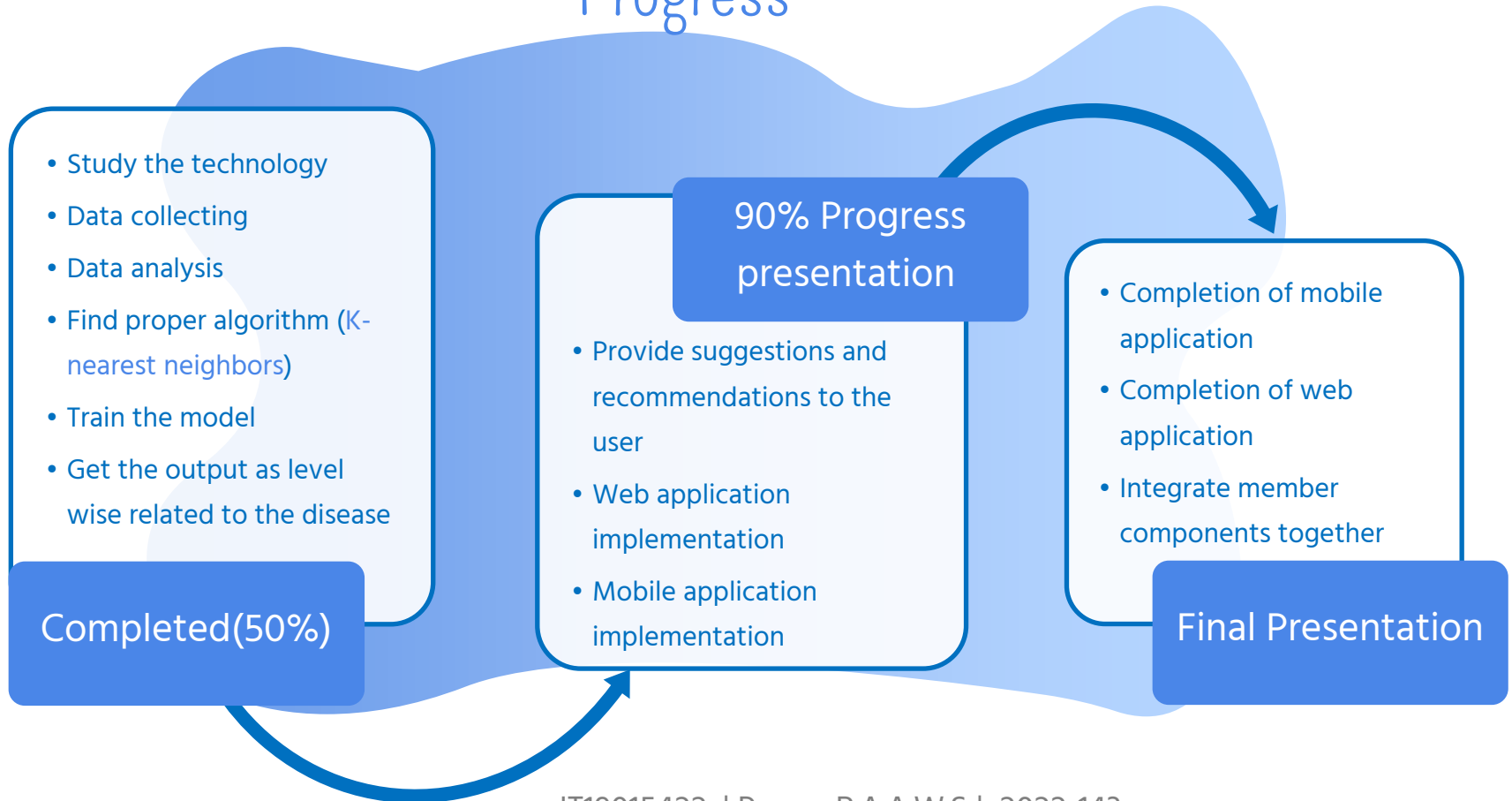
# Achieved – 50%



# System diagram



# Progress



# Latest technologies in MediSafe

## DataSet collection

kaggle



- <https://data.world/informatics-edu/heart-disease-prediction>
- <https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset>

## Model training



- K – nearest neighbor algorithm
- Libraries – pandas, sklearn, joblib, numpy .....

## IDE

ANACONDA



- Vs Code
- Jupyter notebook
- Anaconda prompt

# Requirements

## Functional

- Interoperability
- Authentication.
- Report generate
- User friendly

## Non - functional

- Quality
- Durability
- Security
- Privacy



# Risk mitigation



- ✓ Entering current situation features difficult to known by person . So that those features will get from the implemented device. (In future – 90%)
- ✓ Adults are not well fluent in new technologies.
- ✓ Validity of the disease level will depend on the user inputs.

# Completion of the project

## Model training

```
1 from sklearn.model_selection import train_test_split
2 from sklearn.preprocessing import StandardScaler
3 from sklearn.neighbors import KNeighborsClassifier
4 from sklearn.metrics import classification_report, confusion_matrix
5 import joblib as jb
6 import pandas as pd
7
8 dataframe_H = pd.read_csv('Heart_attack.csv')
9 dataframe_P = pd.read_csv('Pneumonia.csv')
10 dataframe_W = pd.read_csv('Wheezing.csv')
11
12 namesH = [
13     'age', 'Gender', 'Cholesterol', 'Pulse', 'Smoke', 'Alcohol', 'Risk'
14 ]
15
16 namesP = [
17     'age', 'Gender', 'Shortness_of_breath', 'Pulse', 'Smoke', 'Alcohol', 'Risk'
18 ]
19
20 namesW = [
21     'age', 'Gender', 'Anxiety', 'Shortness_of_breath', 'Smoke', 'Alcohol', 'Risk'
22 ]
23
24 xH = dataframe_H.iloc[:, 1:1].values
25 yH = dataframe_H.iloc[:, 6].values
26
27 xP = dataframe_P.iloc[:, 1:1].values
28 yP = dataframe_P.iloc[:, 6].values
29
30 xW = dataframe_W.iloc[:, 1:1].values
31 yW = dataframe_W.iloc[:, 6].values
32
33 X_trainH, X_testH, y_trainH, y_testH = train_test_split(xH, yH, test_size=0.20)
34 X_trainP, X_testP, y_trainP, y_testP = train_test_split(xP, yP, test_size=0.20)
```

```
32
33 X_trainH, X_testH, y_trainH, y_testH = train_test_split(xH, yH, test_size=0.20)
34 X_trainP, X_testP, y_trainP, y_testP = train_test_split(xP, yP, test_size=0.20)
35 X_trainW, X_testW, y_trainW, y_testW = train_test_split(xW, yW, test_size=0.20)
36
37 scalerH = StandardScaler()
38 scalerP = StandardScaler()
39 scalerW = StandardScaler()
40
41 scalerH.fit(X_trainH)
42 scalerP.fit(X_trainP)
43 scalerW.fit(X_trainW)
44
45 X_trainH = scalerH.transform(X_trainH)
46 X_testH = scalerH.transform(X_testH)
47
48 X_trainP = scalerP.transform(X_trainP)
49 X_testP = scalerP.transform(X_testP)
50
51 X_trainW = scalerW.transform(X_trainW)
52 X_testW = scalerW.transform(X_testW)
53
54
55 classifierH = KNeighborsClassifier(n_neighbors=6)
56 classifierP = KNeighborsClassifier(n_neighbors=6)
57 classifierW = KNeighborsClassifier(n_neighbors=6)
58
59
60 classifierH.fit(X_trainH, y_trainH)
61 classifierP.fit(X_trainP, y_trainP)
62 classifierW.fit(X_trainW, y_trainW)
63
64
65 scaler_fileH = "scalerH.save"
```

```
60 classifierH.fit(X_trainH, y_trainH)
61 classifierP.fit(X_trainP, y_trainP)
62 classifierW.fit(X_trainW, y_trainW)
63
64
65 scaler_fileH = "scalerH.save"
66 scaler_fileP = "scalerP.save"
67 scaler_fileW = "scalerW.save"
68
69 model_fileH = "model_fileH.save"
70 model_fileP = "model_fileP.save"
71 model_fileW = "model_fileW.save"
72
73 jb.dump(scalerH, scaler_fileH)
74 jb.dump(scalerP, scaler_fileP)
75 jb.dump(scalerW, scaler_fileW)
76
77 jb.dump(classifierH, model_fileH)
78 jb.dump(classifierP, model_fileP)
79 jb.dump(classifierW, model_fileW)
80
81
82 # y_predict = classifier.predict(X_test)
83
84 # # Print results:
85 # print(confusion_matrix(y_test, y_predict))
86 # print(classification_report(y_test, y_predict))
87
```

# Completion of the project

Level implementation

```
predict.py 5
> ...
from sklearn.model_selection import train_test_split
2 from sklearn.preprocessing import StandardScaler
3 from sklearn.neighbors import KNeighborsClassifier
4 #from sklearn.metrics import classification_report, confusion_matrix
5 import joblib as jb
6 import pandas as pd
7
8 scaler_fileH = "scalerH.save"
9 scaler_fileP = "scalerP.save"
10 scaler_fileW = "scalerW.save"
11
12 model_fileH = "model_fileH.save"
13 model_fileP = "model_fileP.save"
14 model_fileW = "model_fileW.save"
15
16 scalerH = jb.load(scaler_fileH)
17 scalerP = jb.load(scaler_fileP)
18 scalerW = jb.load(scaler_fileW)
19
20 classifierH = jb.load(model_fileH)
21 classifierP = jb.load(model_fileP)
22 classifierW = jb.load(model_fileW)
23
```

```
run.py 4 x predict.py 5
run.py > ...
23
24 print('-----')
25 print('                USER DETAILS                ')
26 print('-----')
27 print('')
28 age = input('Enter your age : ')
29 Gender = input('Enter your gender (1 - Male, 0 - Female) : ')
30 Cholesterol = input('Enter your Cholesterol value : ')
31 Pulse = input('Enter your pulse : ')
32 Smoke = input('Are you smoking (1 - Smoking , 0 - Not smoking ) : ')
33 Alcohol = input('Alcohol usage (1 - Yes, 0 - No) : ')
34 Shortness_of_breath = input('Have any Shortness_of_breath (1 - Yes, 0 - No) : ')
35 Anxiety = input('Have any Anxiety (1 - Yes, 0 - No) : ')
36
37 valH = [age, Gender, Cholesterol, Pulse, Smoke, Alcohol]
38 valP = [age, Gender, Shortness_of_breath, Pulse, Smoke, Alcohol]
39 valW = [age, Gender, Anxiety, Shortness_of_breath, Smoke, Alcohol]
40 valH = scalerH.transform([valH])
41 valP = scalerH.transform([valP])
42 valW = scalerH.transform([valW])
43 y_predictH = classifierH.predict(valH)
44 y_predictP = classifierH.predict(valP)
45 y_predictW = classifierH.predict(valW)
46 print('')
47 print(' Your diseases levels :')
48 print('-----')
49 print('')
50 print('Heart Attack :',y_predictH)
51 print('Pneumonia :',y_predictP)
52 print('Wheezing :',y_predictW)
53
```



# Completion of the project

Output

```
\\WINDOWS\system32\cmd.exe
(test) C:\Users\hp>cd E:\Research\new final\disease
(test) C:\Users\hp>e:
(test) E:\Research\new final\disease>python run.py
-----
                        USER DETAILS
-----

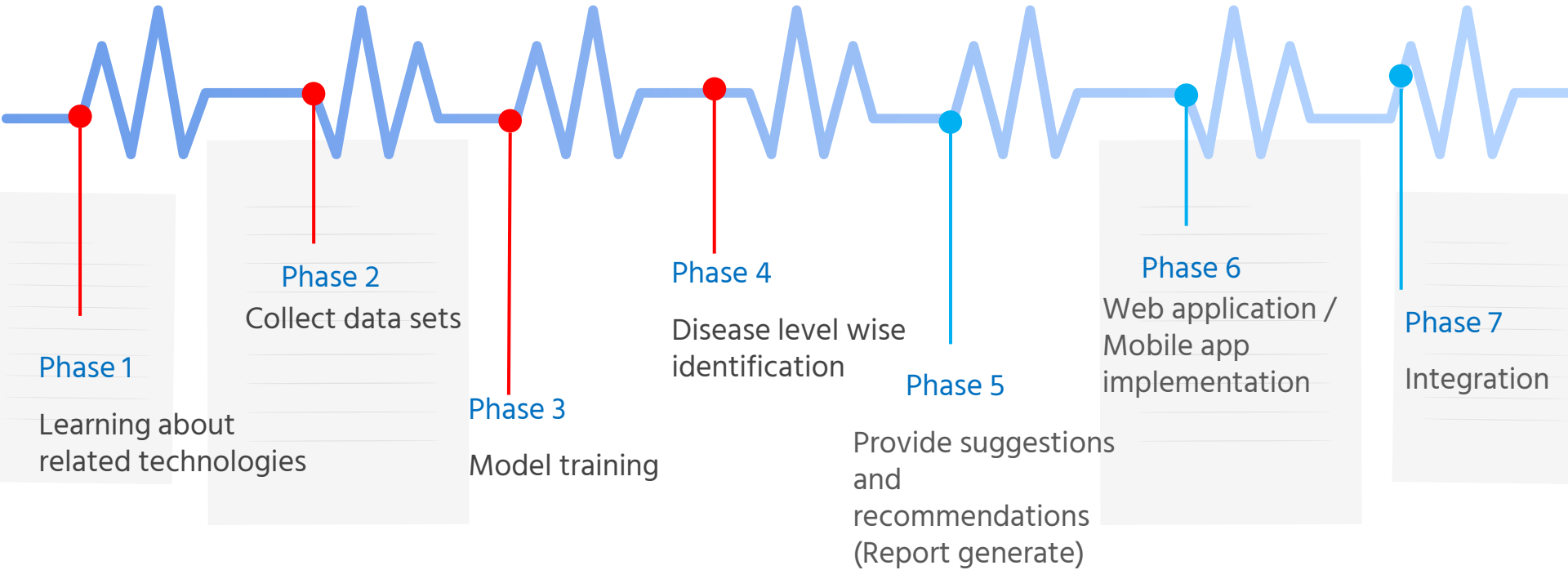
Enter your age : 35
Enter your gender (1 - Male, 0 - Female) : 0
Enter your Cholesterol value : 255
Enter your pulse : 98
Are you smoking (1 - Smoking , 0 - Not smoking ) : 0
Alcohol usage (1 - Yes, 0 - No) : 0
Have any Shortness_of_breath (1 - Yes, 0 - No) : 0
Have any Anxiety (1 - Yes, 0 - No) : 0

Your diseases levels :
-----

Heart Attack : ['Medium']
Pneumonia : ['Low']
Wheezing : ['Low']

(test) E:\Research\new final\disease>
```

# Self work breakdown structure



# Individual overleaf conference paper

The screenshot displays the Overleaf LaTeX editor interface. On the left, a file explorer shows the project structure, including a file named 'IT19015...'. The main editor area is split into two panes. The left pane shows the LaTeX source code, which includes author information for Sri Lanka Institute of Information Technology, Sri Lanka, and several authors: 1<sup>st</sup> N.H.P. Ravi Supunya Swarnakantha, 2<sup>nd</sup> U.U.Samantha Rajapaksha, 3<sup>rd</sup> T M B C K Thennakoon, 4<sup>th</sup> Perera B.A.A.W.S, 5<sup>th</sup> Rasuni Wageesha H.A, and 6<sup>th</sup> Senanayake S.A.M.B.M. The right pane shows the compiled PDF output. The title 'MEDISAFE - STAY AWAY AND DEFEAT DISEASE' is prominently displayed at the top. Below the title, the authors' names and affiliations are listed in two columns. The 1<sup>st</sup> author is N.H.P. Ravi Supunya Swarnakantha, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, ravis@slit.lk. The 2<sup>nd</sup> author is U.U.Samantha Rajapaksha, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, samanthalra@slit.lk. The 3<sup>rd</sup> author is T M B C K Thennakoon, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, it8077698@slit.lk. The 4<sup>th</sup> author is Perera B.A.A.W.S, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, it19015422@slit.lk. The 5<sup>th</sup> author is Rasuni Wageesha H.A, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, it19015040@slit.lk. The 6<sup>th</sup> author is Senanayake S.A.M.B.M, Faculty of computing, SLIIT (of Aff.), Sri Lanka Institute of Information Technology (of Aff.), Sri Lanka, it19011608@slit.lk. Below the author list, the abstract is visible, starting with 'Abstract—Pulse diagnosis is critical in eastern medicine and Ayurveda medicine. Pulse testing is widely believed by Asians to be able to identify and treat human ailments. Humans sought treatment from Ayurveda or handmade remedies before the creation of Western medicine. The attractiveness of Ayurveda is due to the fact that it is completely based on herbs, which people are aware are not chemically synthesized and are readily available. "Nadi Pariksha" is how Sri Lankans refer to "pulse diagnosis." For thousands of years, this approach of identifying sickness by monitoring the pulse has been used. Pulse checks are routinely performed on the patient's wrist. Hela Wedakama, the Sinhala neurologist who identifies the disease, does the pulse test in four different methods. In addition, there are differences in how fingers are utilized to diagnose the illness. Pulse tests are commonly divided into four categories, despite the fact that various traditional Sinhala doctors use different approaches. 1. Doshu Nadi Saasthra (Doshu Nadi Saasthra) 2. Anguli Nadi Saasthra (Anguli Nadi Saasthra) 3. Hastha Nadi Saasthra (Hastha Nadi Saasthra) 4. Boothha Nadi Saasthra (Boothha Nadi Saasthra) Ayurvedic doctors use three fingers to detect

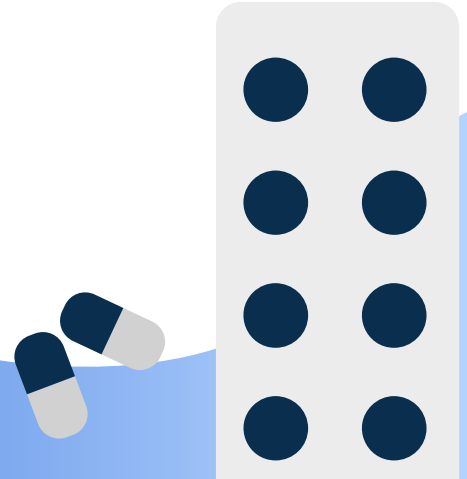
<https://www.overleaf.com/project/626e65c27e9110853ed631c7>



IT19015040

Rasuni Wageesha H.A

Specialization | Information Technology



# Table of Contents

01

Research question

02

Archived – 50%

03

System diagram

04

Progress

05

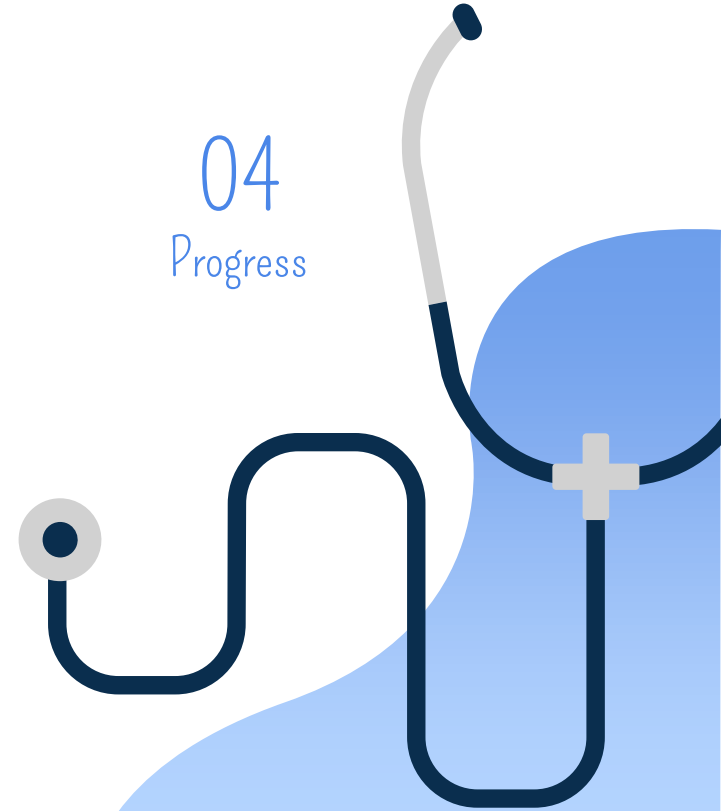
Technologies

06

Requirements

07

Risk mitigation

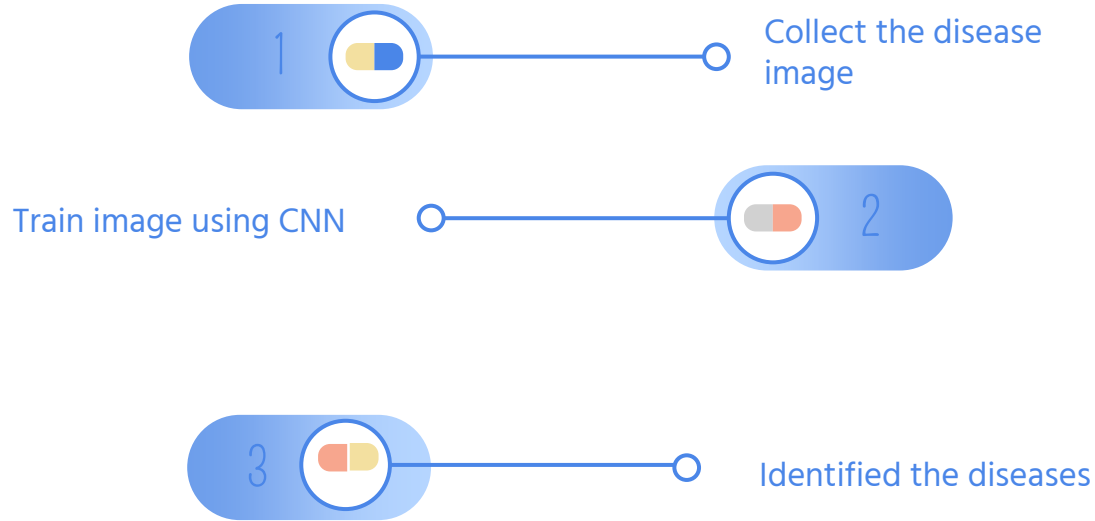


# Research question

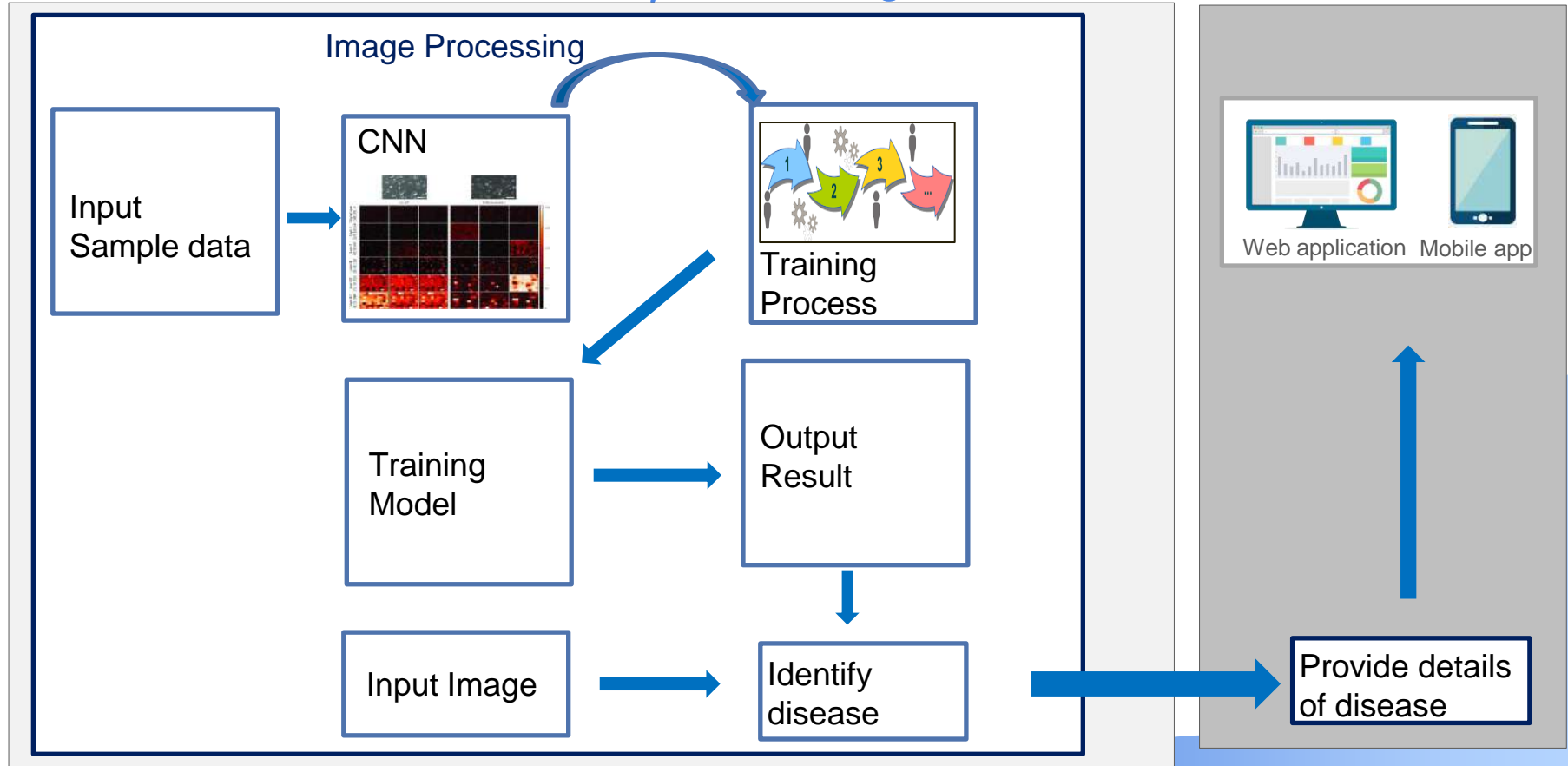


There are many different types of lung diseases and diagnosing one might be difficult.

# Achieved – 50%

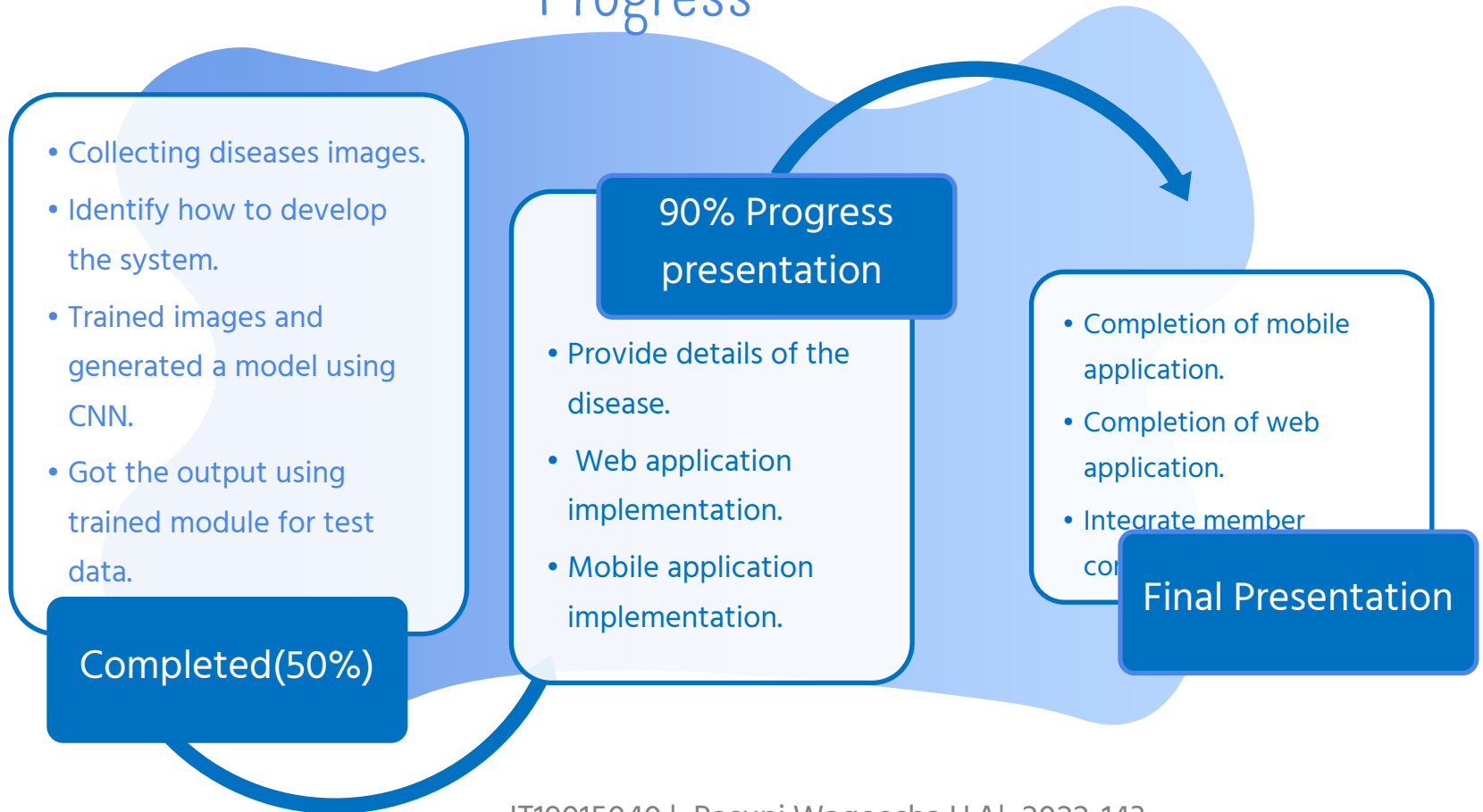


# System diagram





# Progress

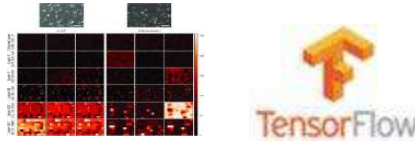


# Technologies in MediSafe

Collected data

kaggle

Algorithm



IDE



<https://www.kaggle.com/datasets/mdsafayet/multiclass-disease-classification>

Convolutional Neural Network

Teser flow

Visual Studio

# Requirements

## Functional

- Upload the lung image to the system.

## Non – functional

- Performance
- Availability
- Reliability



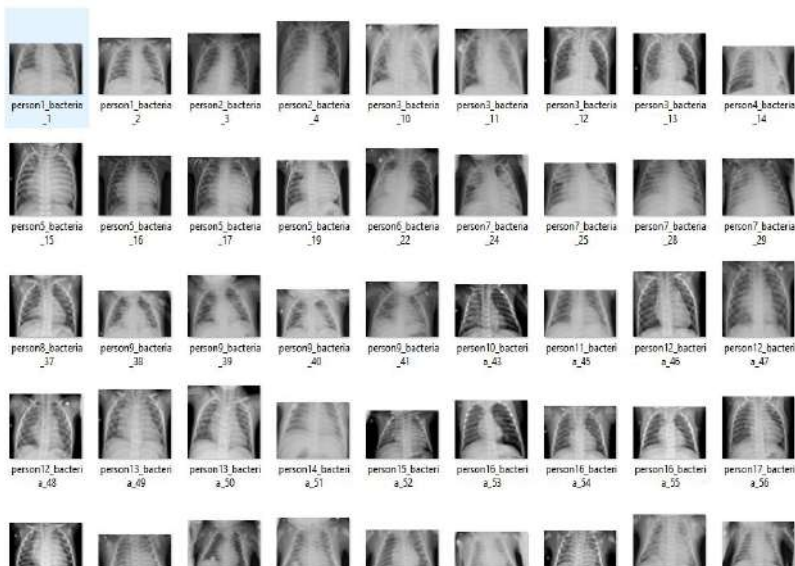
# Risk mitigation



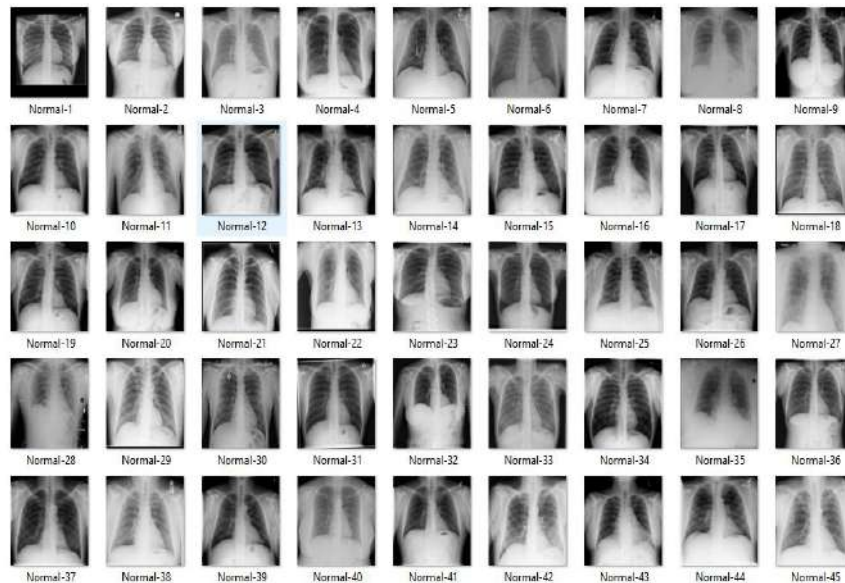
✓ Validity of the clear images.

# Completion of the project

## Normal chest X rays

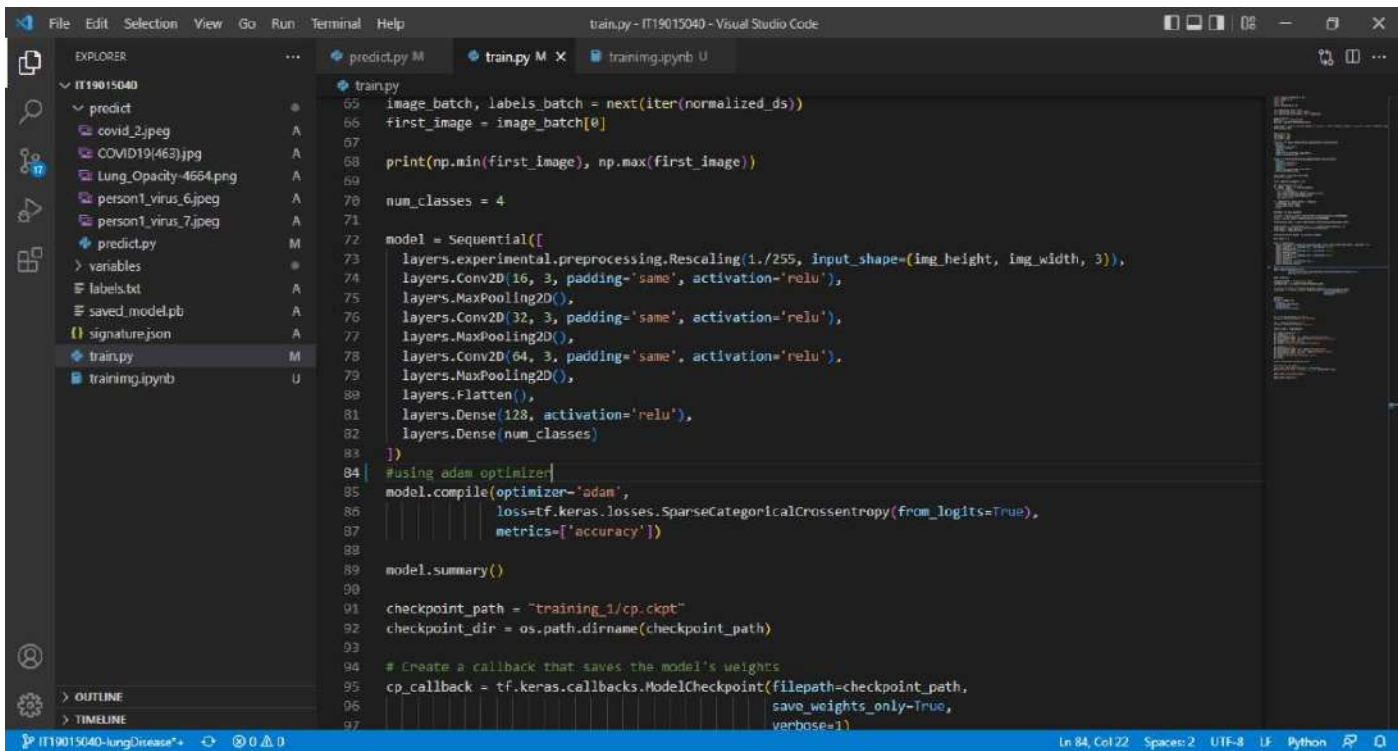


## PNEUMONIA Chest Xray



# Completion of the project

## Model training



```
65 image_batch, labels_batch = next(iter(normalized_ds))
66 first_image = image_batch[0]
67
68 print(np.min(first_image), np.max(first_image))
69
70 num_classes = 4
71
72 model = Sequential([
73     layers.experimental.preprocessing.Rescaling(1./255, input_shape=(img_height, img_width, 3)),
74     layers.Conv2D(16, 3, padding='same', activation='relu'),
75     layers.MaxPooling2D(),
76     layers.Conv2D(32, 3, padding='same', activation='relu'),
77     layers.MaxPooling2D(),
78     layers.Conv2D(64, 3, padding='same', activation='relu'),
79     layers.MaxPooling2D(),
80     layers.Flatten(),
81     layers.Dense(128, activation='relu'),
82     layers.Dense(num_classes)
83 ])
84 #using adam optimizer
85 model.compile(optimizer='adam',
86               loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
87               metrics=['accuracy'])
88
89 model.summary()
90
91 checkpoint_path = "training_1/cp.ckpt"
92 checkpoint_dir = os.path.dirname(checkpoint_path)
93
94 # Create a callback that saves the model's weights
95 cp_callback = tf.keras.callbacks.ModelCheckpoint(filepath=checkpoint_path,
96                                                  save_weights_only=True,
97                                                  verbose=1)
```

# Completion of the project

## Model training



The screenshot displays a Google Colab notebook interface. The top bar shows the notebook title 'training.ipynb' and a star icon. Below the title bar is a menu with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. The 'Help' menu is open, showing 'Last saved at 21:19'. On the right side of the top bar are icons for 'Comment', 'Share', and a settings gear. The left sidebar contains a 'Table of contents' panel with a search icon and a 'Section' header. The main area of the notebook shows two code cells. The first cell contains the code to mount Google Drive: 

```
[ ] from google.colab import drive
drive.mount('/content/drive')
```

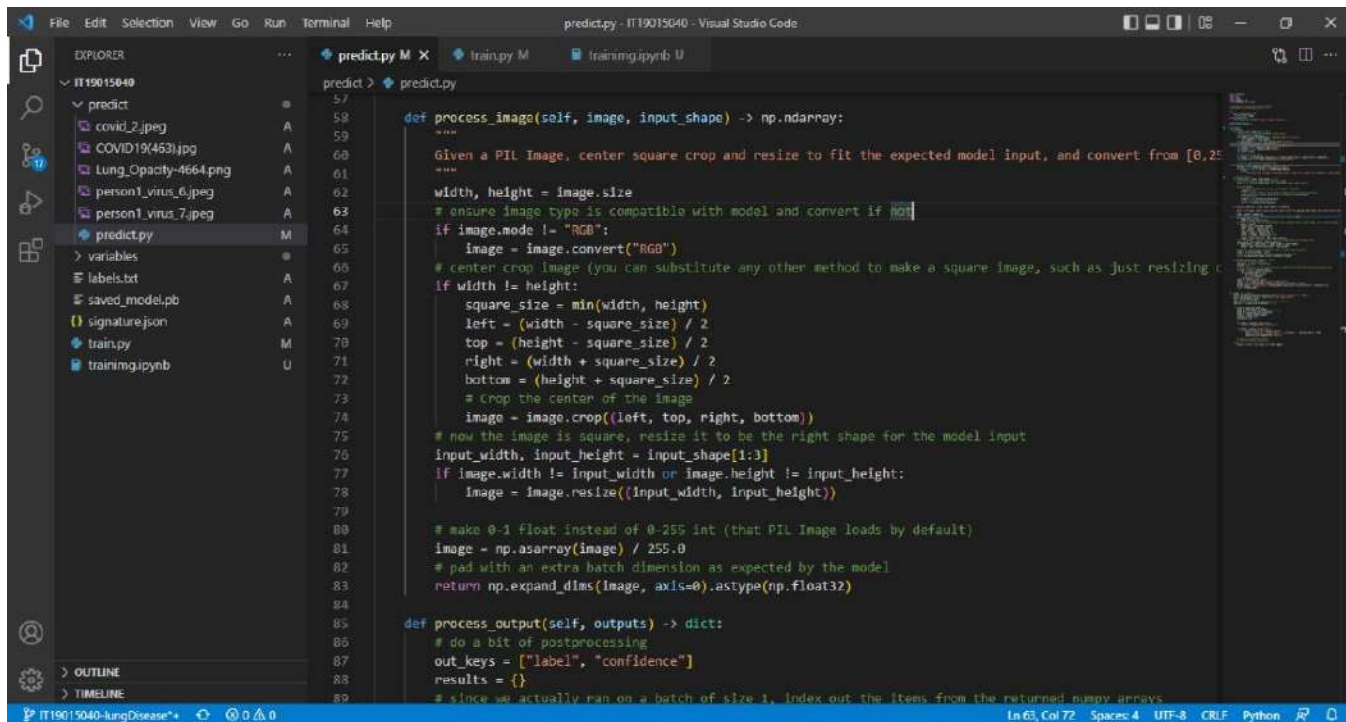
 Below the code is the output: 'Mounted at /content/drive'. The second cell contains the code to list and load a COVID-19 chest X-ray image: 

```
[ ] covid = list(data_dir.glob('covid/*'))
PIL.Image.open(str(covid[0]))
```

 Below the code is the output: a grayscale chest X-ray image showing the lungs and ribcage.

# Completion of the project

## Pre-processing



```
File Edit Selection View Go Run Terminal Help
predict.py - IT9015040 - Visual Studio Code

EXPLORER
IT9015040
├── predict
│   ├── covid_2.jpeg
│   ├── COVID19(453).jpg
│   ├── Lung_Opacity-4664.png
│   ├── person1_virus_6.jpeg
│   ├── person1_virus_7.jpeg
│   └── predict.py
├── variables
├── labels.txt
├── saved_model.pb
├── signature.json
├── train.py
└── training.ipynb

predict.py
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89

def process_image(self, image, input_shape) -> np.ndarray:
    """
    Given a PIL Image, center square crop and resize to fit the expected model input, and convert from [0,255] to [0,1] float.
    """
    width, height = image.size
    # ensure image type is compatible with model and convert if not
    if image.mode != "RGB":
        image = image.convert("RGB")
    # center crop image (you can substitute any other method to make a square image, such as just resizing)
    if width != height:
        square_size = min(width, height)
        left = (width - square_size) / 2
        top = (height - square_size) / 2
        right = (width + square_size) / 2
        bottom = (height + square_size) / 2
        # Crop the center of the image
        image = image.crop((left, top, right, bottom))
    # now the image is square, resize it to be the right shape for the model input
    input_width, input_height = input_shape[1:3]
    if image.width != input_width or image.height != input_height:
        image = image.resize((input_width, input_height))

    # make 0-1 float instead of 0-255 int (that PIL Image loads by default)
    image = np.asarray(image) / 255.0
    # pad with an extra batch dimension as expected by the model
    return np.expand_dims(image, axis=0).astype(np.float32)

def process_output(self, outputs) -> dict:
    # do a bit of postprocessing
    out_keys = ["label", "confidence"]
    results = {}
    # since we actually ran on a batch of size 1, index out the items from the returned numpy arrays
```



# Completion of the project

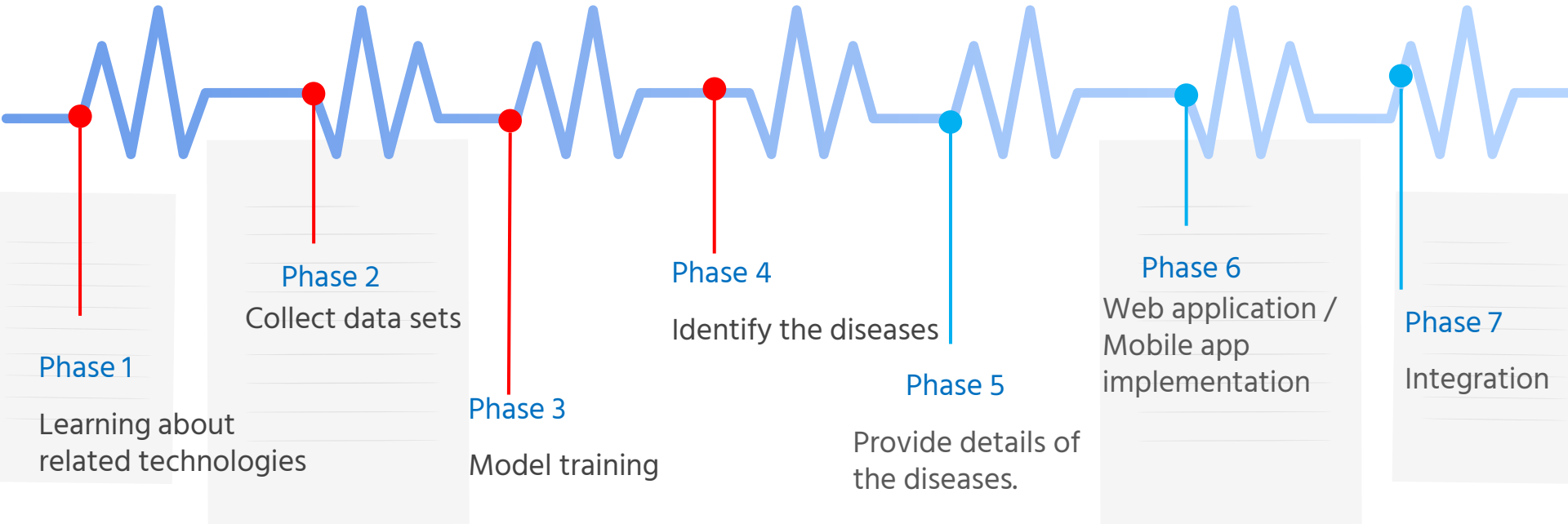
## Output

```
C:\Windows\system32\cmd.exe

(image) C:\Users\User\Desktop\image\model\predict>python predict.py covid_2.jpeg
2022-05-10 23:19:22.352560: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-05-10 23:19:36.645223: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlerror: nvcuda.dll not found
2022-05-10 23:19:36.645438: W tensorflow/stream_executor/cuda/cuda_driver.cc:326] failed call to cuInit: UNKNOWN ERROR (303)
prediction: COVID19 Chest Xray | confidence: 97.37206101417542

(image) C:\Users\User\Desktop\image\model\predict>
```

# Self work breakdown structure



# Individual overleaf conference paper

The screenshot displays the Overleaf LaTeX editor interface. The top navigation bar includes options like Menu, Upgrade, and various utility icons. The left sidebar shows a file explorer with files like 'conference\_101719.pdf', 'fig1.png', and 'IT9015040.tex'. The main editor area shows the LaTeX source code for a document titled 'MEDISAFE - STAY AWAY AND DEFEAT DISEASE'. The code includes package declarations, author information, and a title. The right sidebar shows a preview of the rendered document, which is a conference paper template. The preview includes the title, authors, and a table of contents.

conference-latex-template\_10-17-19

Source Rich Text

conference\_101719.pdf

fig1.png

IEEEtran\_HOWTO.pdf

IEEEtran.cls

IT9015040.tex

```
1 \documentclass[conference]{IEEEtran}
2 \IEEEoverridecommandlockouts
3 % The preceding line is only needed to identify funding in the
  first footnote. If that is unneeded, please comment it out.
4 \usepackage{cite}
5 \usepackage{amsmath,amssymb,amsfonts}
6 \usepackage{algorithmic}
7 \usepackage{graphicx}
8 \usepackage{textcomp}
9 \usepackage{xcolor}
10 \def\BibTeX{{\rm B}\kern-.05em{\sc i}\kern-.025em{\rm t}\kern-.08em
11   T}\kern-.1667em\lower.7ex\hbox{E}}\kern-.125emX}}
12 \begin{document}
13
14 \title{MEDISAFE - STAY AWAY AND DEFEAT DISEASE}
15 {\footnotesize \textsuperscript}
16 \thanks{Identify applicable funding agency here. If none,
  delete this.}
17 }
18
19 \author{\IEEEauthorblockN{1}\textsuperscript{st} N.H.P. Ravi
  Supunya Swarnakantha }
20 \IEEEauthorblockA{\textit{ Faculty of computing, SLIIT (of
  Aff.) }}
21 \textit{Sri Lanka Institute of Information Technology (of
  Aff.)}}
22 Sri Lanka
23 ravi.s@slit.lk
24 }
```

Recompile

MEDISAFE - STAY AWAY AND DEFEAT DISEASE

1<sup>st</sup> N.H.P. Ravi Supunya Swarnakantha  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
ravi.s@slit.lk

2<sup>nd</sup> U.U.Samarathna Rajapaksha  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
samarathna.u@slit.lk

3<sup>rd</sup> T.M.B.C.K. Thennakoon  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
t.m.b.c.k.thennakoon@slit.lk

4<sup>th</sup> Perera H.A.A.W.S  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
h.a.a.w.s.perera@slit.lk

5<sup>th</sup> Rasuni Wageesha H.A  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
rasuni.w@slit.lk

6<sup>th</sup> Senanayake S.A.M.H.M  
Faculty of computing, SLIT (of Aff.)  
Sri Lanka Institute of Information Technology (of Aff.)  
Sri Lanka  
s.a.m.h.m.senanayake@slit.lk

**Abstract**—In medicine, image diagnosis is a crucial issue. Modern facilities enable doctors to diagnose diseases early and precisely, reducing the need for unwanted treatment procedures. As a result, image diagnosis is at the forefront of disease prevention, diagnosis, and therapy. Heart and lung failures are the leading causes of death in the United States, with plain film chest X-rays being the most used screening method (1). Because of the increasing number of patients, doctors are overwhelmed, and they are unable to consult and direct their patients' care. As a result, an image categorization computer system is required. We present a deep learning model for detecting relevant features in chest X-ray pictures in this work. The suggested model, known as Multi-CNN, combines several Convolutional Neural Networks to determine the input image. The digital chest X-ray image dataset was collected from June 2017 to March 2018 at An Binh Hospital in HCM, Vietnam (AF-CXR-Dataset). Each Multi-CNN component is a convolutional neural network created using the ConvNet5 package. Normal/abnormal density is the outcome of the suggested model. We also present a strategy for synthesizing the results of the model's components, which we term Fusion rules, in this work. The experimental results in our X-ray image dataset indicated that the suggested Multi-CNN model is feasible.

**Index Terms**—Image classification, Chest X-ray image, convolutional neural network (CNN), chest X-ray image classification, multi-CNN, Global the disease images

**1. INTRODUCTION**

Heart and lung failures are the leading causes of death in the United States, with plain X-Ray (CXR) being the most prevalent screening method (2). X-ray and CT pictures are highly applicable imaging agency here. If none, delete this.

commonly utilized in the diagnosis of cancer signs in Vietnam and most other nations across the world. Radiographs, or chest X-ray scans, provide a single view of the chest cavity. CT scans can provide a full image of the inside of the chest, making it easier to detect the shape, size, location, and density of lung nodules (3). CT scan schedules, on the other hand, is costly and not always available in smaller institutions or rural locations. Basic chest radiographs, on the other hand, are reasonably inexpensive and quick, and expose the patient to too little radiation, therefore they are frequently used to diagnose abnormalities (3). Machine learning has been employed in the identification and classification of medical pictures in recent years to aid in diagnosis. Identification of exogenous disease at an early stage is can especially assist the radiologist in reducing workload. To measure this difficulty, a variety of classification algorithms have been developed. Popular methods for handling picture classification problems currently include K-Means, K-NN, deep neural networks, and Support Vector Machine (SVM). For picture classification problems, one of the most often used methodologies in the Artificial Neural Networks method. The convolutional neural network (CNN) (1) is a deep learning model that has gained academics' interest in recent years. It's widely utilized in picture categorization, image recognition, language translation, medical diagnosis, and a variety of other fields, with excellent accuracy. As a result, we present a Multi-CNN model based on convolutional neural networks in this research. We also present a strategy for synthesizing the results of the model's components, which we term Fusion rules, in this work.

File outline

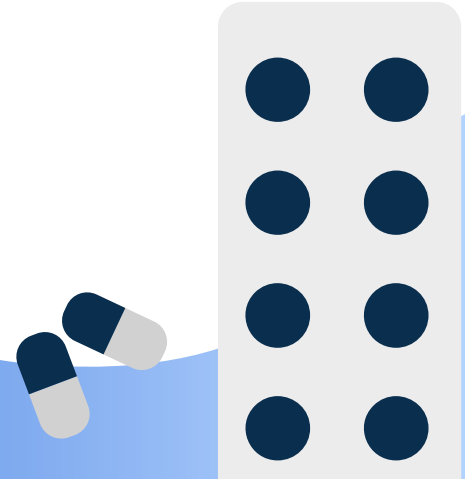
<https://www.overleaf.com/project/626ea0e7a31e802185bd3e61>



IT19011608

Senanayaka S.A.M.A.B.M

Specialization | Information Technology



# Table of Contents

01

Research question

02

Archived – 50%

03

System diagram

04

Progress

05

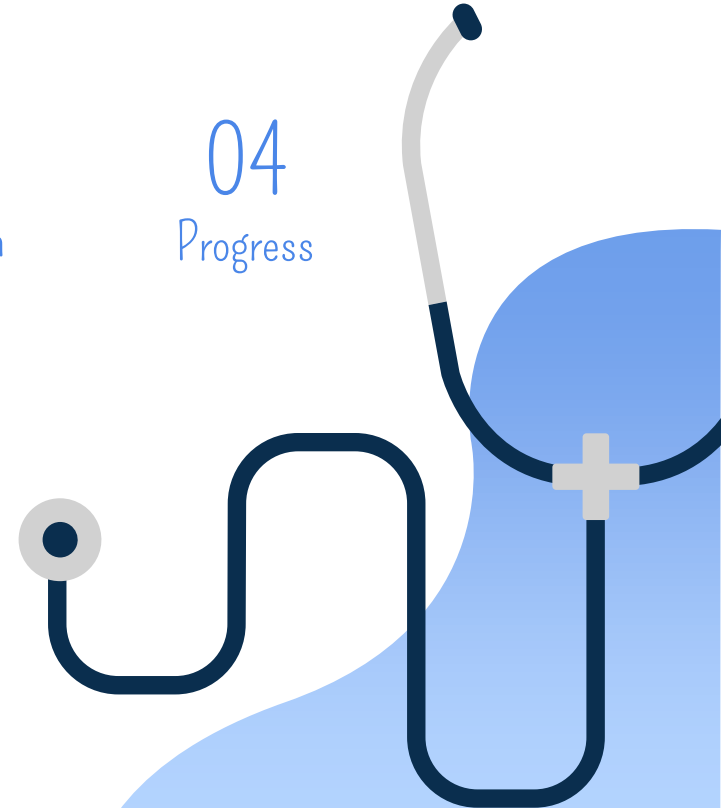
Technologies

06

Requirements

07

Risk mitigation

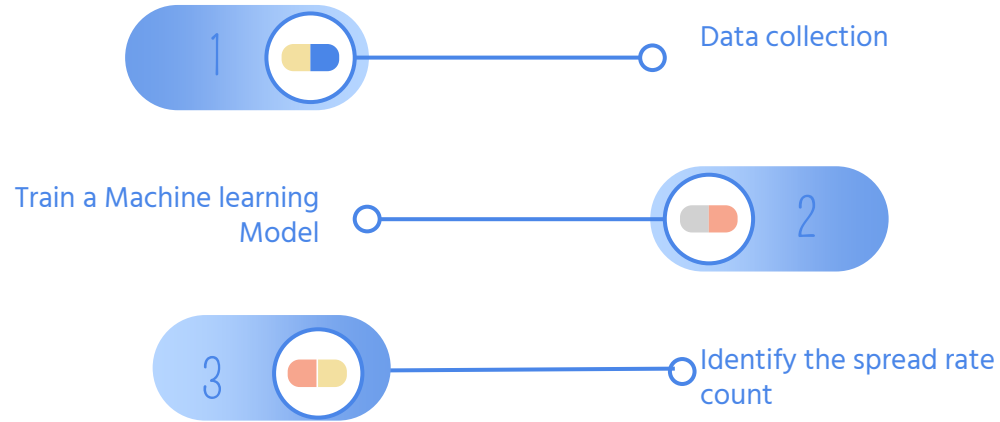


# Research question

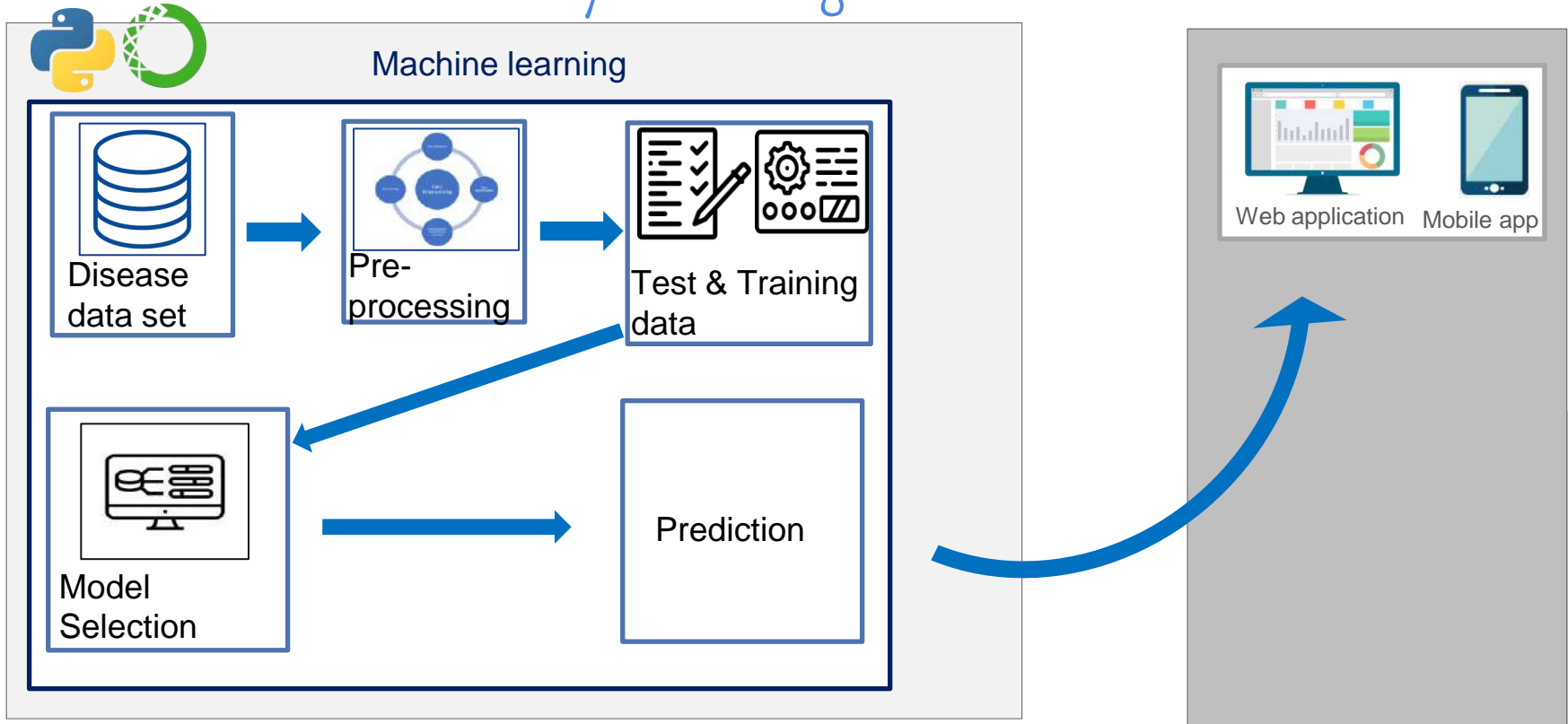


- Identify the diseases count on the sri lanka
- Target domain

# Achieved – 50%

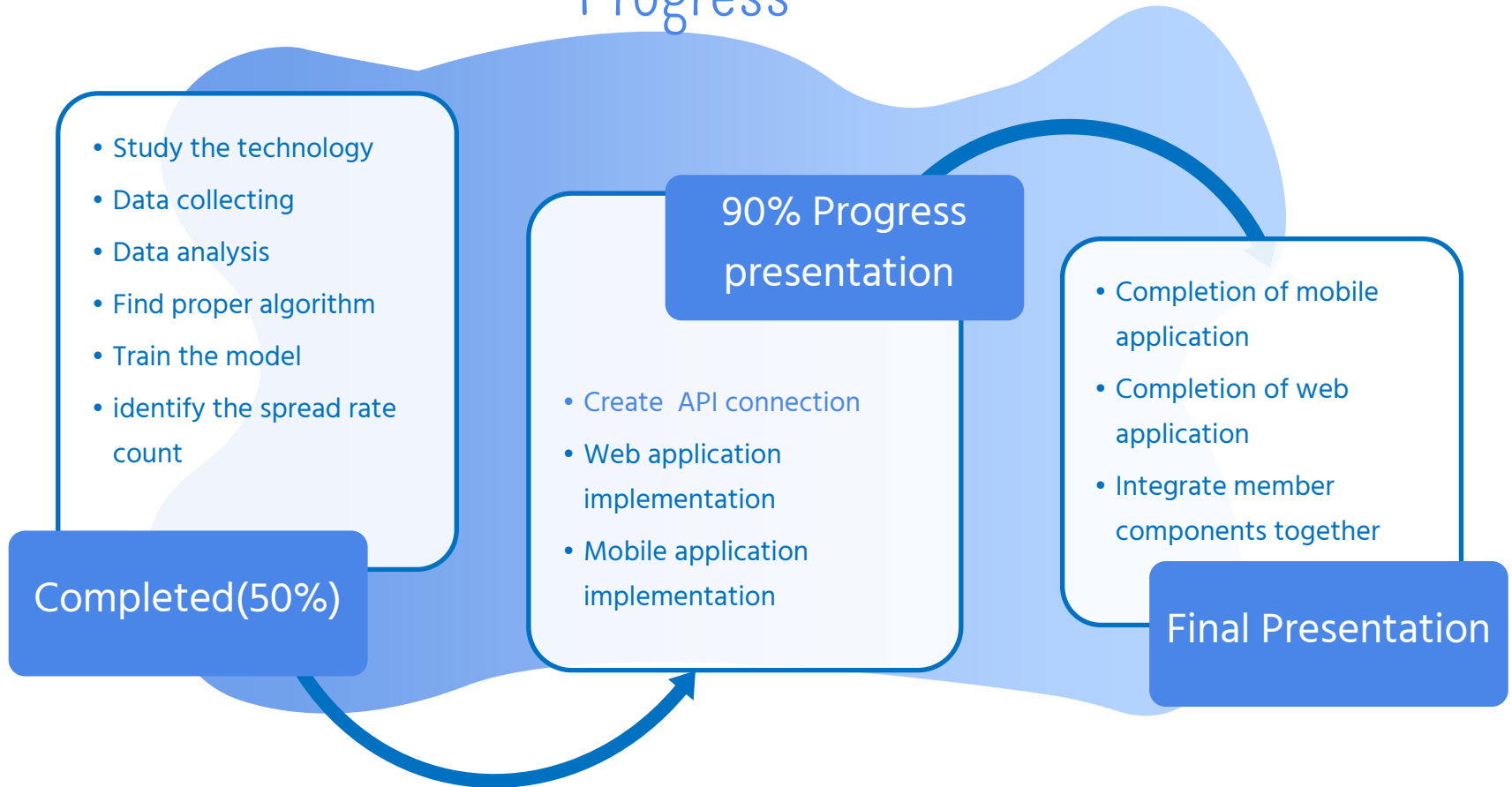


# System diagram





# Progress

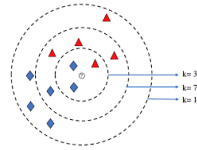


# Technologies in MediSafe

Collected data

kaggle

Algorithm



Ide



- 1) <https://www.healthdata.org/sri-lanka>
- 2) <https://www.kaggle.com/>

Long short term-memory  
algorithms

Jupyter notebook and vs code

# Requirements

## Functional

- Identify the spread rate count
- Display the data healthcare dashboard

## Non – functional

- Accuracy
- Availability



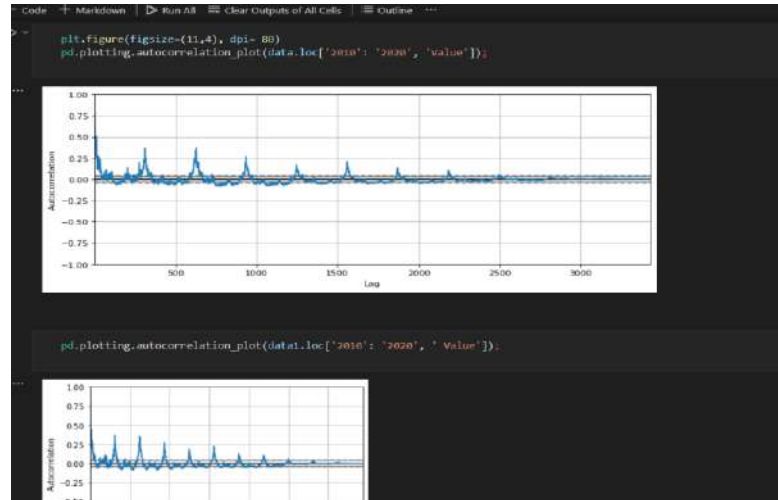
# Risk mitigation

Find relevant and accurate data



# Completion of the project

Pre- processing stage



# Completion of the project

## Model training stage

```
TimePredict.py 2 TimeLstm.ipynb disease_lstm.ipynb
TimeLstm.ipynb > model = Sequential(layers=[Dense(50, input_shape=(train_X.shape[1], train_X.shape[2])),
Code + Markdown | Run All | Clear Outputs of All Cells | Variables | Outline ...
D> model.add(Dense(50, input_shape=(train_X.shape[1], train_X.shape[2])))
model.add(Dense(1))
model.compile(loss='mse', optimizer='adam')

history = model.fit(train_X, train_y, epochs=50, batch_size=32, validation_data=(test_X, test_y), verbose=2, shuffle=False)

pyplot.plot(history.history['loss'], label='train')
pyplot.legend()
pyplot.show()

'''
Output exceeds the size limit. Open the full output data in a text editor
Epoch 1/50
229/229 - 4s - loss: 0.0139 - val_loss: 0.0111 - 4s/epoch - 17ms/step
Epoch 2/50
229/229 - 1s - loss: 0.0110 - val_loss: 0.0099 - 712ms/epoch - 3ms/step
Epoch 3/50
229/229 - 1s - loss: 0.0092 - val_loss: 0.0093 - 670ms/epoch - 3ms/step
Epoch 4/50
229/229 - 1s - loss: 0.0086 - val_loss: 0.0088 - 650ms/epoch - 3ms/step
Epoch 5/50
229/229 - 1s - loss: 0.0086 - val_loss: 0.0088 - 612ms/epoch - 3ms/step
Epoch 6/50
229/229 - 1s - loss: 0.0087 - val_loss: 0.0088 - 649ms/epoch - 3ms/step
Epoch 7/50
229/229 - 1s - loss: 0.0086 - val_loss: 0.0089 - 635ms/epoch - 3ms/step
Epoch 8/50
229/229 - 1s - loss: 0.0086 - val_loss: 0.0095 - 631ms/epoch - 3ms/step
Epoch 9/50
229/229 - 1s - loss: 0.0086 - val_loss: 0.0087 - 608ms/epoch - 3ms/step
```

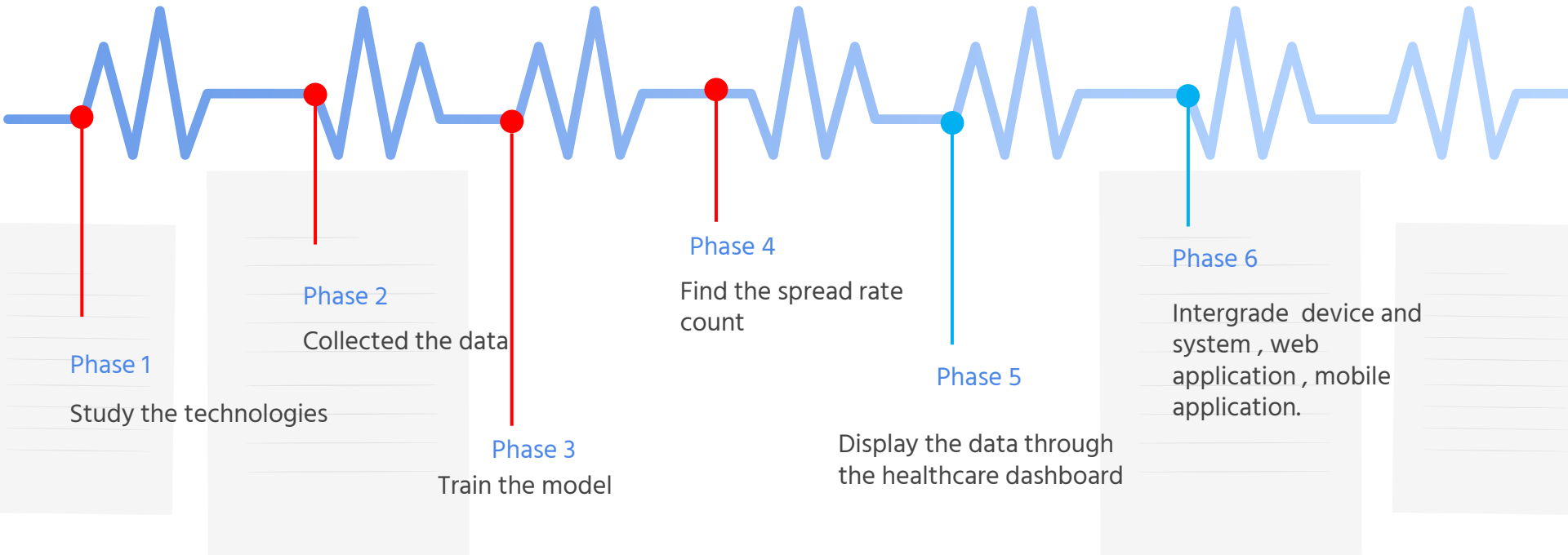
# Completion of the project

## Output

```

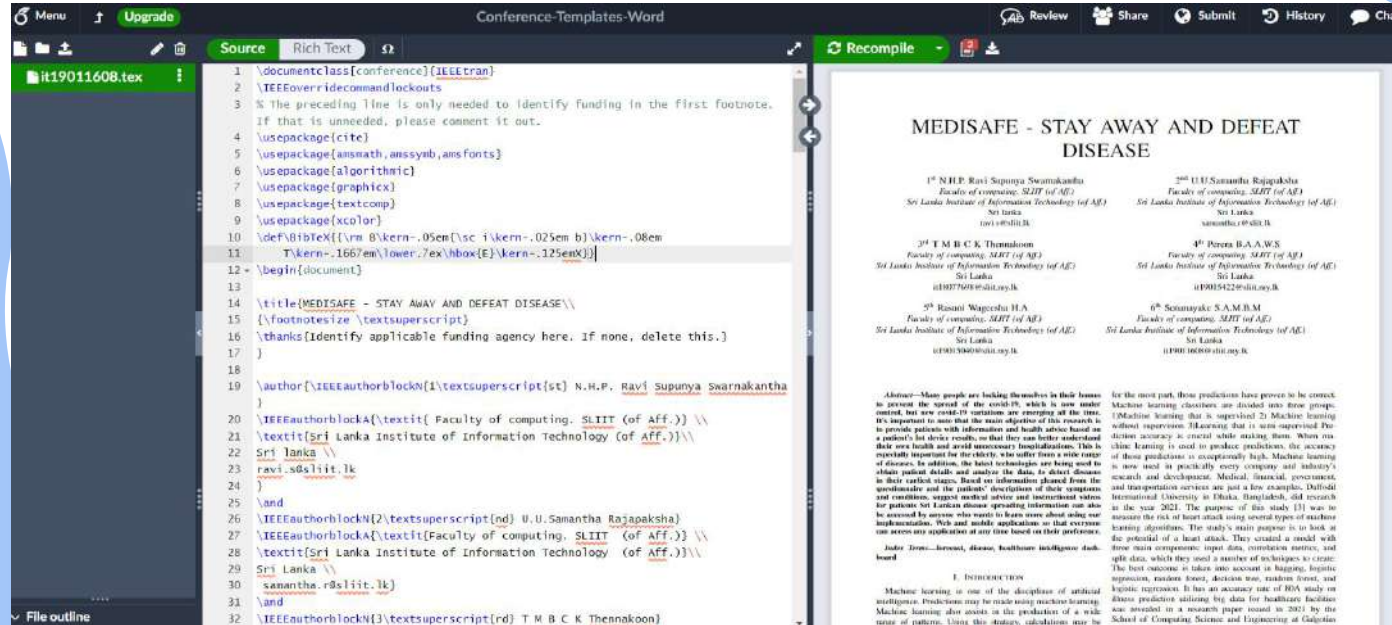
C:\Windows\system32\cmd.exe
2022-05-24 23:20:41.885880: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cufft64_10.dll'; dlerror: cufft64_10.dll not found
2022-05-24 23:20:41.886792: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'curand64_10.dll'; dlerror: curand64_10.dll not found
2022-05-24 23:20:41.887676: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cusolver64_11.dll'; dlerror: cusolver64_11.dll not found
2022-05-24 23:20:41.888567: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cusparse64_11.dll'; dlerror: cusparse64_11.dll not found
2022-05-24 23:20:41.889450: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudnn64_8.dll'; dlerror: cudnn64_8.dll not found
2022-05-24 23:20:41.889608: W tensorflow/core/common_runtime/gpu/gpu_device.cc:1850] Cannot dlopen some GPU libraries. Please make sure the missing libraries mentioned above are installed properly if you would like to use GPU. Follow the guide at https://www.tensorflow.org/install/gpu for how to download and setup the required libraries for your platform.
Skipping registering GPU devices...
2022-05-24 23:20:41.889987: I tensorflow/core/platform/cpu_feature_guard.cc:151] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in performance-critical operations: AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
C:\Users\Ruddhi\.conda\envs\disease\lib\site-packages\sklearn\base.py:329: UserWarning: Trying to unpickle estimator MinMaxScaler from version 1.0.2 when using version 1.1.0. This might lead to breaking code or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/modules/model_persistence.html#security-maintainability-limitations
  warnings.warn(
C:\Users\Ruddhi\.conda\envs\disease\lib\site-packages\sklearn\base.py:329: UserWarning: Trying to unpickle estimator LabelEncoder from version 1.0.2 when using version 1.1.0. This might lead to breaking code or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/modules/model_persistence.html#security-maintainability-limitations
  warnings.warn(
Input your Diseases : Dengue
Input your city : Gampaha
Input the date you want to search for in YYYY/MM/DD : 2023/05/30
[7] [0]
[[0.23333335 0.         0.1338509 ]]
[[0.23333335 0.         0.1338509 ]]
C:\array_function__ internals:180: VisibleDeprecationWarning: Creating an ndarray from ragged nested sequences (which is a list-or-tuple of lists-or-tuples-or ndarrays with different lengths or shapes) is deprecated. If you meant to do this, you must specify 'dtype=object' when creating the ndarray.
the predicted date : 2023/05/30 spread rate count is : 1043
(disease) D:\Research>
```

# Self work breakdown structure





# Individual overleaf conference paper



<https://www.overleaf.com/project/626ea4545e41706fc4ffa8d9>

# Commercialization



01

We discussed with some surgeons and they allow us to test that implemented system in their premises. (Dispensary, Hospitals)

02

After completing our project we are hoping to introduce this to some clinical centers.

03

Advertise our mediSafe product via social media with its benefits.

The background features two large, abstract, organic shapes in a medium blue color. One shape is on the left, curving upwards and to the right. The other is on the top right, curving downwards and to the left. They frame the central text.

# Demonstration

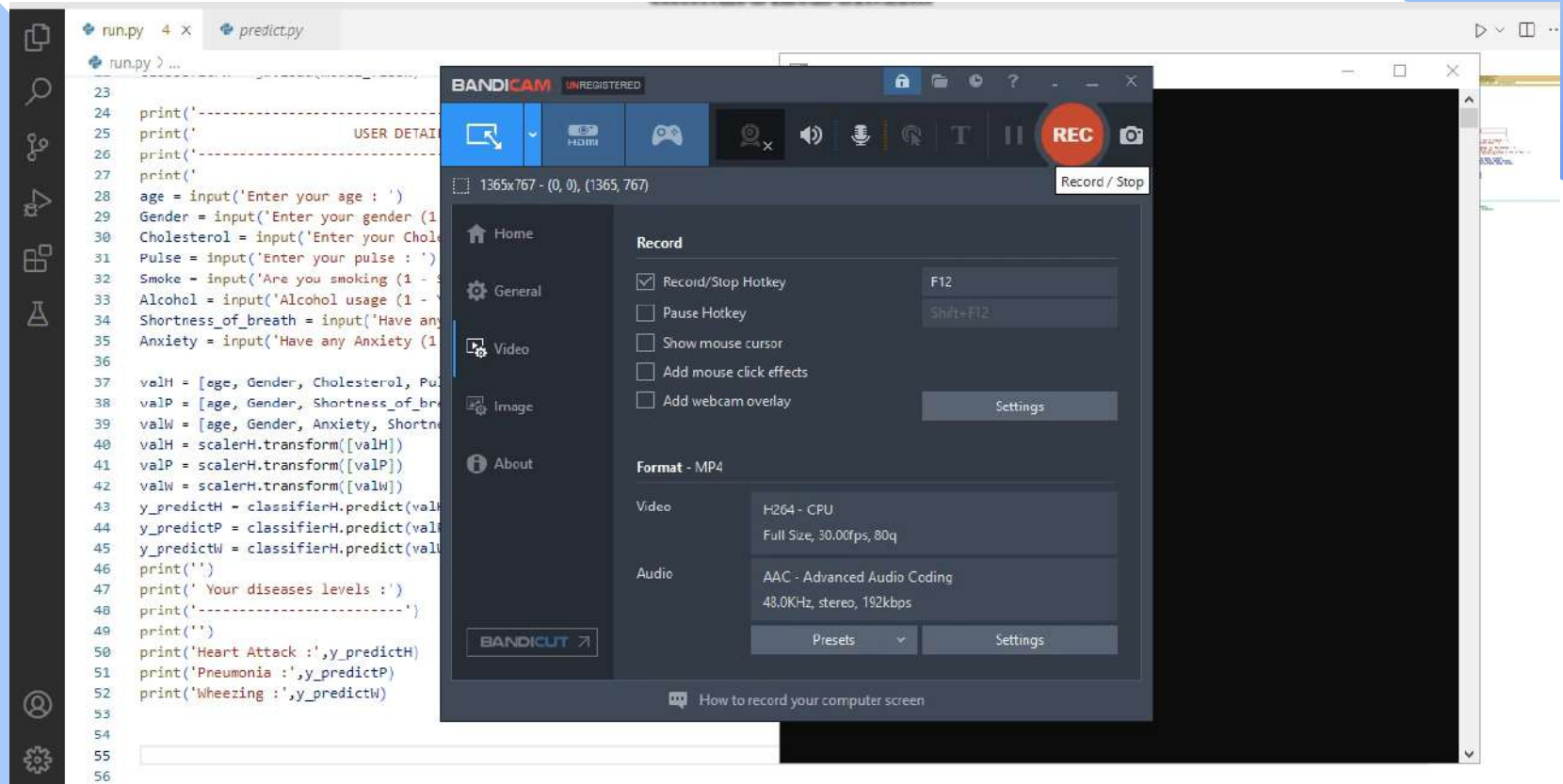
# Demonstration

The screenshot displays a Bandicam recording window. The main content is a web browser showing the Firebase Realtime Database console for a project named 'MediSafe Research'. The 'Data' tab is active, showing a JSON structure under the path 'UNIT\_1':

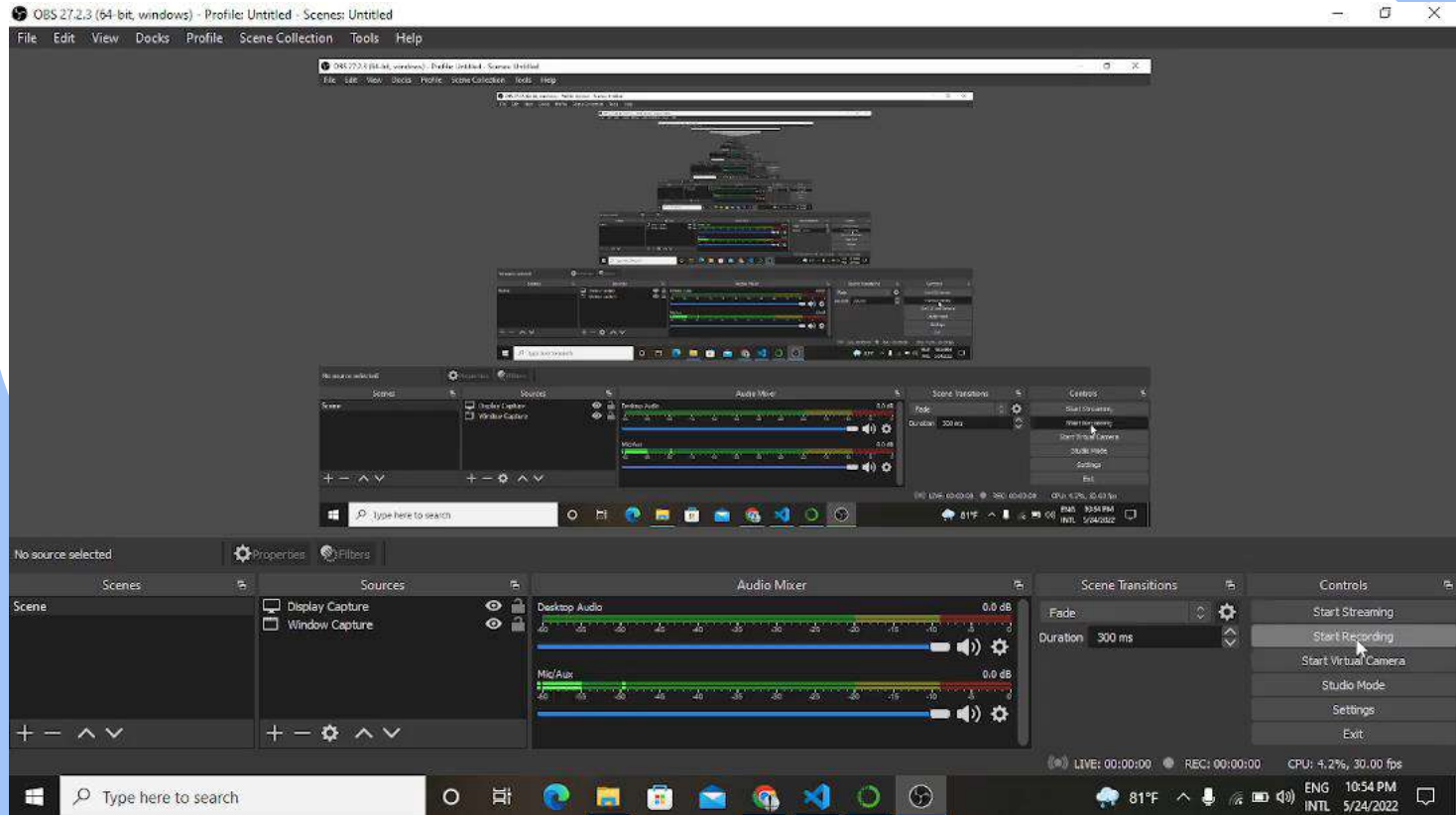
```
https://medisafe-research-default-rtdb.firebaseio.com/  
  
UNIT_1  
  FLOW: 0  
  S02: 98  
  pulse: 40  
  temp: 37.6875
```

Below the JSON, it indicates 'Database location: United States (us-central1)'. To the right of the console, there is a video feed of a physical device, a small white box with a blue cable and a black probe, labeled 'Thennakoon T M B C K it18077698'. Below the video feed is a circular logo with the letters 'PI'. The Windows taskbar at the bottom shows the time as 9:28 AM on 5/22/2022. A watermark 'Thennakoon T M B C K it18077698' is visible in the bottom left corner of the recording area.

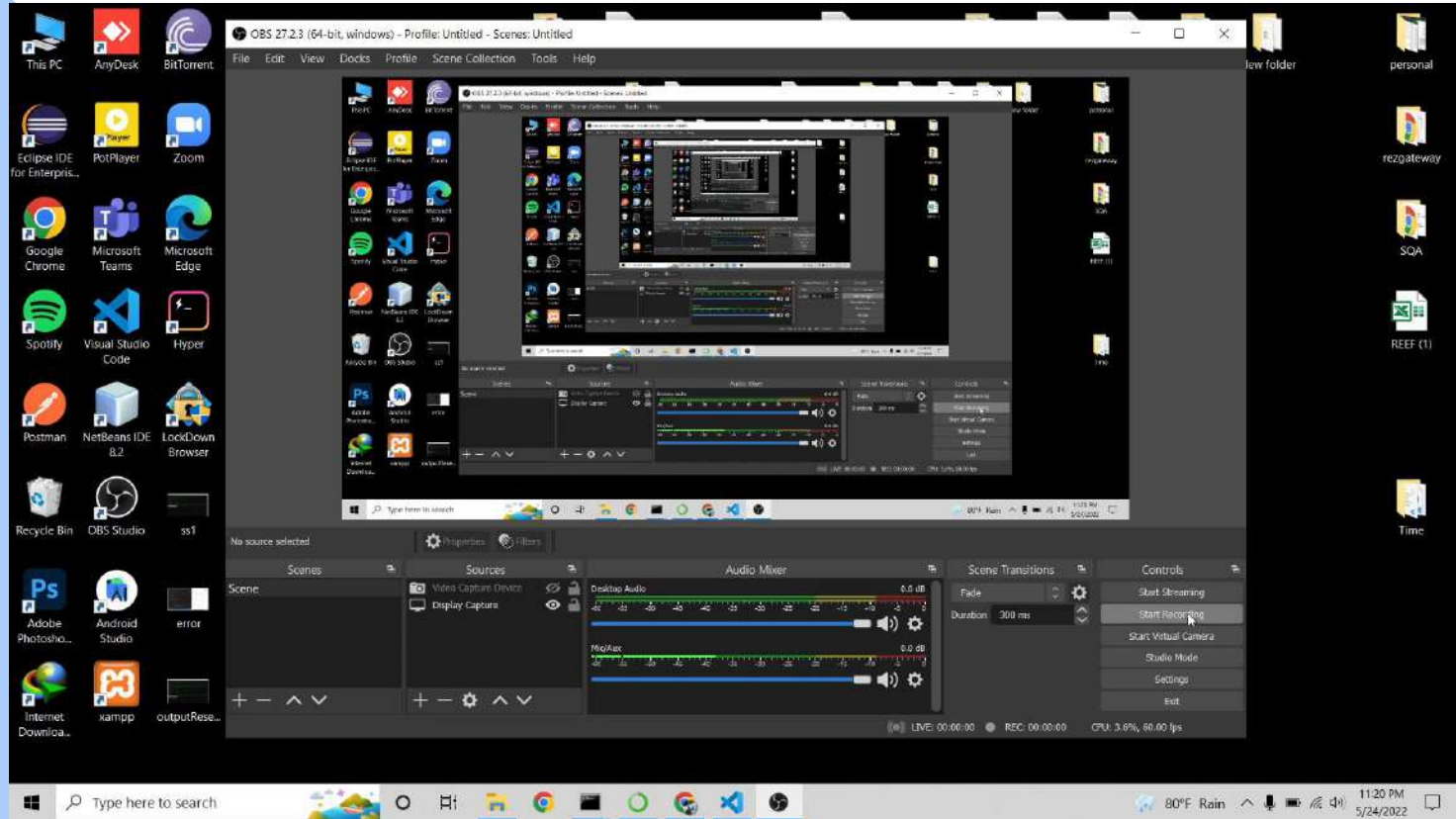
# Demonstration



# Demonstration



# Demonstration





# Thank You

**Do you have any questions?**