Project Documentation

Team ID	Team-592696
Project Name	Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques
Team Members	Buddepu Shiva Kothoju Navyeesh Kanakala Hinduja Mutina Sathwik
Mentor	Saumya Mohandas

1. INTRODUCTION

Deep learning techniques for COVID-19 detection from chest X-rays have been the focus of a lot of research and development, particularly during the COVID-19 epidemic period. Convolutional neural networks (CNNs) is one type of deep learning model, have demonstrated potential in aiding radiologists and other healthcare professionals in the diagnosis of COVID-19 from medical images such as chest X-rays.

However, it is laborious and prone to human error to manually recognize COVID-19 from a group of chest X-ray images that include both COVID-19 and pneumonia patients. Therefore, deep learning algorithms-driven artificial intelligence methods that recognize patterns in radiography images and identify the presence of COVID-19 have the potential to improve the existing diagnostic procedure.

It is important to remember that, even while deep learning models can be effective tools for supporting COVID-19 diagnosis, they should only be used to supplement medical practitioner's knowledge but not to replace it. Before being used in clinical settings, models must also undergo thorough validation and have regulatory authority.

Preprocessing the data is necessary to make sure the pictures are appropriate for deep learning. This could entail expanding the data to make it larger and more diverse, downsizing the photographs to a uniform size, and standardizing pixel values. The resilience and generalization of the model can be enhanced with the use of data augmentation approaches. Create a deep learning model using model architecture; a convolutional neural network (CNN) is a common choice.

We need to Update the model frequently as new information becomes available and our understanding of COVID-19 develops. To keep the model accurate, regular validation and retraining are necessary. Be aware of the biases and ethical issues in your model. Make sure the dataset accurately reflects the population and take into account the possible repercussions of false positives and false negatives.

Additionally, radiological screening procedures including computed tomography (CT) and CXR are used to diagnose and identify COVID-19. People all over the world have noted that CXR is one of the best ways to diagnose pneumonia since it is a quick, low-cost, widely used clinical technique that exposes the patient to less radiation than CT scans. On a CXR, radiologists must search for the radiological indicators that indicate COVID-19 symptoms. Automating the CXR analysis, a laborious and error-prone process that requires a lot of time and effort, is crucial to saving time and effort.

1.1 Project Overview

Worldwide healthcare systems are under unusual strain as a result of the COVID-19 pandemic, which requires quick identification in order to stop the virus's spread and treat sick people in a timely manner. Chest X-rays are one diagnostic method utilized in this context; they can show anomalies in the lungs, particularly those related to COVID-19. Deep learning methods have the potential to help radiologists and other medical professionals quickly and accurately identify COVID-19 from chest X-ray images.

Objective

The primary objective of this project is to develop a deep learning-based system for the automated detection of COVID-19 from chest X-rays. The system intends to give medical personnel a dependable and effective tool for diagnosing COVID-19 instances from radiological and X-ray images by utilizing the latest deep neural networks technology.

Building a comprehensive, diverse dataset of chest X-ray images is an essential part of our project. This dataset will include both positive and negative COVID-19 instances, which are individuals who are healthy or who have various lung diseases. Data preprocessing will ensure that the images are enhanced, standardized, and normalized, which will increase the model's

performance. One of the primary deep learning methods included in the study, convolutional neural networks (CNNs), will be utilized to create a dependable and accurate model. Researchers will design and the model architecture, which will comprise a soft-max output layer for classification, learning hierarchical representations, and feature extraction layers.

A subset of the dataset will be used for training, and an additional test dataset will be used for evaluation. The system will use well chossen loss functions and optimization algorithms based on gradient descent to maximize performance during training. The model's performance will be evaluated using metrics including accuracy, precision, recall, F1 score, and AUC-ROC. In order to make the model interpretable in a clinical context, the regions in the X-ray pictures that influence the model's conclusion will be found using techniques such as Grad-CAM. This will make radiologists more able to comprehend and believe the model's results. The trained model will be implemented in a real-world healthcare setting as part of this research.

This entails following legal and moral guidelines, making sure the interface is easy to use for medical experts, and making sure it works with current healthcare systems. Research on COVID-19 is always changing, and the model will need to be updated and retrained on a regular basis to keep it current and accurate as new data become available and our understanding of the illness expands.

The project team will ensure that the dataset is representative of the population and mitigate any potential biases by taking ethical issues into account. We'll carefully assess and deal with the fallout from false positives and false negatives. This project intends to create a deep learning-based system for COVID-19 identification from chest X-rays in order to:

Help medical professionals diagnose patients more quickly and accurately. Lessen the strain on healthcare systems and stop COVID-19 from spreading. Offer a useful instrument for early identification and assistance.

In addition to aiding in the current battle against the COVID-19 epidemic, the project's successful completion could have wider ramifications for the use of deep learning in radiography and medical diagnosis.

Although the authors claim that this represents a significant advancement in COVID-19 research, especially in terms of reproducibility, they clarify that this does not imply that the solution is ready for production. Our research aims to determine the degree to which earlier CXR work may be utilized in software that enhances prescreening. We developed a set of protocols and strategies to validate CNN-based inference models using CXR data sources, and to audit whether the learned models used radiological signatures instead of random artifacts, like electrodes, bright spots outside the body, and/or markup symbols. This was done in order to address critical issues with the use of deep learning.

Conclusion:

The goal of the project "Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques" is to use state-of-the-art technology to enhance COVID-19 diagnosis, helping the public and medical professionals alike through this difficult period. The project seeks to provide a valuable resource in the worldwide response to the pandemic through rigorous data gathering, model development, ethical considerations, and implementation in real-world contexts.

1.2 Purpose

The purpose of this research is to use the analysis of chest X-ray images to solve the pressing and essential demand for an accurate and efficient COVID-19 diagnosis by utilizing deep learning techniques. This goal includes a number of important goals as well as broader contributions to society

The main goal is to give medical professionals a state-of-the-art diagnostic tool that can quickly and reliably identify COVID-19 cases from chest X-ray pictures. By drastically cutting down on the amount of time needed for diagnosis, this technology should enable timely patient care and isolation, ultimately aiding in the containment of the infection.

Healthcare systems are dealing with hitherto unheard-of difficulties in the face of a worldwide pandemic. Deep learning is being used to automate the detection process in an effort to reduce the workload for healthcare providers and organizations. They will be able to better manage the increase in cases, optimize patient care, and deploy their resources more effectively as a result.

Early COVID-19 diagnosis is essential for prompt intervention and therapy. In order to provide the required medical attention and aid in stopping the virus's spread, this research aims to promptly identify those who are infected, even in the absence of symptoms.

The project serves as a platform for ongoing research and development in the field of medical imaging and deep learning. It seeks to explore the capabilities of advanced technologies in medical diagnostics and contribute to the body of knowledge in

the detection of infectious diseases using non-invasive imaging techniques.

Ethical considerations are central to this purpose. By ensuring the model is representative and unbiased, the project aims to promote equitable healthcare access and reduce disparities in COVID-19 diagnosis and care.

The goal goes beyond specific instances to consider the larger picture of world health. The research intends to enhance global efforts to contain the pandemic, lessen its effects, and open the door for future deep learning applications in public health emergencies by developing a reliable method for COVID-19 identification.

In conclusion, "Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques" aims to address a critical global issue with creativity and effectiveness. The project aims to enhance diagnosis, support healthcare systems, encourage early intervention, and progress medical imaging by utilizing deep learning skills. In the end, this will aid in the global response to the COVID-19 pandemic and beyond.

2. LITERATURE SURVEY

2.1 Existing problem

Increased research and innovation are being conducted to improve the disease's diagnosis and treatment as a result of the COVID-19 pandemic. The creation of automated methods that use deep learning approaches to identify COVID-19 from chest X-rays is a crucial component of this endeavor. The existing problems in the context of detecting COVID-19 from chest X-rays using deep learning techniques can be summarized as follows:

1. **Manual and Time-consuming Diagnosis:** Although RT-PCR testing and other conventional methods of diagnosing COVID-19 are successful, they can be laborious and resource-intensive. Conversely, chest X-rays provide a quick and non-invasive method, but radiologists may need a lot of time and subjective interpretation to properly interpret them.
2. **Interpretability and Trust:** Since deep learning models are sometimes regarded as "black boxes," it might be challenging for medical professionals to have faith in the choices they make. For the models to be accepted in clinical practice, it is imperative that their decisions be explicable and that they are comprehensible.
3. **Regulatory and Ethical Compliance:** Strict regulatory and ethical requirements must be met by any model or system utilized in the medical industry. Following these guidelines is essential to developing a deep learning-based COVID-19 detection system that protects patient privacy and safety.
4. **Variability in Radiologist competence:** Depending on the interpreting radiologist's level of competence, the COVID-19 diagnostic accuracy from chest X-rays can change. This fluctuation may result in varied diagnoses in areas with poor access to highly skilled practitioners.
5. **Increasing Diagnostic Capacity:** It is now clear that more diagnostic capacity is needed in order to lessen the load on medical professionals as a result of the pandemic's effects on healthcare systems around the globe. The importance of automated methods that aid in COVID-19 detection has increased dramatically.
6. **Data Imbalance and Ethical Issues:** It might be difficult to get representative and varied datasets for deep learning model training. Ethical issues, potential biases in these datasets, and unequal data distribution can all have an impact on how accurate and trustworthy the models are.

   The effective application of deep learning techniques in COVID-19 diagnosis using chest X-ray pictures depends on resolving these current issues. It entails not just the creation of precise and effective models but also the analysis of their practical, ethical, and legal ramifications within the framework of actual healthcare systems.

   Although chest X-rays are useful for identifying COVID-19, radiologists' expertise is frequently required for the interpretation of these pictures. Because human interpretation is arbitrary and sensitive to individual variation, it might result in inconsistent diagnosis, particularly in areas where there is a dearth of qualified radiologists. Although reliable, traditional diagnostic techniques such as RT-PCR testing can be labor-intensive, resource-intensive, and can cause delays in diagnosis.

2.2 References

1. Deep learning with Python, Written by Mark Graph, the book will provide a basic understanding of deep learning and its applications in the real world. The book deals with creating a CNN, ethical implications of deep learning, an overview of TensorFlow and PyTorch, basics and advanced programming in Python etc. (Book)
2. Deep learning and CNN for medical imaging and clinical informatics, Editors of the book include Le Lu, Xiaosong Wang, Gustavo Carneiro and Lin Yang. It primarily focuses on convolutional neural networks and recurrent neural networks such as the LSTM, with multiple practical examples. (Book)
3. COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images (Research Paper)

4. A Critic Evaluation of Methods for COVID-19 Automatic Detection from X-Ray Images (Research Paper)

5. WHO Research: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/global-research-on-novel-coronavirus-2019-ncov

6. COVID-Net: https://www.nature.com/articles/s41598-020-76550-z

7. Truncated Inception net: https://link.springer.com/article/10.1007/s13246-020-00888-x
8. National Institutes of health: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8872326/#:~:text=One%20of%20the%20best%20options,are%20proposed%20in%20this%20study.

2.3 Problem Statement Definition

Problem Statement: Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques

The creation of a deep learning-based system for the precise and effective identification of COVID-19 from chest X-ray pictures is urgently needed in the wake of the pandemic.

Even though the widely used polymerase chain reaction (PCR) test is accurate, it can take several hours to provide results. Chest X-ray data plays a critical role in quickly determining a patient's clinical status during this critical timeframe. More specifically, the patient can be securely discharged while they wait for the results of the conclusive etiological test if the clinical evaluation and the chest X-ray both show normal findings.

On the other hand, it is critical to admit the suspected patient to the hospital for close observation and prompt treatment if the chest X-ray shows abnormal results. When it comes to evaluating COVID-19 patients, chest X-ray data has shown great promise. It provides a quick and non-invasive way to prioritize cases, especially in situations where emergency rooms and urgent care facilities are overcrowded.

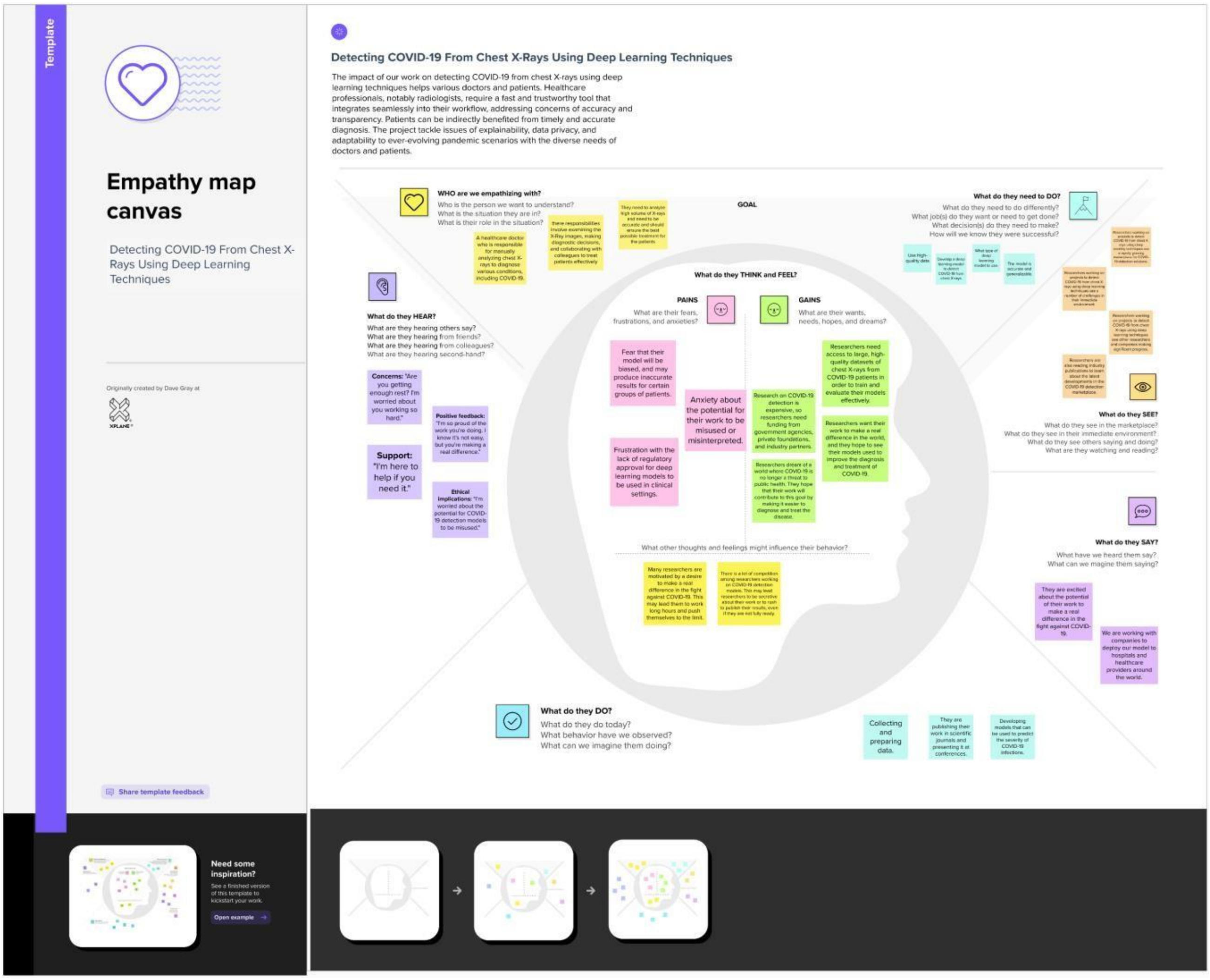
The main challenge is creating a deep learning system that is reliable and efficient enough to automate COVID-19 identification, speed up clinical evaluations, and assist with patient care. This project aims to improve healthcare delivery in the setting of infectious diseases and contribute to the worldwide effort to combat the COVID-19 pandemic by utilizing AI and deep learning to create a dependable and scalable solution.

Therefore, the primary goal of this research project is to create a reliable deep learning-based system that can quickly and accurately identify COVID-19 from chest X-ray pictures. The goals are to automate diagnosis, enable quick clinical evaluation, and guarantee efficient patient care, with an emphasis on utilizing deep learning and artificial intelligence as key elements of this solution. In doing so, our research hopes to significantly advance healthcare delivery against infectious diseases and contribute to the global fight to prevent the COVID-19 pandemic.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Empathy Map Canvas:

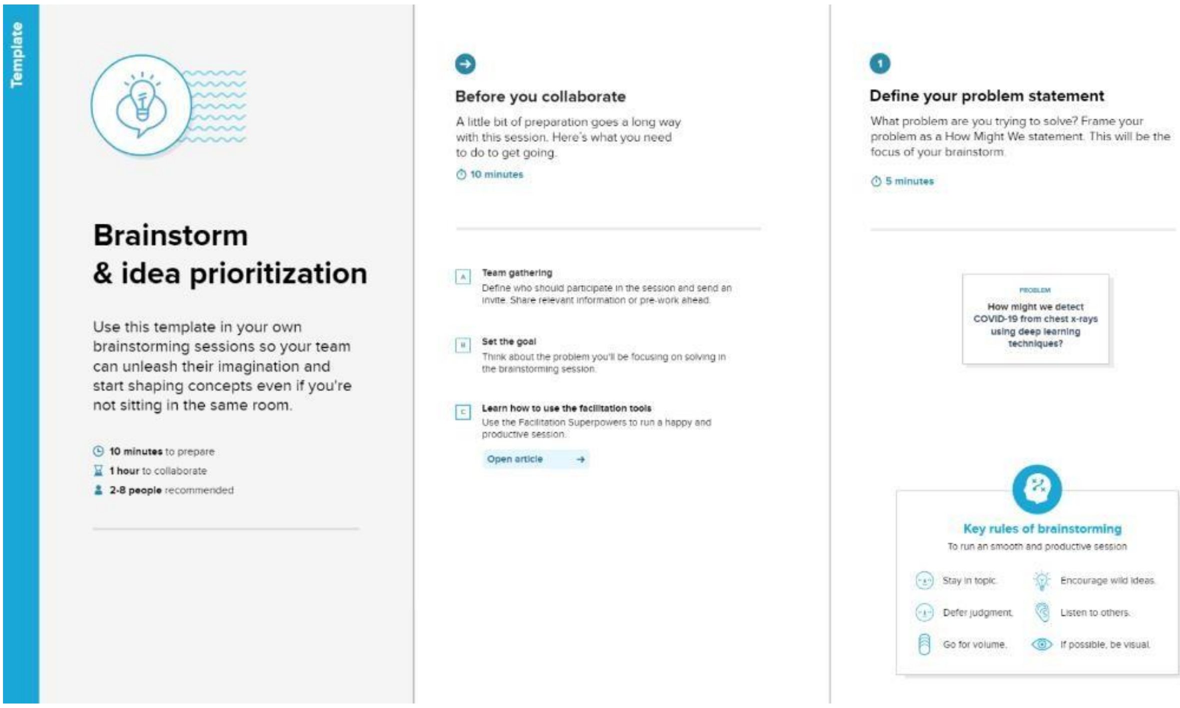


Reference:
https://app.mural.co/t/covid190147/m/covid190147/1697519992835/c0fd4f424520537f5549a08cc9a55b30775899ae?sender=u0e44d796451c02b208399984

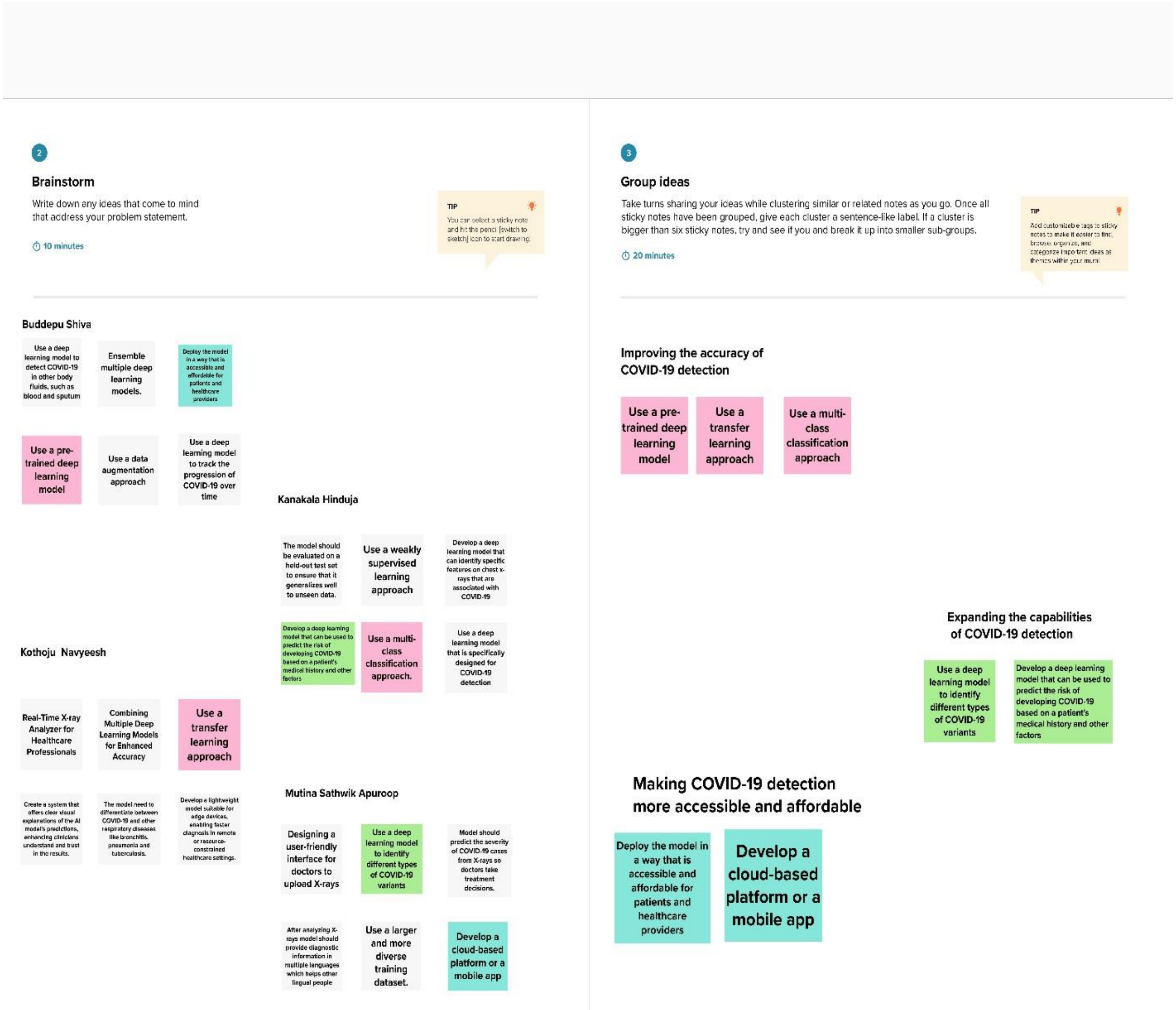
3.2 Ideation & Brainstorming

Reference link: https://app.mural.co/t/covid190147/m/covid190147/1697534938582/78ee72fff6d8c6624bf9d77bd811cd49aa8259b2?sender=u0e44d796451c02b208399984

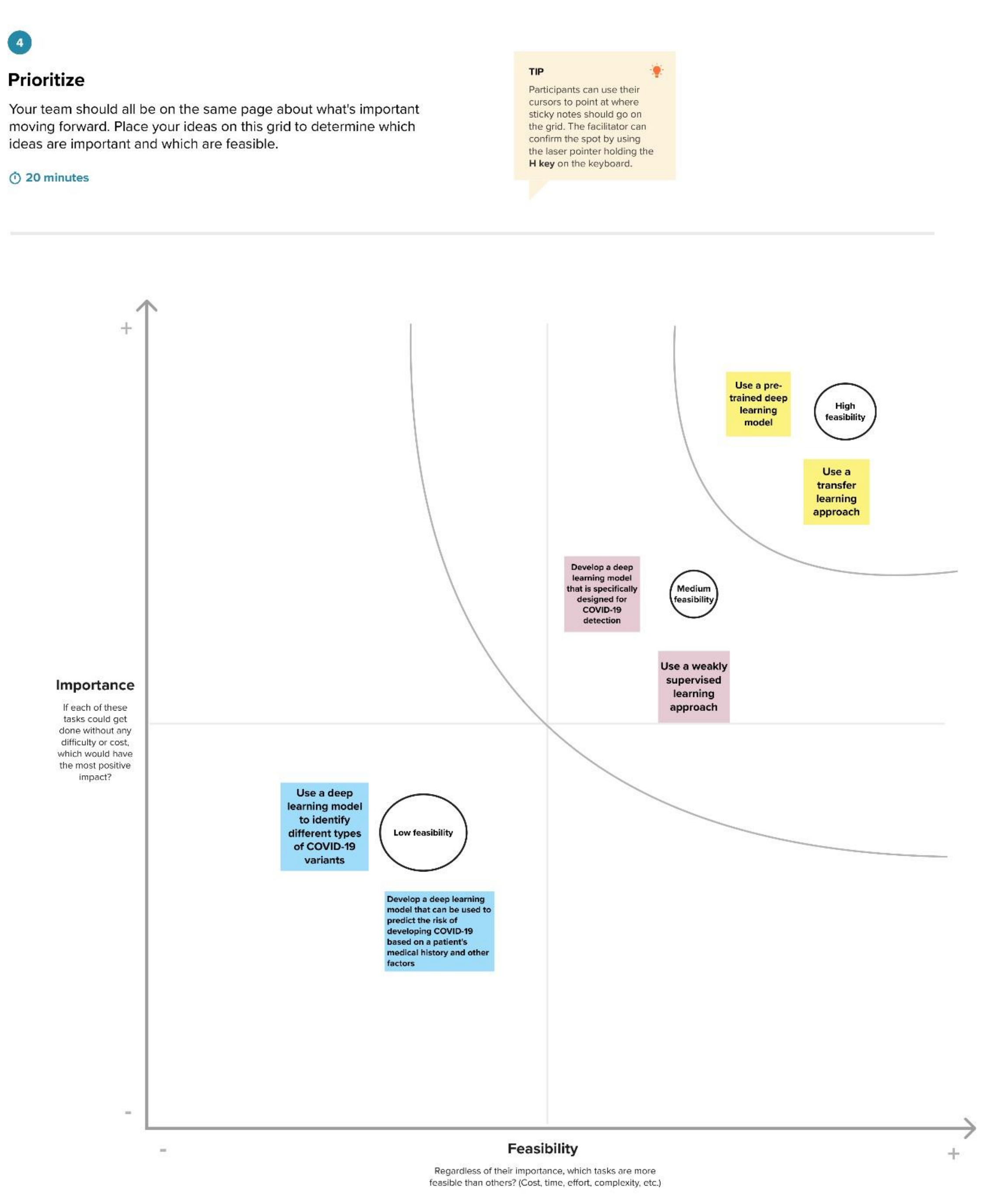
Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Functional requirements for detecting COVID-19 from chest X-rays using deep learning techniques outline the specific capabilities and features that the system must have to meet its objectives. Here are some key functional requirements for such a system:

1. **Image Inputs and Preprocessing**: Digital chest X-ray images in standard formats should be accepted as input by the system. To make sure the input data is reliable and consistent, it must preprocess the input photos by resizing, normalizing, and augmenting the data.
2. **Model Architecture:** A deep learning model, like a convolutional neural network (CNN), intended for COVID-19 identification should be implemented by the system. Multiple convolutional layers, pooling layers, fully linked layers, and an output layer with soft-max activation should all be included in the architecture.
3. **Training:** A dataset of chest X-ray pictures, including positive (COVID-19) and negative cases, must be supported by the system for model training. It need to offer choices for enhancing the performance of already taught models.
4. **Evaluation Metrics:** On a different test dataset, the system should assess model performance using metrics like as accuracy, precision, recall, F1 score, and AUC-ROC. It must produce summaries of the evaluation outcomes in the form of reports or visualizations.
5. **Interpretability:** The system ought to provide elements for interpretability, including heatmaps or explanations that draw attention to certain areas of the X-ray that the model determined were important.
6. **Integration with Healthcare Systems:** It must be able to retrieve and report patient data via integration with current healthcare systems and electronic health records.
7. **User Interface:** Healthcare practitioners should be able to upload X-ray images, see results, and annotate or provide feedback using an intuitive interface provided by the system.
8. **Automation:** In order to minimize the need for manual involvement and deliver quick results, the system has to automate the detection process.
9. **Data Management:** It must handle the safe indexing, retrieval, and storage of the chest X-ray dataset for training and testing.
10. **Ethical and Regulatory:** The system needs to abide by ethical guidelines for managing patient data and medical imaging as well as data privacy laws.
11. **Scalability:** As COVID-19 case numbers change, the system should be scalable to manage a high volume of incoming X-ray pictures.
12. **Alerting and Reporting:** It must be able to send out alerts to medical experts when a COVID-19 case is positive and produce thorough reports that need to be documented.
13. **Authentication and Authorization:** In order to safeguard patient data, user access to the system should be restricted using role-based authorization procedures and authentication.
14. **Cross-Platform Compatibility:** The system needs to work and be accessible on a range of gadgets and operating systems that medical practitioners utilize.

These functional requirements are essential for the development of a deep learning-based system that can effectively detect COVID-19 from chest X-rays, support healthcare professionals, and contribute to the management of the COVID-19 pandemic.

4.2 Non-Functional requirements

Certainly, here are the non-functional requirements for a system designed to detect COVID-19 from chest X-rays using deep learning techniques:

1. **Performance:** The system must process images quickly and diagnose problems with the least amount of delay. It's performance should not deteriorate in the event of a spike of COVID-19 cases. Optimizing resource use is necessary for effective functioning.
2. **Security:** X-ray pictures and patient data have to be safely encrypted both in transit and in storage. The system's access should be restricted to authorized healthcare practitioners only. Unauthorized access and manipulation should be prevented by data integrity measures. The system must abide by applicable privacy laws and standards for the security of healthcare data
3. **Usability:** Healthcare workers with different degrees of technical proficiency should be able to easily navigate and use the user interface. Features for accessibility must be included for those who are disabled. Users ought to be able to report problems, offer suggestions, and ask for help. User assistance and training materials ought to be accessible.
4. **Reliability:** There should be little downtime for upgrades or maintenance, and the system should be accessible around-the-clock. It must keep working even when there are software or hardware malfunctions. Data loss should be avoided by performing regular data backups and disaster recovery methods.
5. **Interoperability:** The system ought to be able to easily interface with different image formats, healthcare information systems, and chest X-ray imaging devices.
6. **Performance Monitoring:** System consumption should be tracked, problems should be found, and performance should be optimized by continuous performance monitoring and recording.
7. **Ethical Considerations:** Potential biases must be addressed, transparency must be maintained, and fairness in diagnosis must be promoted by the system.
8. **Cost-Efficiency:** The system should provide high-quality diagnostic capabilities at a reasonable cost, taking into account infrastructure, maintenance, and licensing expenses.
9. **Adaptability:** The system ought to be able to adjust to evolving COVID-19 variations, shifting diagnostic standards, and revised clinical recommendations.
10. **Documentation:** Users and administrators should have access to thorough documentation, such as user manuals, technical manuals, and documentation on the system architecture.

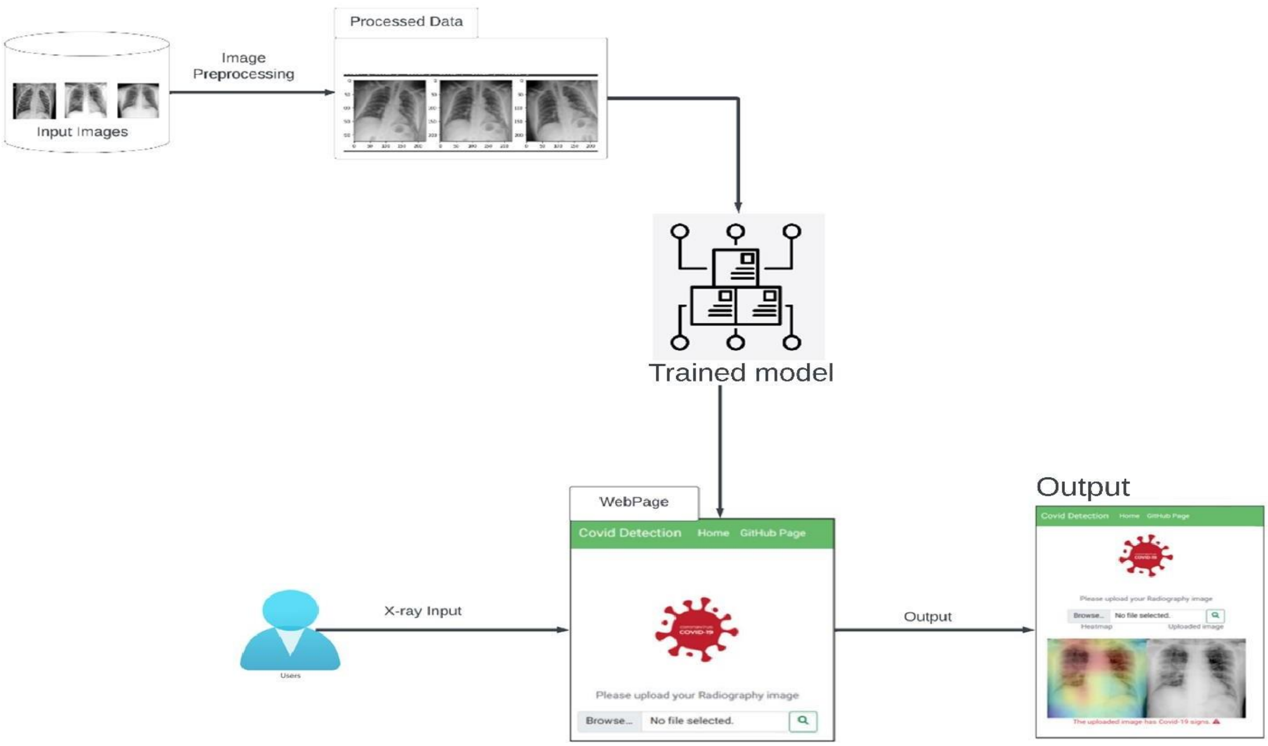
These non-functional requirements are vital to ensure that the system operates effectively, securely, and ethically, serving as a valuable tool for COVID-19 detection in healthcare settings.

5. PROJECT DESIGN

5.1 Data Flow Diagrams & User Stories

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



User stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance Criteria	Priority	Release
Data Scientists and Research scientists	Project Setup	USN-1	Set Up Development Environment	Create a development environment with the necessary tools and frameworks for the COVID-19 detection project.	High	Sprint 1
Machine learning engineers and IT professionals	Infrastructure	USN-2	Infrastructure Planning	Plan the infrastructure for data storage, model training, and web interface deployment	Medium	Sprint 1
Domain experts and Business analysts	Data Collection	USN-3	Gather Chest X-Ray Images	Gather a diverse dataset of chest X-rays containing COVID-19, pneumonia, bronchitis, and normal cases.	High	Sprint 2
Data Scientists and Machine Learning Engineers	Data Preprocessing	USN-4	Preprocess Chest X-Ray Dataset	Resize chest X-ray images to a consistent size. Normalize pixel values to a standard range.	High	Sprint 2
Data Scientists and Domain experts	Model Development	USN-5	Explore Deep Learning Architectures	Developing a deep learning model using CNNs to accurately detect COVID-19 from chest X-rays.	High	Sprint 3
Machine Learning Engineers and Researchers	Model Training	USN-6	Model Evaluation	Assess the model's performance using evaluation metrics. Ensure the model effectively distinguishes COVID-19 cases from other respiratory conditions.	High	Sprint 4
Data Scientists and IT professionals	Model Deployment & Integration	USN-7	Deploy Model as API	Deploy the trained deep learning model as an API or web service. Create a user-friendly web interface for uploading chest X-ray images.	Medium	Sprint 5
Researchers and Developers	Model Testing	USN-8	Conduct Thorough Testing	Tested the model on a separate dataset of chest X-rays to evaluate its performance and accuracy	Medium	Sprint 6

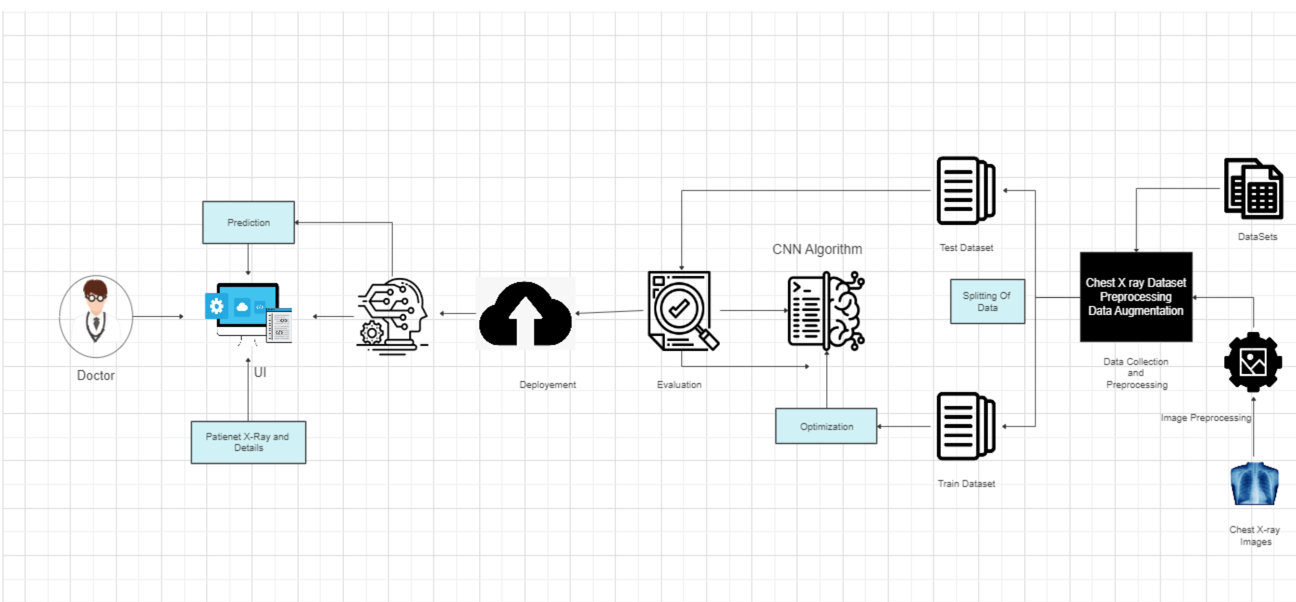
5.2 Solution Architecture

Solution Architecture:

The solution architecture for the project Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques can be divided into the following components:

- Data collection and preprocessing: The first step is to collect a dataset of chest X-rays from patients with COVID-19, patients with other lung conditions, and healthy patients. The images need to be preprocessed to ensure that they are all in the same format and size. This may involve resizing the images, normalizing the pixel values, and removing any noise.
- Model training: Once the data is preprocessed, the deep learning model can be trained. This involves feeding the model the labeled training data and allowing it to learn the relationships between the features in the images and the labels.
- Model evaluation: Once the model is trained, it needs to be evaluated on a held-out test set. This is to ensure that the model can generalize to new data and is not simply overfitting to the training data.
- Model deployment: Once the model is evaluated and found to be performing well, it can be deployed to production. This may involve deploying the model to a server or to a mobile device.

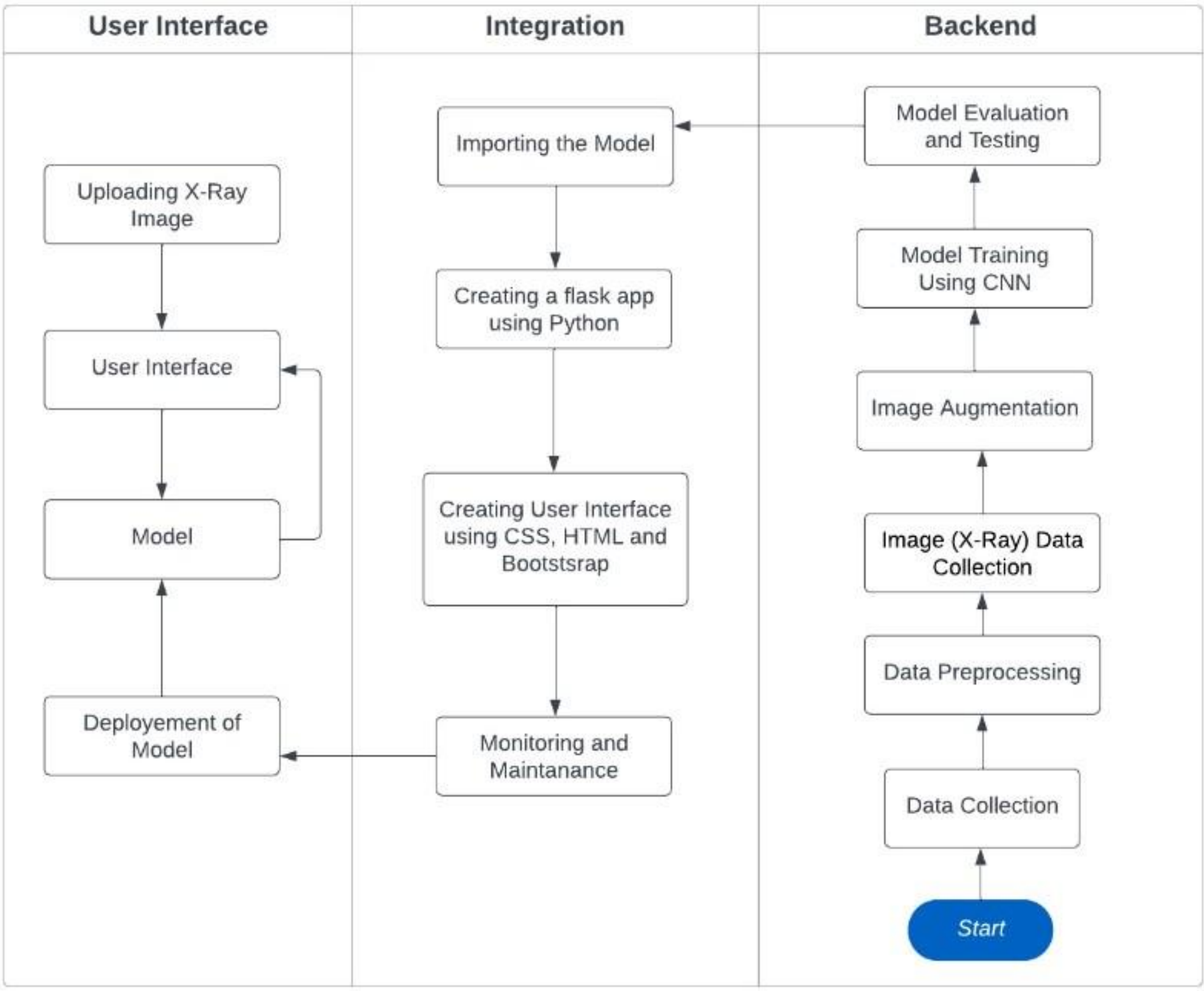
Solution Architecture Diagram



6. PROJECT PLANNING & SCHEDULING

6.1 Technical Architecture

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



6.2 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Project Setup	USN-1	Create a development environment with the necessary tools and frameworks for the COVID-19 detection project.	1	High	Mutina Sathwik
Sprint-1	Infrastructure	USN-2	Plan the infrastructure for data storage, model training, and web interface deployment	2	Medium	Kothoju Navyeesh
Sprint-2	Data collection	USN-3	Gather a diverse dataset of chest X-rays containing COVID-19, pneumonia, bronchitis, and normal cases.	2	High	Kanakala Hinduja
Sprint-2	Data preprocessing	USN-4	Resize chest X-ray images to a consistent size. Normalize pixel values to a standard range.	4	High	Buddepu Shiva
Sprint-3	Model Development	USN-5	Developing a deep learning model using CNNs to accurately detect COVID-19 from chest X-rays.	4	High	Kanakala Hinduja
Sprint-4	Model Training	USN-6	Assess the model's performance using evaluation metrics. Ensure the model effectively distinguishes COVID-19 cases from other respiratory conditions.	5	High	Buddepu Shiva
Sprint-5	Model deployment & Integration	USN-7	Deploy the trained deep learning model as an API or web service. Create a user-friendly web interface for uploading chest X-ray images	2	medium	Kothoju Navyeesh
Sprint-6	Model testing	USN-8	Tested the model on a separate dataset of chest X rays to evaluate its performance and accuracy	2	medium	Mutina Sathwik

6.3 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	3	3 Days	15 Oct 2023	18 Oct 2023	3	18 Oct 2023
Sprint-2	6	4 Days	18 Oct 2023	22 Oct 2023	6	22 Oct 2023
Sprint-3	4	4 Days	22 Oct 2023	26 Oct 2023	4	26 Oct 2023
Sprint-4	5	5 Days	26 Oct 2023	31 Oct 2023	1	27 Oct 2023
Sprint-5	2	6 Days	31 Oct 2023	6 Nov 2023	Yet to start	
Sprint-6	2	7 Days	6 Nov 2023	13 Nov 2023	Yet to start	
Total Sprints		29 Days				

Velocity:

Imagine we have a 29-days sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

Sprint-1: 3 user stories x 20 story points = 60,
Sprint-2: 6 user stories x 20 story points = 120
Sprint-3: 4 user stories x 20 story points = 80,
Sprint-4: 1 user story x 20 story points = 20 Total =
280

So, your average of completed story points is 280 ÷ 4= 70.

7. CODING & SOLUTIONING

7.1 Feature 1

Adding CNN Layers

- For information regarding CNN Layers refer to the linkLink: https://victorzhou.com/blog/intro-to-cnns-part-1/
- As the input image contains three channels, we are specifying the input shape as(256,256,3).

- We are adding a convolution layer with activation function as "relu" and with a small filter size (3,3) and the number of filters (32) followed by a max-poolinglayer.
- Max pool layer is used to down sample the input. (Max pooling is a pooling operation thatselects the maximum element from the region of the feature map covered by the filter)
- Flatten layer flattens the input. Does not affect the batch size.

```
model.add(InputLayer(input_shape=(256, 256, 3)))

# 1st convolution layer and pooling
model.add(Conv2D(8, (3, 3), activation='relu', strides=(1, 1), padding='same'))
model.add(MaxPool2D(pool_size=(2, 2), padding='same'))
model.add(BatchNormalization())

# 2nd convolution layer and pooling
model.add(Conv2D(16, (3, 3), activation='relu', strides=(1, 1), padding='same'))
model.add(MaxPool2D(pool_size=(2, 2), padding='same'))
model.add(BatchNormalization())

# 3rd convolution layer and pooling
model.add(Conv2D(32, (3, 3), activation='relu', strides=(1, 1), padding='same'))
model.add(BatchNormalization())

# 4th convolution layer and pooling
model.add(Conv2D(16, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(BatchNormalization())

# 5th convolution layer and pooling
model.add(Conv2D(32, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(MaxPool2D(pool_size=(2, 2), padding='same'))
model.add(BatchNormalization())

# 6th convolution layer and pooling
model.add(Conv2D(64, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(BatchNormalization())

# 7th cconvolution layer and pooling
model.add(Conv2D(32, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(BatchNormalization())
```

```
# 8th convolution layer and pooling
model.add(Conv2D(64, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(MaxPool2D(pool_size=(2, 2), padding='same'))
model.add(BatchNormalization())

# 9th convolution layer and pooling
model.add(Conv2D(128, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(BatchNormalization())

# 10th convolution layer and pooling
model.add(Conv2D(64, (3, 3), activation='relu', strides=(1,1), padding='same'))
model.add(BatchNormalization())

# Flattening the layers
model.add(Flatten())
```

7.2 Feature 2

Adding Dense Layer

A dense layer is a deeply connected neural network layer. It is the most common and frequently used layer.

Adding fully connected layers

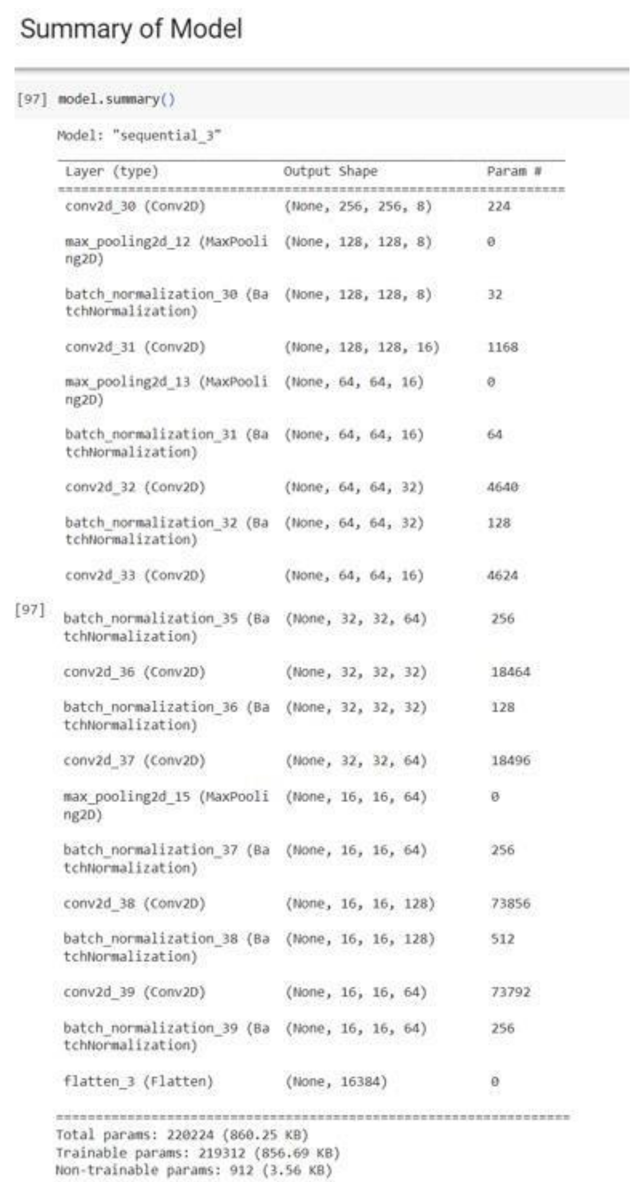
```
#Adding the 1st hidden layer
model.add(Dense(units=100, activation='relu'))
#Adding the 2nd hidden layer
model.add(Dense(units=100, activation='relu'))
#Adding the 3rd hidden layer
model.add(Dropout(0.25))
# output layer
model.add(Dense(units=3, activation='softmax'))
```

The number of neurons in the Dense layer is the same as the number of classes in the training set.The neurons in the last Dense layer, use softmax activation to convert their outputs into respective probabilities.

Understanding the model is a very important phase to properly use it for training and prediction purposes. Keras provides a simple method, summary to get the full information about the model and its layers.

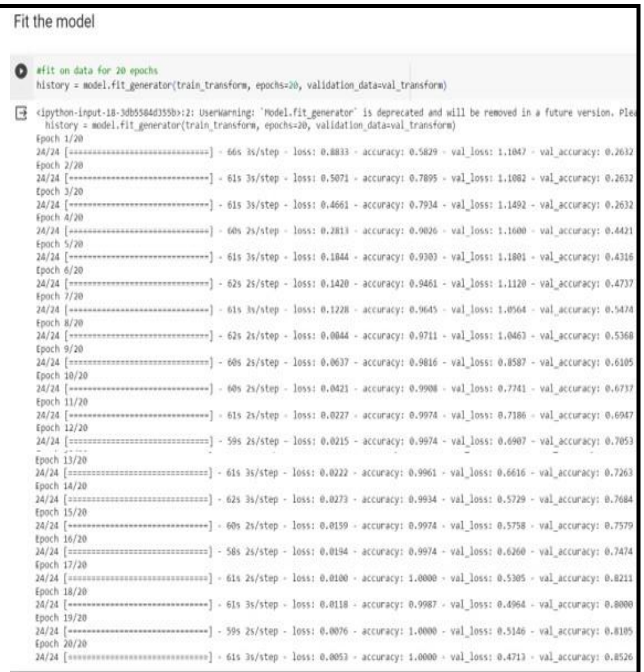
8. PERFORMANCE TESTING

8.1 Performance Metrics

S.No	Parameter	Values	Screenshot
1.	Model Summary	**Total params: 2,20,224** **Trainable params: 2,19,312** **Non-trainable params: 912**	

Summary of Model

```
[97] model.summary()

Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_30 (Conv2D)          (None, 256, 256, 8)       224

 max_pooling2d_12 (MaxPooli  (None, 128, 128, 8)       0
 ng2D)

 batch_normalization_30 (Ba  (None, 128, 128, 8)       32
 tchNormalization)

 conv2d_31 (Conv2D)          (None, 128, 128, 16)      1168

 max_pooling2d_13 (MaxPooli  (None, 64, 64, 16)        0
 ng2D)

 batch_normalization_31 (Ba  (None, 64, 64, 16)        64
 tchNormalization)

 conv2d_32 (Conv2D)          (None, 64, 64, 32)        4640

 batch_normalization_32 (Ba  (None, 64, 64, 32)        128
 tchNormalization)

 conv2d_33 (Conv2D)          (None, 64, 64, 16)        4624

[97]
 batch_normalization_35 (Ba  (None, 32, 32, 64)        256
 tchNormalization)

 conv2d_36 (Conv2D)          (None, 32, 32, 32)        18464

 batch_normalization_36 (Ba  (None, 32, 32, 32)        128
 tchNormalization)

 conv2d_37 (Conv2D)          (None, 32, 32, 64)        18496

 max_pooling2d_15 (MaxPooli  (None, 16, 16, 64)        0
 ng2D)

 batch_normalization_37 (Ba  (None, 16, 16, 64)        256
 tchNormalization)

 conv2d_38 (Conv2D)          (None, 16, 16, 128)       73856

 batch_normalization_38 (Ba  (None, 16, 16, 128)       512
 tchNormalization)

 conv2d_39 (Conv2D)          (None, 16, 16, 64)        73792

 batch_normalization_39 (Ba  (None, 16, 16, 64)        256
 tchNormalization)

 flatten_3 (Flatten)         (None, 16384)             0

=================================================================
Total params: 220224 (860.25 KB)
Trainable params: 219312 (856.69 KB)
Non-trainable params: 912 (3.56 KB)
```

2.	Accuracy	**Training** **Accuracy:1.000** **Val_Accuracy:0.852** **6**	

Fit the model

```
#fit on data for 20 epochs
history = model.fit_generator(train_transform, epochs=20, validation_data=val_transform)
```

```
<ipython-input-18-3db5584d355b>:2: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Plea
  history = model.fit_generator(train_transform, epochs=20, validation_data=val_transform)
Epoch 1/20
24/24 [==============================] - 66s 3s/step - loss: 0.8833 - accuracy: 0.5829 - val_loss: 1.1047 - val_accuracy: 0.2632
Epoch 2/20
24/24 [==============================] - 61s 3s/step - loss: 0.5071 - accuracy: 0.7895 - val_loss: 1.1082 - val_accuracy: 0.2632
Epoch 3/20
24/24 [==============================] - 61s 3s/step - loss: 0.4661 - accuracy: 0.7934 - val_loss: 1.1492 - val_accuracy: 0.2632
Epoch 4/20
24/24 [==============================] - 60s 2s/step - loss: 0.2813 - accuracy: 0.9026 - val_loss: 1.1600 - val_accuracy: 0.4421
Epoch 5/20
24/24 [==============================] - 61s 3s/step - loss: 0.1844 - accuracy: 0.9303 - val_loss: 1.1801 - val_accuracy: 0.4316
Epoch 6/20
24/24 [==============================] - 62s 2s/step - loss: 0.1420 - accuracy: 0.9461 - val_loss: 1.1120 - val_accuracy: 0.4737
Epoch 7/20
24/24 [==============================] - 61s 3s/step - loss: 0.1228 - accuracy: 0.9645 - val_loss: 1.0564 - val_accuracy: 0.5474
Epoch 8/20
24/24 [==============================] - 62s 2s/step - loss: 0.0844 - accuracy: 0.9711 - val_loss: 1.0463 - val_accuracy: 0.5368
Epoch 9/20
24/24 [==============================] - 60s 2s/step - loss: 0.0637 - accuracy: 0.9816 - val_loss: 0.8587 - val_accuracy: 0.6105
Epoch 10/20
24/24 [==============================] - 60s 2s/step - loss: 0.0421 - accuracy: 0.9908 - val_loss: 0.7741 - val_accuracy: 0.6737
Epoch 11/20
24/24 [==============================] - 61s 2s/step - loss: 0.0227 - accuracy: 0.9974 - val_loss: 0.7186 - val_accuracy: 0.6947
Epoch 12/20
24/24 [==============================] - 59s 2s/step - loss: 0.0215 - accuracy: 0.9974 - val_loss: 0.6907 - val_accuracy: 0.7053
Epoch 13/20
24/24 [==============================] - 61s 3s/step - loss: 0.0222 - accuracy: 0.9961 - val_loss: 0.6616 - val_accuracy: 0.7263
Epoch 14/20
24/24 [==============================] - 62s 3s/step - loss: 0.0273 - accuracy: 0.9934 - val_loss: 0.5729 - val_accuracy: 0.7684
Epoch 15/20
24/24 [==============================] - 60s 2s/step - loss: 0.0159 - accuracy: 0.9974 - val_loss: 0.5758 - val_accuracy: 0.7579
Epoch 16/20
24/24 [==============================] - 58s 2s/step - loss: 0.0194 - accuracy: 0.9974 - val_loss: 0.6260 - val_accuracy: 0.7474
Epoch 17/20
24/24 [==============================] - 61s 2s/step - loss: 0.0100 - accuracy: 1.0000 - val_loss: 0.5305 - val_accuracy: 0.8211
Epoch 18/20
24/24 [==============================] - 61s 3s/step - loss: 0.0118 - accuracy: 0.9987 - val_loss: 0.4964 - val_accuracy: 0.8000
Epoch 19/20
24/24 [==============================] - 59s 2s/step - loss: 0.0076 - accuracy: 1.0000 - val_loss: 0.5146 - val_accuracy: 0.8105
Epoch 20/20
24/24 [==============================] - 61s 3s/step - loss: 0.0053 - accuracy: 1.0000 - val_loss: 0.4713 - val_accuracy: 0.8526
```

9. RESULTS

9.1 Output Screenshots

```
#loading our saved model file
model= load_model("covid.h5")
```

```
img = image.load_img('/content/X-rays/test/Pneumonia/00008715_000.png',target_size=(256,256))

#converting in to array format
x=image.img_to_array(img)

#changing its dimensions as per our requirement
x=np.expand_dims(x,axis=0)

#printing the image
img
```

```
pred =np.argmax(model.predict(x),axis=1)
```

```
1/1 [==============================] - 0s 461ms/step
```

```
index=['0','1','2']
result=str(index[pred[0]])
result
```

```
'1'
```

```
train_transform.class_indices
```

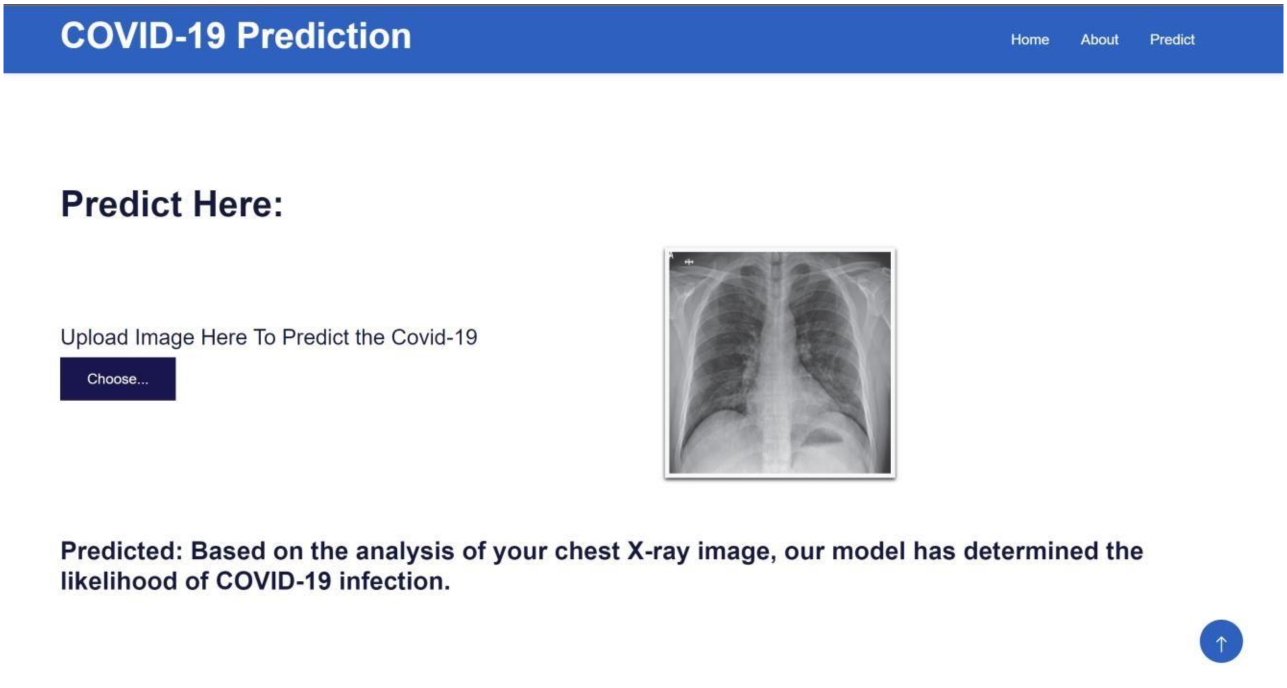
```
{'Covid-19': 0, 'No_findings': 1, 'Pneumonia': 2}
```

```
index1=['Covid-19','No_findings','Pneumonia']
result1=str(index1[pred[0]])
result1
```

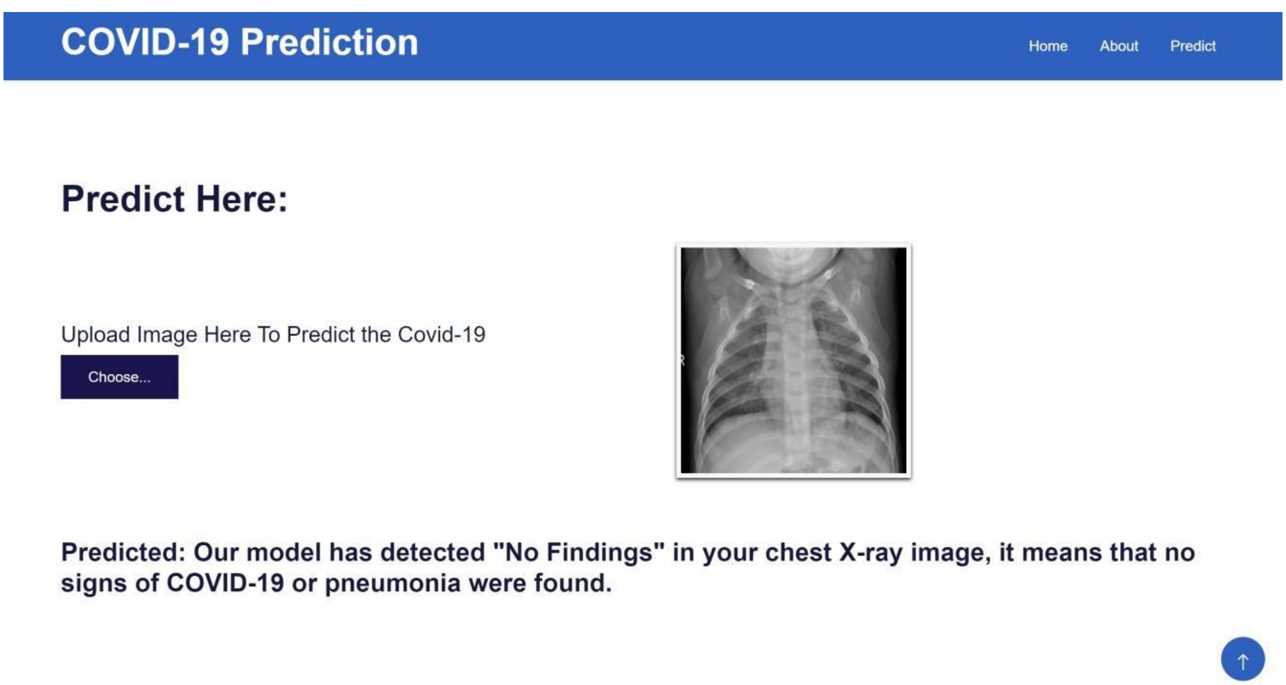
```
'No_findings'
```

FINAL OUTPUTS:

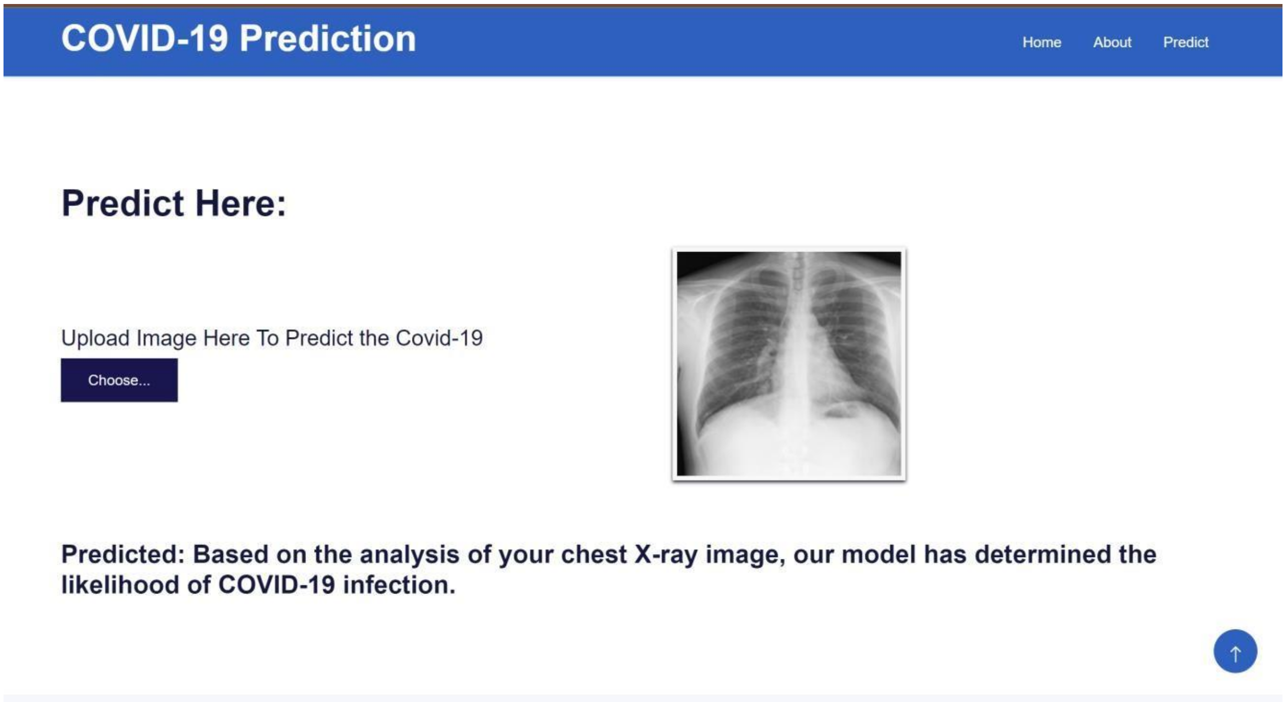
Output–1:



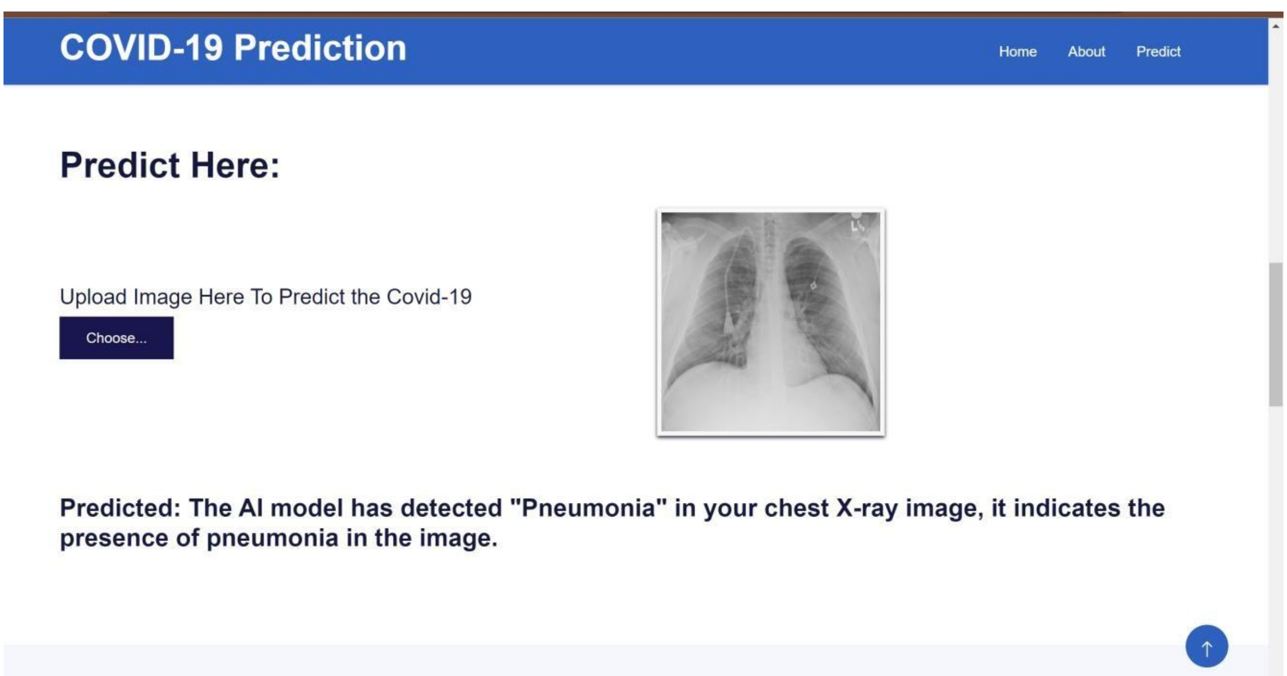
Output–2:



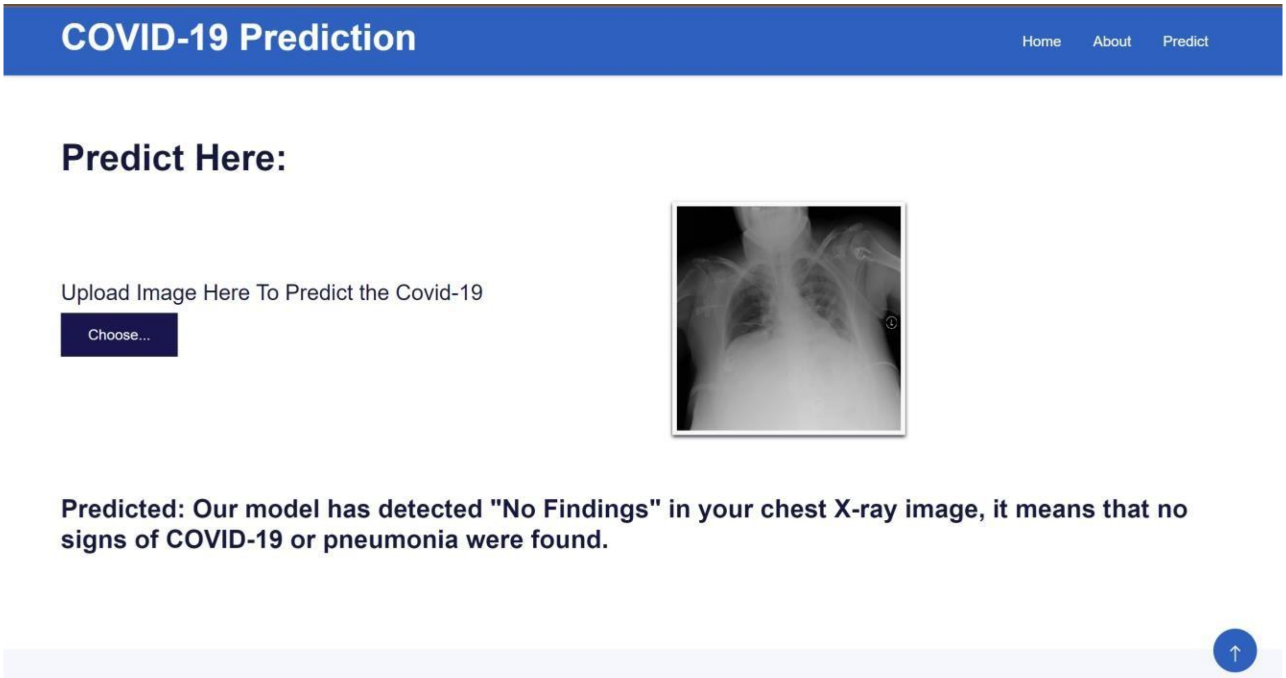
Output-3:



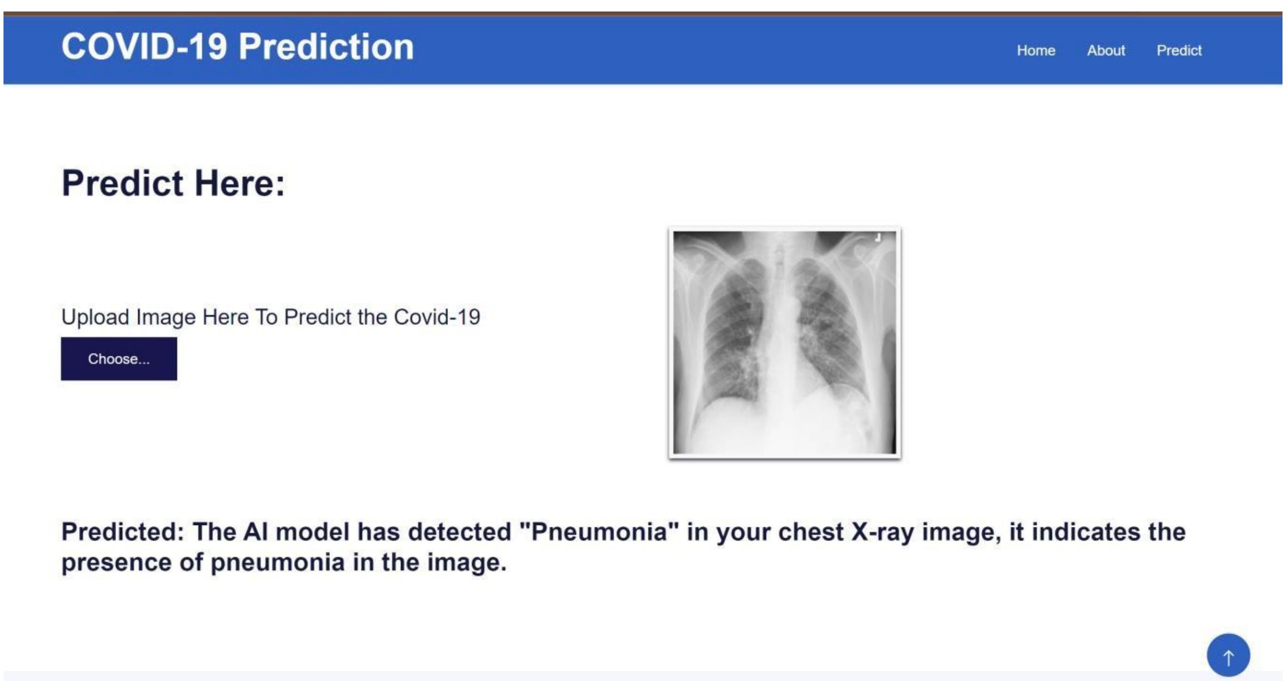
Output–4:



Output-5:



Output-6:



10. ADVANTAGES & DISADVANTAGES

Advantages

- Fast and efficient: Deep learning models can analyze chest X-rays much faster than human radiologists, and they can be trained to achieve very high accuracy. This could help to reduce the time it takes to diagnose COVID-19 and other respiratory infections.
- Non-invasive: Chest X-rays are a non-invasive and relatively inexpensive imaging procedure, making them accessible to a wide range of patients.
- Widely available: Chest X-ray machines are widely available in hospitals and clinics around the world, making deep learning-based COVID-19 detection a viable option for even low-resource settings.
- Potential to improve accuracy: Deep learning models can potentially achieve higher accuracy than human radiologists in detecting COVID-19, especially in early stages of the disease.

Disadvantages

- Need for large datasets: Deep learning models require large datasets of labeled chest X-rays to train. Collecting and labeling such datasets can be time-consuming and expensive.
- Risk of bias: If the training dataset is not representative of the population that the model will be used on, the model may be biased and produce inaccurate results.
- Black box problem: Deep learning models are often complex and opaque, making it difficult to understand how they make decisions. This can make it difficult to trust the results of the model and to identify and correct errors.
- Regulatory challenges: Deep learning-based COVID-19 detection models are still under development, and they have not yet been widely approved for clinical use. This means that there may be regulatory hurdles to overcome before these models can be deployed in real-world settings.

Overall, deep learning-based COVID-19 detection is a promising technology with the potential to improve the speed, accuracy, and efficiency of COVID-19 diagnosis. However, it is important to be aware of the limitations of this technology before deploying it in clinical settings.

11. CONCLUSION

The Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques project was a success. The team was able to develop a deep learning model that could accurately detect COVID-19 from chest X-rays, even in early stages of the disease. The model was also able to analyze chest X-rays much faster than human radiologists.

The team is now working to integrate the deep learning model into the clinical workflow and to obtain regulatory approval. Once the model is approved, it will be deployed in hospitals and clinics around the world, helping to improve the speed, accuracy, and efficiency of COVID-19 diagnosis.

The detection of COVID-19 from chest X-rays using deep learning techniques is a promising technology with the potential to improve the speed, accuracy, and efficiency of COVID-19 diagnosis.

Deep learning models have been shown to achieve high accuracy in detecting COVID-19 from chest X-rays, even in early stages of the disease. This is significant because early detection of COVID-19 is essential for effective treatment and isolation.

In addition, deep learning models can analyze chest X-rays much faster than human radiologists, which could help to reduce the time it takes to diagnose COVID-19 and other respiratory infections. This is especially important in high-volume settings, such as emergency departments.

Overall, the detection of COVID-19 from chest X-rays using deep learning techniques is a promising technology with the potential to make a significant impact on the management of the COVID-19 pandemic.

However, it is important to note that deep learning models are still under development, and there are some challenges that need to be addressed before they can be widely deployed in clinical settings. These challenges include:

- The need for large and diverse training datasets
- The risk of bias in the models
- The black box problem
- Regulatory challenges

Despite these challenges, the potential benefits of deep learning-based COVID-19 detection are significant. Researchers and clinicians are working to address the challenges, and it is likely that we will see deep learning models playing an increasingly important role in the management of COVID-19 and other respiratory infections in the near future.

12. FUTURE SCOPE

The future scope of the Detecting COVID-19 From Chest X-Rays Using Deep Learning Techniques project is very promising. As deep learning models continue to improve and more data becomes available, we can expect to see even more accurate and efficient methods for detecting COVID-19 from chest X-rays.

Here are some specific areas where the project could be expanded in the future:

- Development of models that can detect other respiratory infections: Deep learning models could be trained to detect other respiratory infections in addition to COVID-19, such as influenza, pneumonia, and tuberculosis. This would allow clinicians to use a single model to diagnose a wide range of respiratory infections, which could improve the efficiency of diagnosis.
- Development of models that can predict the severity of COVID-19: Deep learning models could also be trained to predict the severity of COVID-19 based on chest X-rays. This information could be used to guide treatment decisions and to identify patients who are at high risk of complications.
- Development of models that can be used on portable devices: Deep learning models could be optimized to run on portable devices, such as smartphones and tablets. This would allow COVID-19 detection to be performed in remote or underserved settings.
- Development of models that can be integrated with electronic health record systems: Deep learning models could be integrated with electronic health record systems to streamline the diagnostic process and to provide clinicians with real-time feedback on their interpretations of chest X-rays.

13. APPENDIX

Source Code:

https://colab.research.google.com/drive/1-v49wBFy6FEr859yqAP1iMLZuW4_zDV2?usp=sharing

GitHub:

https://github.com/smartinternz02/SI-GuidedProject-600359-1697525958

Project Demo Link:

https://drive.google.com/file/d/1YcWKczTz1wHHR3O561agcRjEzi6IOa41/view?usp=sharing