**A Project Report**

on

**“Covid Prediction App Based on Symptoms"**

****

Submitted in partial fulfilment of the requirements of the degree Of

**BACHELOR OF TECHNOLOGY**

In

**Computer Engineering**

Under the guidance of

**Prof. Sunita Jalal**

**(Professor)**

Submitted By:

Arushi I.D. No. (53500)

Simran Chandna I.D. No. (53525)

Vipul Raj I.D.No.(53527)

**Department of Computer Engineering, College of Technology, Govind Ballabh Pant University of Agriculture and Technology, Pantnagar-263145, India**

******

**ACKNOWLEDGEMENT**

Every project big or small is successful largely due to the effort of a number of wonderful people who have always given their valuable advice or lent a helping hand. We sincerely appreciate the inspiration, support and guidance of all those people who have been instrumental in making the project a success.

At this juncture we feel deeply honoured in expressing our sincere gratitude to Prof. Sunita Jalal, Professor, Department of Computer Engineering, for bestowing confidence in us and entrusting our project entitled "Covid-19 Prediction Application Using Machine Learning and Android", making the resources available at the right time and providing valuable insights leading to the successful completion of our project.

We would also like to thank all the faculty members, Dr. S.D. Samantaray, Professor and Head, Prof. B.K. Singh, Prof. C.S. Negi, Prof. Rajeev Singh, Prof. P.K. Mishra and Prof. Jalaj Sharma for their critical advice and guidance.

We are also highly thankful to all the teaching personnel of the department for their never-ending support and constructive feedback for the project. Thanks are also due to the Dean, College of Technology, staff of both University and College libraries and staff of the KNSCCF, for providing us uninterrupted internet facilities.

Last but not least we place a deep sense of gratitude to our family members and friends who have been a constant source of inspiration and help during the preparation and completion of this project work.

Arushi I.D. No. (53500) ………………………………

Simran Chandna I.D. No. (53525) ………………………………

Vipul Raj I.D. No. (53500) ………………………………

******

**CERTIFICATE**

This is to certify that the project work entitled “**Covid Prediction App Based on Symptoms**” which is

submitted by:

**Name Id. No.**

Arushi 53500

Simran Chandna 53525

Vipul Raj 53527

is a record of a students work carried by them in partial fulfilment of requirements

for the award of degree of Bachelor of Technology in the Department of Computer

Engineering, College of Technology, G.B. Pant University of Agriculture and

Technology, Pantnagar.

PANTNAGAR **(Prof. Sunita Jalal)**

Date: Project Guide

**Department of Computer Engineering, College of Technology, Govind Ballabh Pant University of Agriculture and Technology, Pantnagar-263145, India**

**APPROVAL**

This project report entitled “**Covid Prediction App Based on Symptoms**” is hereby approved as a creditable study of an engineering subject, as a prerequisite to the degree for which it has been submitted.

**Committee Member:**

1. Prof. Sunita Jalal

(Project Guide)

1. Dr. S.D. Samantaray

(Head of Department)

1. Dr P.K. Mishra
2. Dr Rajeev Singh
3. Prof. B.K. Singh

**Signature of Head of Department**

**DEPARTMENT OF COMPUTER ENGINEERING**

**COLLEGE OF TECHNOLOGY**

**G.B. Pant University of Agriculture &Technology**

**Pantnagar – 263 145, INDIA**

**ABSTRACT**

COVID-19 started in the Chinese province of Hubei's Wuhan in December 2019. Since then, several waves of covid-19 have hit people all around the world. As the whole world was striving to combat the coronavirus disease (COVID-19), healthcare and health monitoring systems were struggling to confront the virus. Many cases had been observed where the COVID-19 could not be identified at a specific time. Furthermore, any effective strategy that could monitor the coronavirus state in the human body had not been established. As a result, patients of the coronavirus could not receive proper treatment when necessary. Therefore, the death toll due to COVID-19 was rising. Although, situation of covid-19 has subsided currently, precautions need to be taken in advance to keep further waves at bay. Our project proposes a systematic approach to combat the COVID-19 pandemic more efficiently by using various machine learning algorithms and comparing their accuracy and using the best outcome to predict the disease. With eight binary features, our model was able to predict the COVID-19 test outcomes with high accuracy. Our project proposes a practical solution with the help of the developed health monitoring system that can mitigate the loss done by the COVID-19. When testing resources are few, our framework can be used, among other things, to prioritise testing for COVID-19.

**TABLE OF CONTENTS**

1. **Introduction……………………………………………………………….….7**

1.1 CoronaVirus History ……………………………………………..…....7

1.2 Challenges Faced in Coronavirus Prediction…………………………………...7

1.3 Motivation………………………………………...........8

1. **Literature Review……………………………………………………….…...9**

2.1 Machine Learning Models used in Covid Prediction………………………

2.2 Android Apps for Covid Prediction

1. **Problem Specification……………………………………14**

3.1 Objectives of the project………………………...12

3.2 Contribution……………………………………………….……12

1. **Requirement Analysis……………………………..….14**

4.1 Feasibility Study……………………………………....17

4.1.1 Technical Feasibility

4.1.2 Operational Feasibility

4.1.3 Economical Feasibility

4.2 Functional and Non-Functional Requirements..……………………..............19 4.3 Hardware Requirements…………………………………................22

4.4 Software Used For Development

4.5 Parameters Used

1. **Design……………………...……………………………..…….........24**

5.1 Data Processing………………………...……………………………………...25 5.2Machine Learning Models..............................................................................25

5.3 Proposed Approach…………………………………………………………25

1. **Implementation…...……..………………………………………..................28** 6.1 Data Set Description…………………………………...31

6.2 Tools and Technology Used…………………………….

6.3 GUI Interface……………………………

1. **Results and Discussion……………………………..……………….….........37** 7.1 Evaluation Of Machine Learning Models Used for Covid Prediction……...38 7.2 Result Analysis………………………………….................43
2. **Summary……………………………..……………….…...............................45**
3. **Literature Cited……………………………..……………….…....................46**
4. **Bio-data of the students……………………………..……………….……...47**

**Appendix I…………………………………………………………………....48**

**Appendix II…………………………………………………………………..50**

**Appendix III………………………………………………………………….51**

CHAPTER 1

**INTRODUCTION**

Coronavirus disease (COVID-19) is an infectious disease caused by the SARS-CoV-2 virus. Most people infected with the virus will experience mild to moderate respiratory illness and recover without requiring special treatment. However, some will become seriously ill and require medical attention. Older people and those with underlying medical conditions like cardiovascular disease, diabetes, chronic respiratory disease, or cancer are more likely to develop serious illness. Anyone can get sick with COVID-19 and become seriously ill or die at any age. The virus can spread from an infected person’s mouth or nose in small liquid particles when they cough, sneeze, speak, sing or breathe. These particles range from larger respiratory droplets to smaller aerosols.

**1.1 CoronaVirus History**

First identified in the city of [Wuhan](https://en.wikipedia.org/wiki/Wuhan), [Hubei](https://en.wikipedia.org/wiki/Hubei), China, the [World Health Organisation](https://en.wikipedia.org/wiki/World_Health_Organization) declared the outbreak a [Public Health Emergency of International Concern](https://en.wikipedia.org/wiki/Public_Health_Emergency_of_International_Concern) on 30 January 2020, and a [pandemic](https://en.wikipedia.org/wiki/Pandemic) on 11 March 2020.

The original source of viral transmission to humans remains unclear, as does whether the virus became [pathogenic](https://en.wikipedia.org/wiki/Pathogen) before or after the [spillover event](https://en.wikipedia.org/wiki/Spillover_event). Because many of the early infected were workers at the [Huanan Seafood Market](https://en.wikipedia.org/wiki/Huanan_Seafood_Market), it has been suggested that the virus might have originated from the market. However, other research indicates that visitors may have introduced the virus to the market, which then facilitated rapid expansion of the infections.A March 2021 WHO-convened report stated that human spillover via an intermediate animal host was the most likely explanation, with direct spillover from bats next most likely. Introduction through the food supply chain and the Huanan Seafood Market was considered another possible, but less likely, explanation.An analysis in November 2021, however, said that the earliest-known case had been misidentified and that the preponderance of early cases link

**1.2 Challenges Faced In CoronaVirus Prediction**

As the whole world is striving to combat the coronavirus disease (COVID-19), healthcare and health monitoring systems are struggling to confront the virus. Many cases have been observed where the COVID-19 could not be identified at a specific time. Furthermore, any effective strategy that can monitor the coronavirus state in the human body has not been established yet. As a result, patients of the coronavirus could not receive proper treatment when necessary. Therefore, the death toll due to COVID-19 is rising. This paper proposes a systematic approach to combat the COVID-19 pandemic more efficiently by using various machine learning algorithms and comparing their accuracy and using the best outcome to predict the disease machine learning (ML).With eight binary features, our model was able to predict the COVID-19 test outcomes with high accuracy. This paper proposes a practical solution with the help of the developed health monitoring system that can mitigate the loss done by the COVID-19. When testing resources are few, our framework can be used, among other things, to prioritise testing for COVID-19.

**1.3 Motivation**

With the arrival of covid-19, the need for instant medical facilities and required infrastructure in our university has been felt.We developed a recommender system to adapt to the needs of university and cope with the special requirements of the health domain and related queries as the students remain hesitant and don't get diagnosis on time.

Therefore, we worked on a self monitoring platform for people to diagnose themselves based on their symptoms so that they can get the treatment on time.

Initially we planned to build a project on predicting the disease on general symptoms but it was turning out to be very imprecise and we were advised to get specific on the disease we want to predict. We did our research and couldn’t find a better solution than making a covid prediction model based on the symptoms. We came up with this project when covid was at peak and hence felt the need of a system that could mitigate the burden on healthcare staff.

Although, covid-19 cases have subsided for now, In the third year of the coronavirus pandemic, well after the Omicron variant surge, unexpected and irregular rise in cases have been observed in many places. Thus, this model will be useful in curbing the no. of cases beforehand only to deal with further waves.

Also, this project helped us to work on and learn various aspects of machine learning and android.

CHAPTER 2

**LITERATURE REVIEW**

Human body is guarded by the immune system, but sometimes this immune system alone is not capable of preventing our body from diseases. Environmental conditions and living habits of people are the cause of many diseases that are the main reason for a huge number of deaths in the world, and diagnosing these diseases sometimes becomes challenging. We need an accurate, feasible, reliable, and robust system to diagnose diseases in time so that these can be properly treated. With the growth of medical data, many researchers are using these medical data and some machine learning algorithms to help the healthcare communities in the diagnosis of many diseases. There have been numerous studies done related to predicting the disease using different machine learning techniques and algorithms which can be used by medical institutions. Continuous growth in medical data gave us a way to extract the required information to predict the disease. Data Science and Big Data can be applied to detect various types of diseases by using past health data collected from the patient. These disease prediction models are very important to know the presence of disease. For the detection of the diseases we require machine learning techniques like supervised, semi-supervised, unsupervised learning, etc. and raw medical data. This raw data could easily be obtained from famous government hospitals. Machine learning techniques can use the raw data for the learning process and based on that learning they can predict the disease later. There are plenty of literature reviews available in Disease Prediction.

2.1 Machine Learning Models used in Covid Prediction

**1. MIN CHEN:-** MIN CHEN et al, proposed a disease prediction system in his paper where he used machine learning algorithms. In the prediction of disease, he used techniques like CNN-UDRP algorithm, CNN-MDRP algorithm, Naive Bayes, K-Nearest Neighbour, and Decision Tree. This proposed system had an accuracy of 94.8%.

**2. Sayali Ambekar:**- Sayali Ambekar et al, recommended Disease Risk Prediction and used a convolution neural network to perform the task. In his paper machine learning techniques like CNN-UDRP algorithm, Naive Bayes, and KNN algorithm are used. The system uses structured data to be trained and its accuracy reaches 82% and is achieved by using Naïve Bayes. **3. Naganna Chetty:-** Naganna Chetty et al, developed a system that gives improved results for disease prediction and used a fuzzy approach. And used techniques like KNN classifier, Fuzzy c-means clustering, and Fuzzy KNN classifier. In this paper diabetes disease and liver disorder prediction is done and the accuracy of Diabetes is 97.02% and Liver disorder is 96.13.

**4. Dhiraj Dahiwade**:- Dhiraj Dahiwade et al, designed a model for prediction of the disease using approaches of machine learning and used techniques like KNN and CNN. This paper suggests disease prediction i.e. based on the patient's symptoms. The accuracy of KNN is 95% and the accuracy of CNN is 98%.

**5. Lambodar Jena:-** Lambodar Jena et al, focused on risk prediction for chronic diseases by taking advantage of distributed machine learning classifiers and used techniques like Naive Bayes and Multilayer Perceptron. This paper tries to predict Chronic-Kidney-Disease and the accuracy of Naïve Bayes and Multilayer Perceptron is 95% and 99.7% respectively.

**6. Dhomse Kancha:-** Dhomse Kanchan B. et al, studied special disease prediction utilising principal component analysis using machine learning algorithms involving techniques like Naive Bayes classification, Decision Tree, and Support Vector Machine. The accuracy of this system is 34.89% for Diabetes and 53% for Heart disease.

Some other works on disease prediction are as follows:-

| Reference no | Contribution | Machine Learning Model used | Data Set Used | Conclusion |
| --- | --- | --- | --- | --- |

| 1  2 | Predict the disease of a human, based on the symptoms that he/she  posses.  Test set using the auROC.  Probability of being diagnosed with a COVID-19 | * K-Fold Cross-Validation * Support Vector * Classifier * Gaussian Naive Bayes Classifier * Random Forest Classifier * Gradient-Boosting Predictor trained with the LightGBM Python package. | <https://www.kaggle.com/datasets/kaushil268/disease-prediction-using-machine-learning>  Israeli Ministry of Health website. | Combined Accuracy: 100.0  Can be used to prioritise testing for COVID-19 when testing resources are limited. |
| --- | --- | --- | --- | --- |

| 3  4    5 | wrapper feature selection (i.e., GA) and a different set of classifiers to determine the existence of  COVID-19 based on laboratory findings  This paper focuses on the simulation and prediction of the COVID-19 daily positive increasing numbers  Predict the disease of a human, based on the symptoms that he/she  posses. | * CNN, * Decision trees KNN * Naïve Bayes      * SVM * KNN * DT * K-nearest neighbor * Scikit-learn * ensemble classifiers are applied to the resultant datasets | Hospital Israelita Albert Einstein at Sao Paulo, Brazil.  [www.kaggle.com](https://ieeexplore.ieee.org/document/www.kaggle.com,)  Hospital Israelita Albert Einstein, Brazil. | Excellent performance of BGA with CNN with accuracy 76%.  K-Nearest Neighbor performs better among all models in terms of accuracy of 98.34 recall with 97%.  A number of base as well as ensemble classifiers are applied to the resultant datasets for the two cases |
| --- | --- | --- | --- | --- |

| 6  7 | * The Sigmoid, SEIR model, ARIMA model and LSTM model were implemented. * This work estimated the outbreak peaks   This paper focuses on the simulation and prediction of the COVID-19 daily positive increasing numbers. The basic idea is to excavate the relationship between data of several days in a row. | * Sigmoid fitting * ARIMA Model * SEIR Model * LSTM Model   Regression Models   * Ridge Regression * Lasso Regression * Multivariate Polynomial Regression * Neural Network | John Hopkins University - JHU CSSE. [<https://systems.jhu.edu/research/public-health/ncov/>].  Three separate portions of their data are applied in experiments, including US nationwide pandemic data, US statewide pandemic data, and Twitter data. | There are 9.36 Lakh confirmed cases as on 14th July 2020. It is predicted that it crosses 2 Cr  In the first span, lasso regression wins out with the lowest MSE; lasso regression and ridge regression |
| --- | --- | --- | --- | --- |

**2.2 Android Apps for Covid Prediction**

1.<https://github.com/MikkoVihtakari/COVID-19-app>

This app illustrates how COVID-19 infection could develop in your country and why the drastic measures to fight the outbreak are justified. The model uses simple exponential maths, median estimates and ignores a whole lot of important parameters, such as reporting error, development of immunity, population density, demography, variation, and uncertainty.

Consequently, the model is not accurate but gives an idea of how the outbreak could develop during the uncontrolled start phase most European countries have been going through in March 2020.

2.<https://docs.google.com/document/d/1RmetpUbUCdfc4qwPae9flgt6SPJRV5AykVH5N1xG7KU/edit#>

The COVID Symptom Study is a unique prospective population-based study collecting daily reports of symptoms from millions of users. The smartphone app offers a guided interface to report a range of baseline demographic information and comorbidities. Hypothesised that longitudinal symptoms reported during the illness would cluster into distinct subtypes with differing clinical needs. The ROC-AUC (receiver operating characteristic – area under the curve) of need for respiratory support was 78.8%.

3.<https://www.medrxiv.org/content/10.1101/2021.06.16.21258691v3>

The app-based COVID Symptom Study was launched in Sweden in April 2020 to contribute to real-time COVID-19 surveillance. Data from 19,161 self-reported PCR tests were used to create a symptom-based model to estimate the individual probability of symptomatic COVID-19, with an AUC of 0.78 (95% CI 0.74–0.83) in an external dataset.

CHAPTER 3

**PROBLEM SPECIFICATION**

This pandemic continues to challenge medical systems worldwide in many aspects, including sharp increases in demands for hospital beds and critical shortages in medical equipment, while many healthcare workers have themselves been infected. Thus, the capacity for immediate clinical decisions and effective usage of healthcare resources is crucial. The most validated diagnosis test for COVID-19, using reverse transcriptase polymerase chain reaction (RT-PCR), leads to delayed reports when the cases are at peak. This contributes to increased infection rates and delays critical preventive measures.

**3.1 Objectives**

1. To develop a covid-19 prediction android application (using kotlin) that uses machine learning model at the backend to predict the disease on the basis of symptoms entered.
2. To develop a feasible platform with a user friendly interface.
3. To prepare a dataset and train machine learning model for the dataset

using machine learning Algorithms on the basis of symptoms entered.

**3.2 Contribution**

Since there was no proper description of symptoms declared by the WHO, based on some existing symptoms, we defined a model used to predict the disease according to the accuracy given by the model.

We did a lot of research to find the correct dataset through which we can accurately predict the disease and finally decided to move forward with the one made public by the Israeli Government. We tested the model on the data collected by our team through survey, to find the accuracy of model on different sets of data

We developed the model for predicting COVID-19 diagnosis by asking eight basic questions. Our framework can be used, among other considerations, to prioritise testing for COVID-19 when testing resources are limited. In addition, the methodology presented in this study may benefit the health system response to future epidemic waves of this disease and of other respiratory viruses in general.

**REQUIREMENTS ANALYSIS**

**4.1 Feasibility study**

The first phase of the project /software development is feasibility study. This is done to assess whether the software project is technically and economically viable. This involves the following steps:

Problem is defined.

Determination of technical and economical viability.

Study the alternative solution to attempt the problem.

Cost/benefit analysis is done.

A layout of the plan is prepared to get the project executed.

At the end of this phase, a report called Feasibility study is prepared by a group of software engineers This report determines whether the project is feasible or not.

**4.1.1 Behavioural Feasibility:**

People are inherently resistant to change and computers and mobiles have been known to facilitate changes. An estimate should be made of how strong the user is likely to move towards the development of a computerised system. There are various levels of users to ensure proper authentication and authorization and security of sensitive data of the organisation. The system working is quite easy to use and learn due to its simple but attractive interface. Users require no special training for operating the system.

**4.1.2 Economically Feasibility**

Economic analysis is most frequently used for evaluation of the effectiveness of the system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and savings that are expected from a system and compare them with cost, a decision is made to design and implement the system. This part of the feasibility study gives the economic justification of the system. The system being developed is economic. It is cost effective in the sense that it has eliminated the paper work completely. The system is also time effective because the predictions and functionalities are automated which are made as per the user requirement. The result obtained contains minimum errors and are highly accurate as the data is required.

**4.1.3 Operational Feasibility**

Automation makes our life easy. The proposed system is highly user friendly and is much easier to interact with. Only little instruction is required for the user so that he can easily operate the system.

**4.2 Functional and nonfunctional requirements**

**Functional Requirements**

Functional Requirements Specification describes what is required to meet the users' business needs. It specifies which actions the design must provide in order to benefit the system's users. These are determined by the needs, user, and task analysis of the current system. Functional requirements needed in E-Commerce are as follows:

**USER**

Functional Requirement analysis of user module:

**Login**

**Definition:** For the user to be able to use this system, he has to enter user credentials (Email Id and Password).

**Input:**Email Id and password.

**Output:** The system will state whether inputs are correct or not.

**Pre-conditions**: The user must have signed in the system and have a valid email and password. Then the system will show the main page to the valid user but if the user entered the wrong username or password then he/she will be invalid user and will see a message “Login failed” and to login again.

**Post conditions:** The user will enter the main page of the app.

**Signup**

**Definition:** For the user to be able to login in the system, he has to first signup in the system by entering details.

**Input:**Email, Password (2 times for confirmation) and Name.

**Output:** The system will confirm the two passwords entered and lead to the next page.

**Pre-conditions**: We check if the password in both fields is the same, if it is not it shows a message ‘Password did not match’.

If the password is the same it directs to the Logged In activity.

If the user is already registered ,it shows ‘Already registered’ otherwise it updates the table with new data and shows ‘User Added’.

**Post conditions:** The user will enter the main page of the app.

**Proceed**

**Definition:** For the user to be able to enter symptoms, he/she must be confirmed to login and then proceed further.

**Input:**On click.

**Output:** Next screen of options to enter symptoms or sign out.

**Pre-conditions**: Users need to have an account in cloud firestore and then must be authenticated by firebase authentication service.

**Post conditions:** Options to sign out or enter symptoms are shown.

**Sign Out**

**Definition:** The user will be logged out and the current session of logging in will be expired. Now the next user can login and use the app.

**Input:** On Click

**Output:** Main Activity of login screen is displayed for the next user to login.

**Pre-conditions**: User must be logged in.

**Post conditions:** Session is expired.

**Click here**

**Definition:** The user can enter symptoms through fragments by clicking here.

**Input:** click on this option.

**Output:** Fragment of Symptom 1, i.e; cough.

**Pre-conditions**: User must be logged in.

**Post conditions:** The user may enter value as yes or no.

**Yes:** It is a radio button that will take the value of symptom as 1.

**Input:** click on this option.

**Output:** Option will be selected.

**Pre-conditions**: User must be logged in.

**Post conditions:** It will assign the value of symptom.

**No**

**Definition:** It is a radio button that will take the value of symptom as 0 and it will be assumed that the user doesn’t have this symptom.

**Input:** click on this option.

**Output:** User must be logged in..

**Pre-conditions**: It will assign the value of symptom.

**Post conditions:** It will assign the value of symptom.

**Next**

**Definition:** The user can move to the next fragment to enter another symptom.

**Input:** On click.

**Output:** Next fragment of symptom will be displayed.

**Pre-conditions**: User must be logged in.

**Post conditions:** Fragment should be replaced by the next fragment.

**Previous**

**Definition:** The user can move to the previous fragment to enter or change the previous symptom.

**Input:** On click.

**Output:** Previous fragment of symptom will be displayed.

**Pre-conditions**: User must be logged in.

**Post conditions:** Fragment should be replaced by the previous fragment.

**Submit**

**Definition:** The user can confirm his details to get the prediction result.

**Input:** On click.

**Output:** Prediction report will be displayed.

**Pre-conditions**: User must have entered all the details.

**Post conditions:** Fragment should be replaced by the result fragment.

**Details**

**Definition:** The user can go to the next activity to know the technical details.

**Input:** On click.

**Output:** Next Activity of technical details will be displayed.

**Pre-conditions**: User must have used the model.

**Post conditions:** Fragment should be replaced by the next activity.

**4.6 Non-Functional Requirements**

Non-functional requirements are requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability, response time etc. They may specify system performance, security, availability, and other emergent properties. This means that they are often more critical than individual functional requirements. System users can usually find ways to work around a system function that doesn’t really meets their needs. However, failing to meet a non-functional requirement can mean that the whole system is unusable. Non-functional requirements needed in this ecommerce app are as follows:

**Security**

We understand that there is nothing more important than knowing that their account and details are safe. Email and Password protection occurs at the first level within the Android App. To access the app users are required to enter their registered email id and password. Without these, access to the app is denied. Also, for all this, the default service of firebase provides the functionality of creating strong passwords. Passwords with lengths less than six are not accepted. Passwords get encrypted, so no one can access the passwords of the users.

**Efficiency**

The proposed system is quite efficient as it saves the time by easily selecting the symptoms and details.

**Reliability**

CoviExpert renders time and location irrelevant, and empowers users with greater control of their accounts. The system should accurately perform operations such as predicting through model and handling details.

**Maintainability**

It is the aptitude of the system to undergo repair and evolution. The proposed system can be easily modified to correct faults, improve performance or other attributes, or adapt to a changed environment. The list of items available and their prices can also be changed easily. The hardware system or component can be easily retained in or restored to a state in which it can perform its required functions.

**Portability**

It is the usability of the same software in different environments. The pre-requirement of portability is the generalised abstraction between the application logic and the system interfaces. The proposed system fulfils the portability requirement.

**Usability**

The system is designed for a user friendly environment so that the user can perform various tasks easily and in an effective way.

**4.3Hardware requirements**

● Processor : Compatible with Android Studio

● Hard Disk : 50GB

● Memory: 4GB RAM

**4.4 Software used for development**

1. Jupyter Notebook
2. Pycharm
3. Android Studio
4. Android Phone
5. Postman
6. Firebase
7. Cloud Firestore
8. Heroku CLI
9. Python
10. Kotlin
11. XML
12. Git

**4.5 Parameters used**

1.Sex (male/female).

2.Age ≥60 years (yes/no/none)

3.Cough (0/1).

4.Fever (0/1).

5.Sore throat (0/1).

6.Shortness of breath (true/false).

7.Headache (0/1).

8.test\_indication (Known contact with an individual confirmed to have COVID-19 )(Other/Abroad/Contact with confirmed ).

CHAPTER 5

**Design**

**5.1 Data Processing**

**5.1.1 Data collection**

The proposed dataset of COVID-19 consists of 8 symptoms observed in 278848 patients in Israel which are collected by the Ministry of Health, Israel. The dataset contains initial records, on a daily basis, of all the residents who were tested for COVID-19 nationwide.Most of the variables in the dataset are in binary format. The value of a feature variable being '1' means a specific symptom is present, and inversely, the value '0' means there are no symptoms. The training-validation set consisted of records from 51,831 tested individuals (of whom 4769 were confirmed to have COVID-19),

from the period March 22th, 2020 through March 31st, 2020. The test set contained data from the subsequent week, April 1st through April 7th (47,401 tested individuals, of whom 3624 were confirmed to have COVID-19).The training-validation set was further divided to training and validation sets at a ratio of 3:1.

Second dataset is collected through a survey in our university which is used for testing purposes only to get the accuracy of the model and analyse the parameters and factors affecting them in India.

**5.1.2 Data preprocessing**

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

## 1) Get the Dataset

To create a machine learning model, the first thing we require is a dataset as a machine learning model completely works on data. The collected data for a particular problem in a proper format is known as the dataset.

## 2) Importing Libraries

In order to perform data preprocessing using Python, we need to import some predefined Python libraries. These libraries are used to perform some specific jobs. There are three specific libraries that we will use for data preprocessing, which are:

import numpy as np #to perform numerical operations

import pandas as pd #for reading csv files

import matplotlib.pyplot as plt

import seaborn as sn

from sklearn.model\_selection import train\_test\_split

sklearn: scikit learn has many classification,regression, clustering algorithms in machine learning,statistical modelling

## **Importing the Datasets**

Now we need to import the datasets which we have collected for our machine learning project. But before importing a dataset, we need to set the current directory as a working directory.

To import the dataset, we will use the read\_csv() function of the pandas library, which is used to read a csv file and perform various operations on it. Using this function, we can read a csv file locally as well as through URL.

## 4) Handling Missing data:

The next step of data preprocessing is to handle missing data in the datasets. If our dataset contains some missing data, then it may create a huge problem for our machine learning model. Hence it is necessary to handle missing values present in the dataset.

**Ways to handle missing data:**

There are mainly two ways to handle missing data, which are:

**By deleting the particular row:** The first way is used to commonly deal with null values. In this way, we just delete the specific row or column which consists of null values. But this way is not so efficient and removing data may lead to loss of information which will not give the accurate output.

**By calculating the mean:** In this way, we will calculate the mean of that column or row which contains any missing value and will put it on the place of missing value. This strategy is useful for the features which have numeric data such as age, salary, year, etc. Here, we will use this approach.

## 5) Encoding Categorical data:

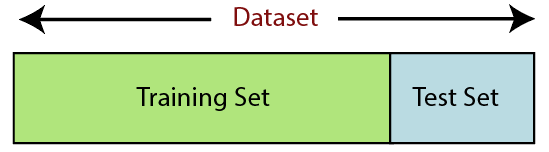
Since the machine learning model completely works on mathematics and numbers, but if our dataset would have a categorical variable, then it may create trouble while building the model. So it is necessary to encode these categorical variables into numbers.

## 6) Splitting the Dataset into the Training set and Test set

In machine learning data preprocessing, we divide our dataset into a training set and test set. This is one of the crucial steps of data preprocessing as by doing this, we can enhance the performance of our machine learning model.

Suppose, if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:



**Training Set:** A subset of dataset to train the machine learning model, and we already know the output.

**Test set:** A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

For splitting the dataset, we will use the below lines of code:

1. from sklearn.model\_selection import train\_test\_split
2. x\_train, x\_test, y\_train, y\_test= train\_test\_split(x, y, test\_size= 0.2, random\_state=0)

**5.2 Machine Learning Models Description**

According to Arthur Samuel (1959), ML is the field of study that gives computers the ability to learn without being explicitly programmed. Thus, we can define ML as the field of computer science in which machines can be designed that can program themselves.

The process of learning is simply learning from experience or observations from previous work, such as examples, or instruction, to look for patterns in data and with the help of examples, provided the system can make better decisions. The basic aim of ML is to make computers learn automatically with no human intervention and to adjust and perform actions accordingly .

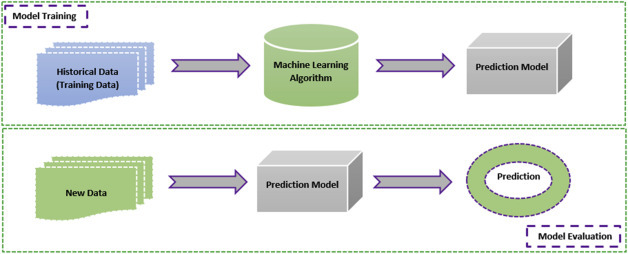


Figure shows the process of ML.

Past data are used to train the model, and then this trained model is used to test new data and then for prediction. The trained ML model's performance is evaluated using some portion of available past data (which is not present during training). This is usually referred as the validation process. In this process, the ML model is evaluated for its performance measure, such as accuracy. Accuracy describes the ML model's performance over unseen data in terms of the ratio of the number of correctly predicted features and total available features to be predicted.

ML is used in various fields, including medicine to predict disease and forecast its outcome. In medicine, the right diagnosis and the right time are the keys to successful treatment. If the treatment has a high error rate, it may cause several deaths. Therefore, researchers have started using artificial intelligence applications for medical treatment. The task is complicated because the researchers have to choose the right tool: it is a matter of life or death .

For this task, ML achieved a milestone in the field of health care. ML techniques are used to interpret and analyse large datasets and predict their output. These ML tools were used to identify the symptoms of disease and classify samples into treatment groups. ML helps hospitals to maintain administrative processes and treat infectious disease.

ML techniques were previously used to treat cancer, pneumonia, diabetes, Parkinson disease, arthritis, neuromuscular disorders, and many more diseases; they give more than 90% accurate results in prediction and forecasting.

ML techniques have been used to predict whether patients are infected by the virus based on symptoms defined by WHO and CDC.Moreover, social distancing can be monitored by ML; with the help of this approach, we can keep ourselves safe from COVID-19.

A symptom-based predictive model was proposed to predict COVID-19 based on symptoms defined by the WHO and CDC.

Because there is no proper description of symptoms declared by the WHO, based on some existing symptoms, we defined a model used to predict the disease according to the accuracy given by the model .

Algorithms used-

# **Logistic Regression**

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, True or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification.

The Logistic regression equation can be obtained from the Linear Regression equation. The mathematical steps to get Logistic Regression equations are given below:

We know the equation of the straight line can be written as:

Logistic Regression in Machine Learning

In Logistic Regression y can be between 0 and 1 only, so for this let's divide the above equation by (1-y):

Logistic Regression in Machine Learning

But we need range between -[infinity] to +[infinity], then take logarithm of the equation it will become:

Logistic Regression in Machine Learning

The above equation is the final equation for Logistic Regression.

# **Gradient Boosting**

Gradient boosting algorithm is one of the most powerful algorithms in the field of machine learning. As we know that the errors in machine learning algorithms are broadly classified into two categories i.e. Bias Error and Variance Error. As gradient boosting is one of the boosting algorithms it is used to minimise bias error of the model.

Unlike, Adaboosting algorithm, the base estimator in the gradient boosting algorithm cannot be mentioned by us. The base estimator for the Gradient Boost algorithm is fixed and i.e. *Decision Stump*. Like AdaBoost, we can tune the n\_estimator of the gradient boosting algorithm. However, if we do not mention the value of n\_estimator, the default value of n\_estimator for this algorithm is 100.

Gradient boosting algorithms can be used for predicting not only continuous target variables (as a Regressor) but also categorical target variable (as a Classifier). When it is used as a regressor, the cost function is Mean Square Error (MSE) and when it is used as a classifier then the cost function is Log loss.

**RandomForest Classifier**

A random forest is a meta estimator that fits a number of decision tree classifiers on variAs the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."

Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.ous sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

The sub-sample size is controlled with the max\_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree.

# **K-Nearest Neighbour(KNN)**

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.**Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.

## Advantages of KNN Algorithm:

* It is simple to implement.
* It is robust to the noisy training data
* It can be more effective if the training data is large.

## Disadvantages of KNN Algorithm:

* Always needs to determine the value of K which may be complex some time.
* The computation cost is high because of calculating the distance between the data points for all the training sample.

**NAIVE BAYES**

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is mainly used in *text classification* that includes a high-dimensional training dataset.Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.Some popular examples of Naïve Bayes Algorithm are spamfiltration, Sentimental analysis, and classifying articles.

## Bayes' Theorem:

Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

The formula for Bayes' theorem is given as:

Naïve Bayes Classifier Algorithm

**Where,**

**P(A|B) is Posterior probability**: Probability of hypothesis A on the observed event B.

**P(B|A) is Likelihood probability**: Probability of the evidence given that the probability of a hypothesis is true.

### Advantages of Naïve Bayes Classifier:

* Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
* It can be used for Binary as well as Multi-class Classifications.
* It performs well in Multi-class predictions as compared to the other Algorithms.
* It is the most popular choice for text classification problems.

### Disadvantages of Naïve Bayes Classifier:

* Naive Bayes assumes that all features are independent or unrelated, so it cannot learn the relationship between features.

### Applications of Naïve Bayes Classifier:

* It is used for Credit Scoring.
* It is used in medical data classification.
* It can be used in real-time predictions because Naïve Bayes Classifier is an eager learner.
* It is used in Text classification such as Spam filtering and Sentiment analysis.

**Support Vector machine**

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine.

**SVM can be of two types:**

* **Linear SVM:** Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
* **Non-linear SVM:** Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

**5.3 Proposed approach**

A symptom-based predictive model was proposed to predict COVID-19 based on symptoms defined by the WHO and CDC .Because there is no proper description of symptoms declared by the WHO, based on some existing symptoms, we defined a model used to predict the disease according to the accuracy given by the model.

Feature selection took place as part of preprocessing data. The data were divided into training data (80% of data) and test data (20% of data), usually known as the train-test split process. This split is generally done in a stratified or random manner so that population distribution in both groups consists of shuffled data, which leads to minimise bias or skewness in the data. Training data were used to train the ML classifier that we used in the model, and test data were used to test that classifier in terms of accuracy received over a predefined unseen portion of the dataset.

This dataset was divided into two sets (training set and testing set) using the test-train split method. The system was trained on the basis of training set data and the accuracy of the ML classifier, and then evaluated over the testing set. Finally, the model was used to predict the probability of infection from the disease using new patient data in terms of positive or negative.

Chapter 6

**IMPLEMENTATION**

**6.1 Dataset Description**

The COVID-19 suggested dataset is made up of 8 symptoms that were noted in 278848 Israeli patients and was gathered by the Israeli Ministry of Health.The Israeli Ministry of Health publicly released data of all individuals who were tested for SARS-CoV-2 via RT-PCR assay of a nasopharyngeal swab. During the first months of the COVID-19 pandemic in Israel, all diagnostic laboratory tests for COVID-19 were performed according to criteria determined by the Israeli Ministry of Health.

While subject to change, the criteria implemented during the study period included the presence and severity of clinical symptoms, possible exposure to individuals confirmed to have COVID-19, certain geographical areas, and the risk of complications if infected.

Except for a small minority who were tested under surveys among healthcare workers, all the individuals tested had indications for testing. Thus, there was no apparent referral bias regarding the vast majority of the subjects in the dataset used in this study; this contrasts with previous studies, for which such bias was a drawback. In addition, all negative and positive COVID-19 cases in this dataset were confirmed via RT-PCR assay.

The dataset's symptoms include cough, fever, sore throat, shortness of breath, headache, corona result, age 60 and up, gender, and test indication. The corona result tells whether or not people may have the coronavirus in their bodies. The dataset's majority of variables are in binary format. If a feature variable's value is "1," it signifies that a certain symptom is present; if it is "0," there is no symptom.

The records from 51,831 tested people, of whom 4769 had COVID-19 confirmed, were included in the training-validation set. These tests took place between March 22 and March 31, 2020. The data in the test set covered the following week, from April 1 through April 7. (47,401 tested individuals, of whom 3624 were confirmed to have COVID-19). The training-validation set was then divided further into training and validation sets at a 3:1 ratio.

**6.2 Tools and Technology**

**Android studio** is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (E-ADT) as the primary IDE for native Android application development. Android Studio provides the fastest tools for building apps on every type of Android device. The following features are provided in the current stable version:

1. Gradle-based build support
2. Android-specific refactoring and quick fixes
3. Lint tools to catch performance, usability, version compatibility and other problems
4. Template-based wizards to create common Android designs and components
5. A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations
6. Support for building Android Wear apps
7. Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine[18]
8. Android Virtual Device (Emulator) to run and debug apps in the Android studio.

**Kotlin** is an open-source, modern, statically typed language for Java Virtual Machine (JVM), and a pragmatic programming language that lets developers use both object-oriented and functional programming techniques. It is concise, safe, and fully interoperable with Java. Kotlin is now an official language for Android development and the community of Kotlin developers is growing rapidly, with major companies like Netflix, Pinterest, and Google using Kotlin. We have used kotlin version 1.4.21 for our project. The reasons why we preferred Kotlin are to apply hands on this new technology which is way more concise than java because of its data classes, smart casts, type interface & properties and more reliability with fewer bugs and crashes on the UX side. Kotlin is an open-source, modern programming language that lets developers use both object-oriented and functional programming techniques. It is concise, safe, and fully interoperable with Java. Kotlin is now an official language for Android development and the community of Kotlin developers is growing rapidly, with major companies like Netflix, Pinterest, and Google using Kotlin. Nullability issues have been one of Java's well-known sore points. Since it's a common thing in Android for the absence of certain values to be represented as “null,” Kotlin comes to address these issues by placing null right in its type system. This functionality of kotlin has saved us many times while making our project. Some instances where we used this functionality are:

In Kotlin, the type system distinguishes between references that can hold null (nullable references) and those that cannot (non-null references). For example, a regular variable of type String cannot hold null: To allow nulls, we can declare a variable as a nullable string, written String?

if(item?.itemId==R.id.item\_confirm){ var U="http://192.168.29.155/Salesweb/confirm\_order.php?mobile="+UserInfo.mobile

The not-null assertion operator (!!) converts any value to a non-null type and throws an exception if the value is null. We can write b!!, and this will return a non-null value of b.

var v= inflater!!.inflate(R.layout.fragment\_qty, container, false)

**Java Development Kit** is required for Kotlin development on the JVM. Kotlin works with JDK 1.6+. However the JDK isn't always required. Kotlin Native and Kotlin JS do not require the JDK, because the targets they compile to isn't on the JVM. We have used JDK-7 in our project.

**Python** is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public Licence (GPL).

* Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
* Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
* Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
* Python is a Beginner's Language − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games
* It supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* It supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

**The Jupyter Notebook App** is a server-client application that allows editing and running [notebook documents](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#notebook-document) via a web browser. The Jupyter Notebook App can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet.

In addition to displaying/editing/running notebook documents, the Jupyter Notebook App has a “Dashboard” ([Notebook Dashboard](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#dashboard)), a “control panel” showing local files and allowing to open notebook documents or shutting down their [kernels](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html#kernel).

The notebook extends the console-based approach to interactive computing in a qualitatively new direction, providing a web-based application suitable for capturing the whole computation process: developing, documenting, and executing code, as well as communicating the results. The Jupyter notebook combines two components:

**A web application**: a browser-based tool for interactive authoring of documents which combine explanatory text, mathematics, computations and their rich media output.

**Notebook documents**: a representation of all content visible in the web application, including inputs and outputs of the computations, explanatory text, mathematics, images, and rich media representations of objects.

**Scikit-learn** is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction.

### Components of scikit-learn:

Scikit-learn comes loaded with a lot of features. Here are a few of them to help you understand the spread:

* Supervised learning algorithms: Think of any supervised machine learning algorithm you might have heard about and there is a very high chance that it is part of scikit-learn. Starting from Generalised linear models (e.g Linear Regression), Support Vector Machines (SVM), Decision Trees to Bayesian methods – all of them are part of scikit-learn toolbox. The spread of machine learning algorithms is one of the big reasons for the high usage of scikit-learn. I started using scikit to solve supervised learning problems and would recommend that to people new to scikit / machine learning as well.
* Cross-validation: There are various methods to check the accuracy of supervised models on unseen data using sklearn.
* Unsupervised learning algorithms: Again there is a large spread of machine learning algorithms in the offering – starting from clustering, factor analysis, principal component analysis to unsupervised neural networks.
* Various toy datasets: This came in handy while learning scikit-learn. I had learned SAS using various academic datasets (e.g. IRIS dataset, Boston House prices dataset). Having them handy while learning a new library helped a lot.
* Feature extraction: Scikit-learn for extracting features from images and text (e.g. Bag of words)

**NumPy** is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It is open-source software. It contains various features including these important ones:

* A powerful N-dimensional array object
* Sophisticated (broadcasting) functions
* Tools for integrating C/C++ and Fortran code
* Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

**Pandas** is an open-source library that is made mainly for working with relational or labelled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

Before Pandas, Python was capable of data preparation, but it only provided limited support for data analysis. So, Pandas came into the picture and enhanced the capabilities of data analysis. It can perform five significant steps required for processing and analysis of data irrespective of the origin of the data, i.e., load, manipulate, prepare, model, and analyse.

## Key Features of Pandas

* It has a fast and efficient DataFrame object with the default and customised indexing.
* Used for reshaping and pivoting of the data sets.
* Group by data for aggregations and transformations.
* It is used for data alignment and integration of the missing data.
* Provide the functionality of Time Series.
* Process a variety of data sets in different formats like matrix data, tabular heterogeneous, time series.
* Handle multiple operations of the data sets such as subsetting, slicing, filtering, groupBy, re-ordering, and re-shaping.
* It integrates with the other libraries such as SciPy, and scikit-learn.
* Provides fast performance, and If you want to speed it, even more, you can use the Cython.

**Firebase** is a platform developed by Google for creating mobile and web applications. It was originally an independent company founded in 2011. In 2014, Google acquired the platform and it is now their flagship offering for app development.

**Cloud Firestore** is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud. Like Firebase Realtime Database, it keeps your data in sync across client apps through real time listeners and offers offline support for mobile and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud products, including Cloud Functions.

6.3 GUI Interface

Android is an Operating System for mobile devices developed by Google, which is built upon Linux kernel. Android competes with Apple's iOS (for iPhone/iPad), RIM's Blackberry, Microsoft's Windows Phone, Symbian OS, and many other proprietary mobile OSes.

Kotlin Programming Language

In May 2017, Google announced support for Android app development in the Kotlin programming language, supported in Android Studio 3.0. "Kotlin is a statically-typed programming language that runs on the Java virtual machine and also can be compiled to JavaScript source code. While the syntax is not compatible with Java, Kotlin is designed to interoperate with Java code and is reliant on Java code from the existing Java Class Library, such as the collections framework."

3. Android Application Framework

Android application framework is radically different from traditional programming frameworks, such as C/C++ and Java. There is no single entry point (you can't find the main()?!). This is because the Android apps are targeted at mobile devices with limited capabilities. "Reusing" components from other applications is crucial to reduce their footprint.

Android applications consist of components, and can communicate and use components of other applications seamlessly (e.g., making a phone call, or taking a photo), without including the codes of the other applications or linking to it. An Android app can simply start other application's component by instantiate Java objects for that component. An Android application does not have a single entry point (there is no main() method), but consists of components that can be instantiated and run as the need arises.

There are four main types of application components:

Activity: An activity has a single screen, which usually composes of one of more views. An activity interacts with the user to perform one task.

Service:

Background processes (similar to Windows' service or Unix's daemon), e.g., playing music.

Broadcast Receiver: Receives and reacts to system messages, e.g., low battery life

.Content Provider: Android defines a content provider mechanism for applications to share data without exposing the underlying implementation. Via content provider, your application can share data with other applications, or use data from another application (e.g., from SQLite Database).

Activity

An Android application typically comprises several activities. An activity has a single screen, which usually composes of one of more views. An activity, as its name implied, interacts with the user to do ONE (and only ONE) thing, such as viewing data, creating data, or editing data.

View

Views are UI components (or widget, or control) (such as button, label, text field) as well as containers (of components), that could be used to build the user interface. In the Hello-world, we use a view called TextView to display the message.

Android supports many core JDK packages, except graphics packages AWT and Swing. It provides its own 2D graphics supports, via views and widgets. It supports 3D graphics via OpenGL ES.

Fragment

Fragment was introduced in Android 3.0 to support wider screen, which could be difficult to put all the functions in a single activity. Fragments are like sub-activities. An activity can display one or more fragments on the screen at the same time. For a smaller screen, an activity is more likely to contain just one fragment.

**Chapter 7**

**RESULTS AND DISCUSSIONS**

**7.1 Evaluation of machine learning models for covid-19 prediction**

Classification report

A classification report is a performance evaluation metric in machine learning. It is used to show the precision, recall, F1 Score, and support of your trained classification [model](https://thecleverprogrammer.com/2020/11/27/machine-learning-algorithms-with-python/)

Accuracy score

Accuracy is one metric for evaluating classification models. Informally, **accuracy** is the fraction of predictions our model got right.

Confusion matrix

A confusion matrix is a table that is used to define the performance of a classification algorithm. A confusion matrix visualises and summarises the performance of a classification algorithm.

Logistic Regression:

Accuracy on Israel data- 94.99%

Accuracy on Indian data( collected by survey)-75.83%

KNN:

Accuracy on Israeli data- 95.04%

Accuracy on Indian data-71.14%

Gradient Boosting:

Accuracy on Israeli data-95.61%

Accuracy on Indian data- 73.15%

Random Forest:

Accuracy on Israeli data- 93.97%

Accuracy on Indian data-78.52

Naive bayes(Gaussian Naive Bayes used):

Accuracy on Israeli data-94.22

Accuracy on Indian data-72.48%

Scalar Vector Machine:

Accuracy on Israeli data-94.41%

Accuracy on Indian data-73.82%

Chapter 8

**SUMMARY**

Mobile apps are considered to be a valuable tool for citizens, health professionals, and decision makers in facing critical challenges imposed by the pandemic, such as reducing the burden on hospitals, providing access to credible information, tracking the symptoms and mental health of individuals, and discovering new predictors.

We have created a fully functional mobile application that predicts whether or not a person is suspected to be covid positive.

Along with it also displays the accuracy of our model which tells the user the chances of being covid positive .

Our app poses 8 straight forward questions to user and by using a machine model at the backend predicts the result.

We developed the model for predicting COVID-19 diagnosis by asking eight basic questions. Our framework can be used, among other considerations, to prioritise testing for COVID-19 when testing resources are limited. In addition, the methodology presented in this study may benefit the health system response to future epidemic waves of this disease and of other respiratory viruses in general.

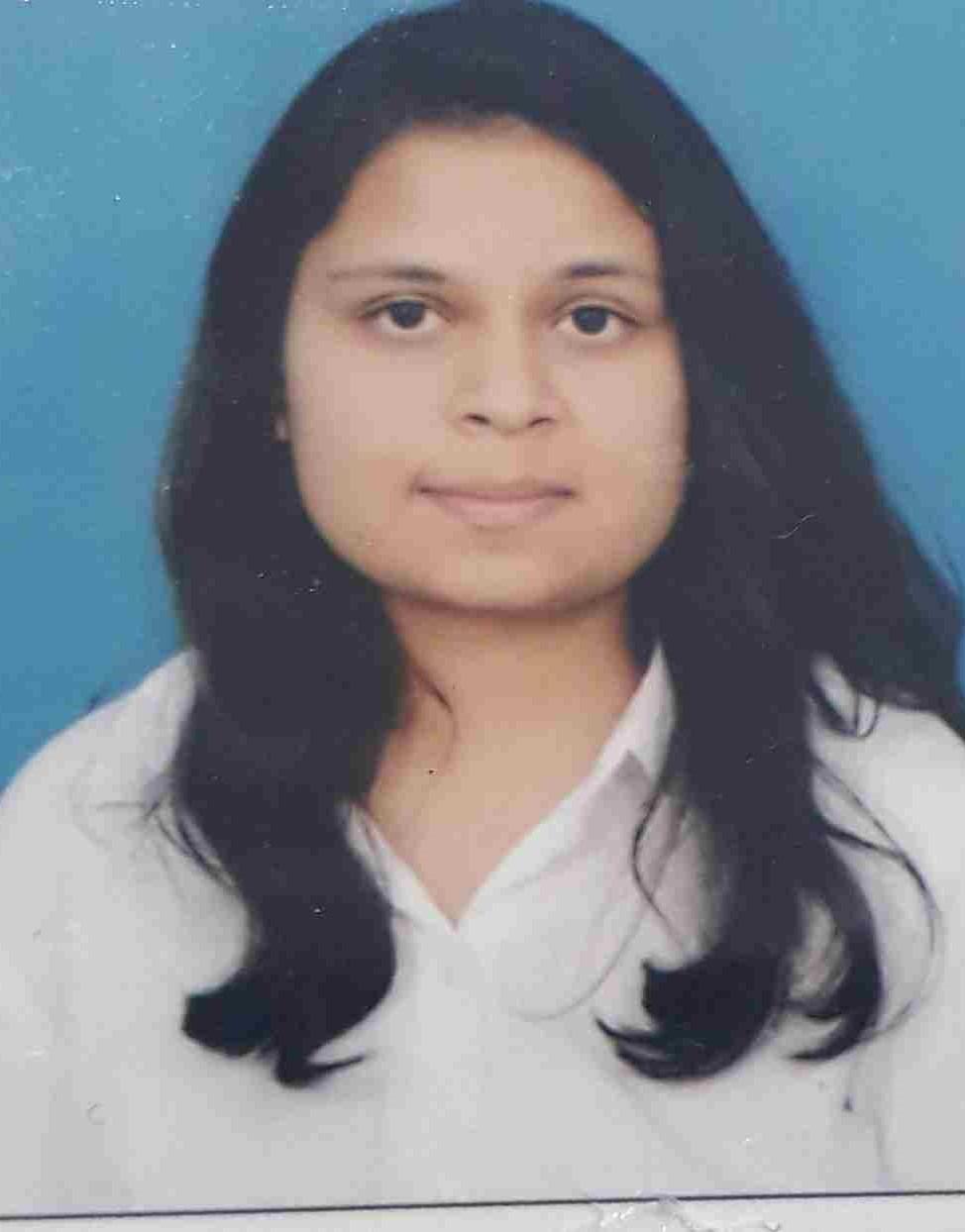
CHAPTER 9

**LITERATURE CITED**

1. https://www.geeksforgeeks.org/disease-prediction-using-machine-learning/
2. <https://www.nature.com/articles/s41746-020-00372-6>
3. eeexplore.ieee.org/document/9430233
4. <https://www.kaggle.com/code/harshaggarwal7/covid-19-symptom-analysis/data>
5. https://ieeexplore.ieee.org/document/9393123
6. <https://ieeexplore.ieee.org/document/9276852>
7. <https://www.sciencedirect.com/science/article/pii/S2405844021022465>

CHAPTER 10

**BIO-DATA OF STUDENTS**



Name - Arushi

ID No - 53500

Place - Roorkee



Name - Simran Chandna

ID No - 53525

Place - Roorkee



Name - Vipul Raj

ID No - 53527

Place - Roorkee

**APPENDIX I**

**Installation of the project**

Android has become a successful platform with a large market share. Thus more and more device vendors leverage the Android OS or the Android Open Source Project (AOSP) for their devices. Google-proprietary applications that are not Open Source can not be included in this 'AOSP' Open Source project. The more device vendors that choose the 'AOSP', the more devices that subsequently enter the market without Google-proprietary apps such as Google Play

"Android developer:

Are you an Android applications developer? Click here to set up your developer account!"

As a developer, one can click "Upload an application" on the "My Actions" menu and follow the provided instructions.

It's important to know they operate a curated store; so apps that do not qualify, or don't meet their quality expectations unfortunately are not accepted.

Application submissions are reviewed by SlideME staff to ensure they meet their minimum standards and quality guidelines. Applications that do not meet these requirements are not approved or are denied. Due to the volume of applications owing off your application. SlideME will then approve your applications for distribution via our partners channels. One can also have a discussion with one’s users or they can review your application right on your page.submitted to the marketplace, it can take a variable amount of time for your application to be properly reviewed.

From the moment one stocks an application, one has options to add descriptions, screenshots, promotional images, target a country, set parental rating (if targeting the Youth channels), add translations, and videos.

One’s application also shows up in the SlideME Market(SAM), their mobile client for discovery and download of Android applications. One can include screenshots and a YouTube video within the mobile catalog, giving potential users a good idea of what one’s app does.

One’s paid applications can be protected using SlideLock, the most robust mechanism for application protection right now.

And all of that, for no fee. There is no setup fee, so one can start uploading one’s applications as soon as one creates the developer account.

SlideME does not prohibit developers from including their own 'in-app' billing within their applications. They are developer savvy and there to help one create awareness of one’s apps and monetize however one wishes to do so.

Our app (will) be uploaded on SlideMe, from where users will be able to download it. ( <http://slideme.org/> )

**APPENDIX II**

**Source Code**

The source code of our project has been uploaded on GitHub. For further reference, please visit <https://github.com/sanjanaaaa/E-commerceAndroidApp> .

**APPENDIX III**

**User Manual & Tutorial**

1.PRE-INSTALLATION REQUIREMENTS:

1.1 Android Phone

1.2 Internet

2. APP INSTALLATION & INTERFACE GUIDE

2.1 App Installation

The user can install the app or apk file from SlideMe (<http://slideme.org/>) once it gets approved and deployed there. The user needs to go to this website, search for the “CoviExpert” and click on install. The app will install in the android phone by itself.

Grant the necessary permissions being asked while installation.

2.2 App Features

2.2.1 Login

2.2.2 Signup

2.2.3 Sign Out

2.2.4 Entering Symptoms

2.2.5 Getting Details

3. DIRECTIONS FOR USE:

Once the user installs the app, it starts with a splash screen of CoviExpert followed by a login page asking for user credentials (Email Id and Password) after which he/she is taken to a category page. If the user is not registered already, “Wrong Details'' message is displayed and the user is required to register using the signup facility provided which will lead him/her to the Login confirmation page to inform him/her about getting authenticated. Here, the user can proceed by clicking on it which will further take the user to a new screen giving him the option either to sign out or proceed to enter the symptoms and basic details. If the user clicks on sign out, his session will expire and he will be logged out from the firebase authentication account and the device will be free for the new user to login.

If the user opts for entering the symptoms and details, the symptoms will be displayed one by one in new screens for the user to select from the options. The user can go to next and previous symptoms by using next and previous buttons respectively. After filling all the details the user can provide confirmation by clicking on submit button and the next screen will be displayed which will provide the prediction result and the details. Here the user can click on the details to know about the technical details and te algorithms used. This will tell user about the accuracy of the prediction made by the app.