IBM – NALAIYATHIRAN PROJECT

HEART DISEASE VISUALISATION WITH AN INTERACTIVE DASHBOARD

INDUSTRY MENTOR : MAHIDHAR , SAUMYA

FACULTY MENTOR : GAYATHRI S

TEAM ID : PNT2022TMID37414

TEAM LEAD : MUSTHABUSHIRA S M

TEAM MEMBER : HUSSAINA PARVEEN J B

TEAM MEMBER : NAVEENA R

TEAM MEMBER : SUSMITHA V

TABLE OF THE CONTENT

CHAPTER-1

INTRODUCTION

a. PROJECT OVERVIEW

Machine Learning is a way of Manipulating and extraction of implicit, previously unknown/known and potentially useful information about data. MachineLearning is a very vast and diversefield and its scope and implementation is increasing day by day. Machine learning Incorporates various classifiers of Supervised, Unsupervised and Ensemble Learning which are used to predictand Find the Accuracy of the given dataset. We can use that knowledgein our project of Heart Disease Prediction as it will help a lot of people. Cardiovascular diseases are very common these days, they describe a range of conditions that could affect your heart. The World health organization estimates that 17.9 million global deaths from (Cardiovascular diseases) CVDs . It is the primary reason for deaths in adults. Our project can help predict the people who are likely to be diagnosed with a heart disease by help of their medicalhistory. It recognizes who all are having any symptoms of heart disease such as chest painor high blood pressure and can help in diagnosing disease with less medical tests and effective treatments, so that they can be cured accordingly. This project focuseson mainly three data mining techniques namely: (1) Logistic regression, (2) KNN and (3) RandomForest Classifier. The accuracy of our projectis 87.5% which is better than the previous system where only one data mining technique was used. So, using more data mining techniques increased the HDPS accuracy and efficiency. Logistic regression falls underthe category of supervised learning. Only discrete values are used in logistic regression. The objective of this project is to check whether the patient is likely to be diagnosed with any cardiovascular heart diseases based on their medical

attributes such as gender, age, chest pain, fasting sugar level, etc. A dataset is selected from the Kaggle. By using this dataset,we predict whetherthe patient can have a heart diseaseor not. To predict this, we use 13 medical attributes of a patient and classify him if the patient is likely to have a heart disease. These medical attributes are trained under three algorithms: K Nearest Neighbour Classifier, Support Vector Classifier, Decision Tree Classifier and Random Forest Classifier. I varied parameters across each model to improve their scores. In the end, K NearestNeighbors Classifier achieved the highest score of 87% with 8 nearest neighbors.

b. PURPOSE:

As we all know, the heart is the most important part of our body other than the brain.Itpumps blood throughthe blood vesselsof the circulatory system. The circulatory system is extremely importantbecause it transports blood, oxygen and other materials to the different organs of the body. Heart plays the most crucial role in the circulatory system. If the heart does not function properly then it willlead to serioushealth conditions including death.For having a healthy heart,there are many solutions available in the market. Exercisecan also play an important role for maintaining heart health. Apart from medical treatments, technology can also prove to be very useful in treating any heart disease. Any heart disease is predicted beforehand, then curing it would be not much complex. But predicting it would be a tough task. Medicalscience has made excellent use of technological breakthroughs to raise the standardof healthcare These technological developments have opened the path for precise illness diagnosis and prognosis Machine learning might be a great option for you obtain a high level of accuracy when it comes to forecasting heart illnesseswith the help ofalgorithm

CHAPTER 2

LITERATURE SURVEY

a. EXISTING PROBLEM:

Before we did the experiments, we did researchon how people explored heart disease prediction so that we can broaden our horizons and learn from them. In 2011, Ujma Ansari[1] made use of the Decision Tree model to predict heart disease and get a high accuracy of 99%, which inspires us to use a better version of Decision Tree and it is RandomForest. Unfortunately, the paper uses a dataset with 3000 instances but does not provide a reference of how they get the data. The UCI websiteonly provides 303 instances of dataset so we doubt where the author gets 3000 instancesof dataset. In 2012, Chaitrali S. Dangare [2] made the prediction by using three models and such models are Naïve Bayes, DecisionTrees and Neural Network. We are using the same dataset as he did. The difference between his work and ours is that he added 2 more features into the dataset, which means thereare 15 features of his work while there are 13 featuresin our dataset. Though there is no big difference between 13 features and 15 features in his work, what he did on dataset inspiresus to make useful changes to our dataset (Try normalization on dataset) to make our results comprehensive. However, during this paper there are only 3 models. More models need to be considered so that the results are comprehensive. In 2017, Kaan Uyar and Ahmet İlhan[3] did the same experiment and used the same datasetas we did for projects. During their analysis, “Class distributions are interpreted as 54% absence and 46% presenceof a heart disease”.The dataset we download from Kaggle has 54% 1s and 46% 0s in the target column. From their analysis, we realize 1 indicates absence of heart disease and vice versa. To make it easily understood, we switched 1s and 0s in the target column so that 1 indicates presence of heart disease to show our confusionmatrix[10] in our results. After reviewing paper [4] and [5], we have learnedthat a neural network has the advantageof fault tolerance and it has the ability to work with inadequate knowledge as human beings.Therefore, in our project we decide to spend some time workingon neural network to detect heart diseases.

b. REFERENCES

- https://en.wikipedia.org/wiki/Cardiovascular_disease
- https://www.heart.org/en/health-topics/heart-attack/warning-signs-of-a-heart-attack
- https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)
- https://food.ndtv.com/health/world-heart-day-2015-heart-disease-in-india-is-a-growing-concern-ansari-1224160

c. PROBLEM STATEMENT DEFINITION:

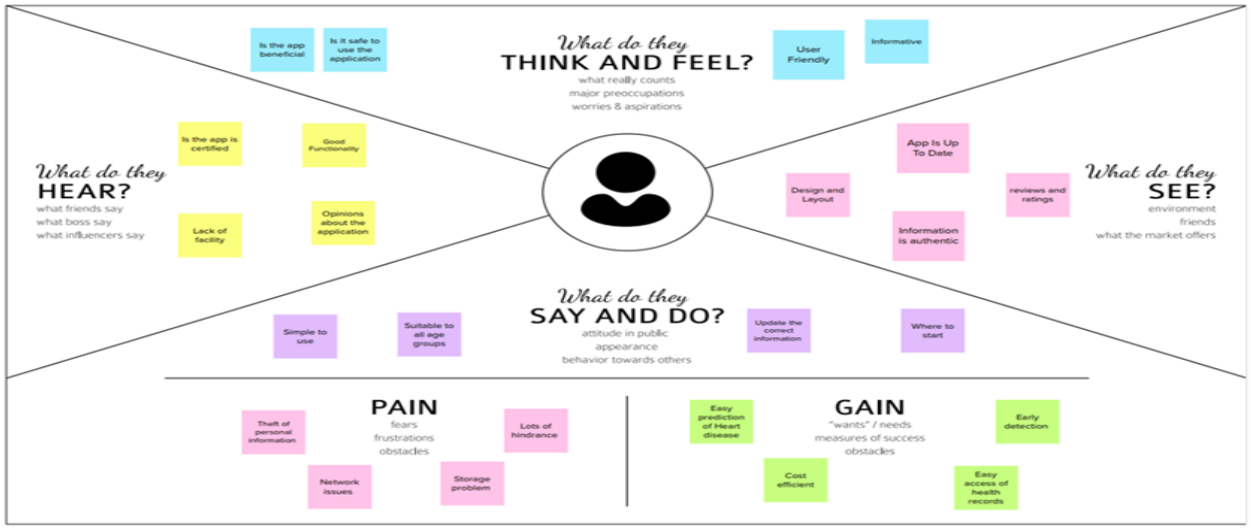
Heart acts a major role in corporeal organism. The diseases of heart wants more perfection and exactness for diagnose and analyses. Heart disease is dangerous disease. This disease occurs due to various problems such as over pressure, blood sugar, high blood pressure, Cholesterol etc. in human body By using Python and machine learning, this paper is analyzed and predicted of the heart disease.. We can predict this disease by using various attributes in the data set. We have collected a data set consists of 13 elements and 383 individual value to analyze the patients performance. The main aim of the paper is to get a better accuracy to detect the heart disease using ML algorithm.

CHAPTER-3

IDEATIONAND PROPOSE SOLUTION

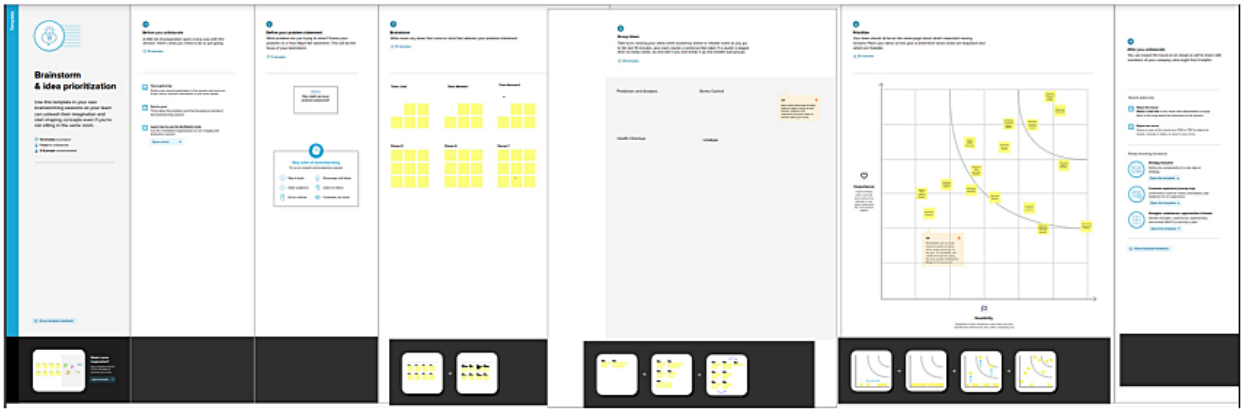
a. EMPATHY MAP CANVAS:

An empathy map is a simple, easy-to-digest visual that capturesknowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding thetrue problem and the person who is experiencing it. The exercise of creating the map helps participants considerthings from the user's perspective along with his or her goals and challenges. An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers.



b. IDEATION AND BRAIN STROMING

A group problem-solving techniquethat involves the spontaneous contribution of ideas fromall members of the group.The mulling over of ideas by one or more individuals in an attempt to devise or find asolution to a problem.



c. PROPOSED SOLUTION

Visualizing And Predicting Heart DiseasesWith An Interactive Dashboard

- The leading cause of death in the developed world is heart disease.Therefore, there needs to be work done to help prevent the risks ofhaving a heart attack orstroke.
- This database contains of 14 fields. The "goal" field refers to the presence of heart diseasein the patient. It is integervalued from 0 (nopresence) to 4.
- Use this dataset to predict which patients are most likely to suffer from aheart diseasein the near future using the featuresgiven.

THE DATA DICTIONARY IS AS FOLLOWS

S.NO	FIELD NAME
1	Age
2	Sex
3	Chest paintype
4	BP
5	Cholesterol
6	FBS over 120
7	EKG results
8	Max HR
9	Exercise angina

10	ST depression
11	Slope of ST
12	Number of vessels fluro
13	Thallium
14	Heart Disease

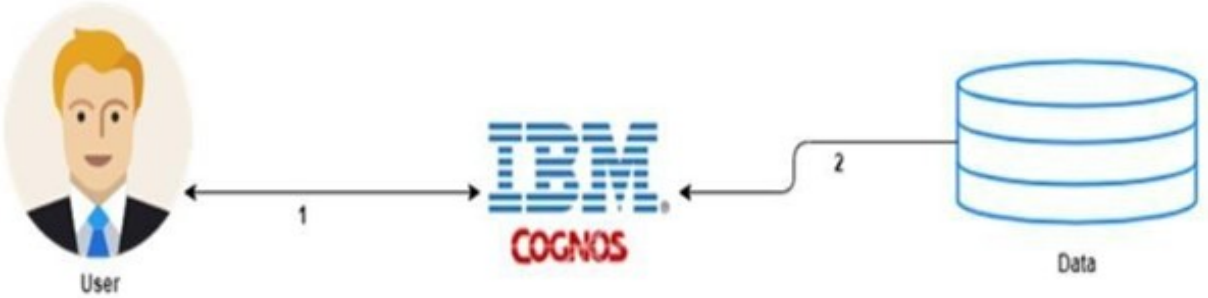
- **NOVELTY**

Heart diseases are the most common cause of death worldwide over the last few decades in the developed as well as underdeveloped and developing countries. Earlydetection of cardiac diseases and continuous supervision of clinicians can reduce the mortality rate. However, it is not possible to monitor patients every day in all cases accurately and consultation of a patientfor 24 hours by a doctor is not available since it requires more sapience, time and expertise. In this project, we have developed and researched about models for heart diseaseprediction through the various heartattributes of the patient and we are going to create the interactive dashboard through which we can analyse the heart diseases based on age, sex, blood pressure of a person,etc.Dataset available publicly in Kaggle Website,further evaluating the results using confusion matrix and cross-validation.

- **FEASIBILITY OF THE IDEA**

Know fundamental concepts and can work on IBM Cognos Analytics, Gain a broad understanding of plotting different visualizations to provide a suitable solution. Able to create meaningful Visualizations and Dashboard(s). We consider a dataset which is having 14 fields by using that we are going to do explorations and buildingvisualizations so that we cananalyse the heart diseases of the patient.

- **BUSINESS MODELS :**

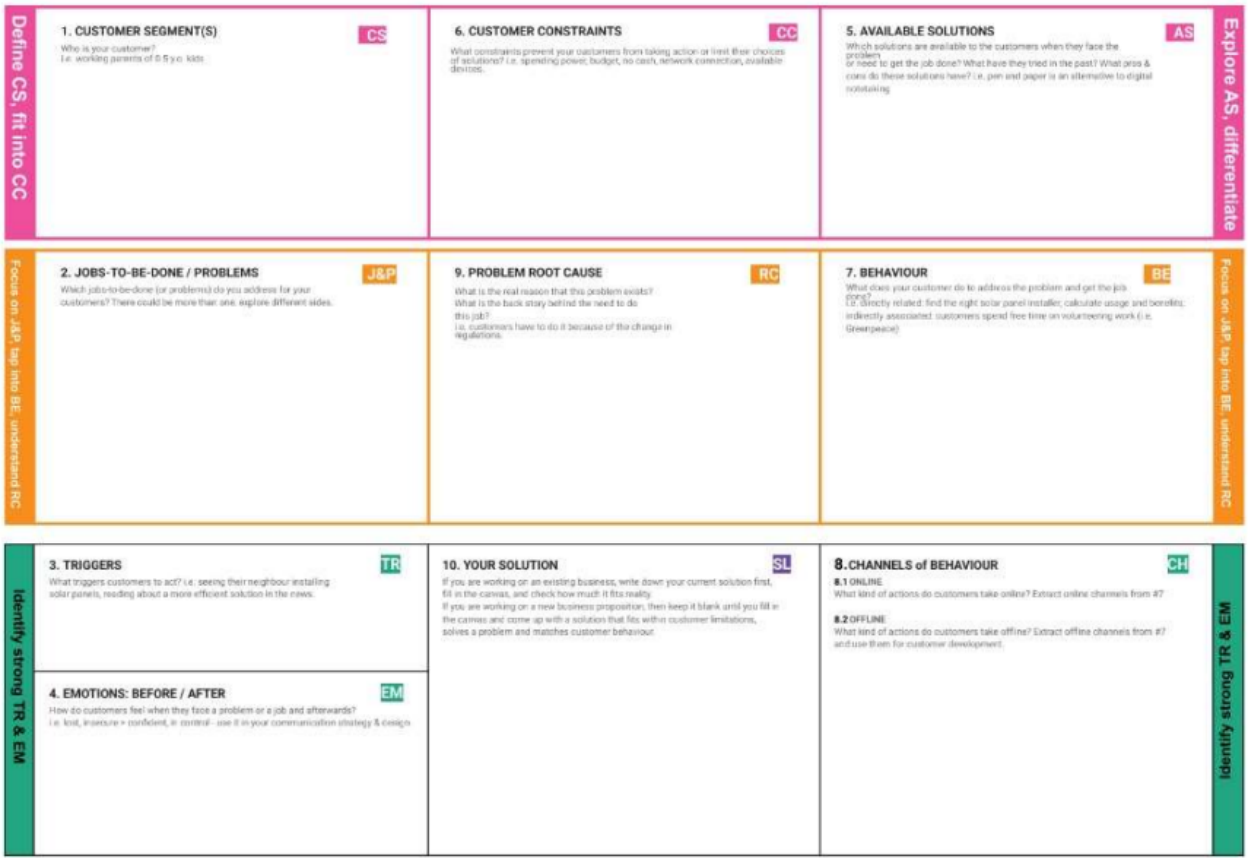


SCALABILITY OF THE SOLUTION:

- We are going to do explorations and visualizations ,Exploration of bp versus chest paintype and gender ,Exploration of max heart rate duringthe chest pain,Exploration of BP by age,Exploration of cholestrol by ageand gender these are the explorations we are going to use.

- Average age for different chest pain types,Average exercise angina during chest pain,BP variation with respect to age,Effort of existing heartdisease on average of exercise angina , Average age for different types of chest pain in existing heart diseases , Maximum heart rate in existing heart disease by exercise angina these allare the visualizations.

d. PROBLEM SOLUTION FIT



1. **COSTUMER SEGMENTS:**

- Hospitals
- Aged people
- Heart patients

2. **JOBS-TO-BE-DONE/PROBLEMS:**

- Wrong prediction or analysis

3. **TRIGGERS:**

- Interactive games related to heart diseases.

4. **EMOTIONS:**

- Before using this app ,they would have been in a misconception .after using it,they've got a clear cut knowledge.

5. **AVAILABLE SOLUTIONS:**

- Short animated awareness videos
- Sensing body conditions with a sensor
- 100% doctor certified

6. **BEHAVIOUR:**

- User-friendly app

7. **CUSTOMER CONSTRAINTS:**

- High paid subscription
- Unwanted advertisements
- Slow working

8. **CHANNELS OF BEHAVIOUR:**

ONLINE:

- We communicate using chat bot

OFFLINE:

- We provide them a phone number through which they can contact us.

9. **PROBLEM ROOT CAUSE:**

- Customer input problems
- Grouping data wrong

10. **YOUR SOLUTION:**

- To create a well interactive dashboard that solves all the customer limitations such as high paid subscriptions, unwanted interuption like ads etc and problems and satisfy customer needs.

CHAPTER-4

REQUIREMENT ANALYSIS

a. FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Network Allowance	Access internet through Wi-Fi Access internet through mobile data
FR-4	App Permissions	Permission for media Permission for camera Permission for notification Permission for calls
FR-5	Terms and Conditions	Accept the policies
FR-6	Inevitable questions	Must attend all the unavoidable questions

b. NON FUNCTIONAL REQUIREMENT

Following are the non-functional requirements of the proposed solution.

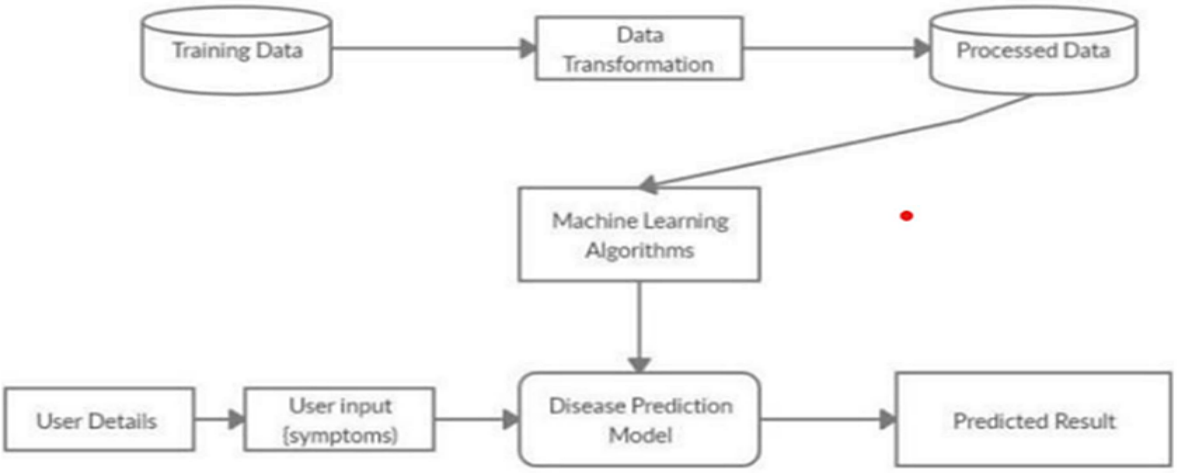
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The dashboard allows the user to perform the tasks easily, efficiently and effectively.
NFR-2	**Security**	Assuring all data inside the application or its part will be protected against malware attacks or Unauthorized access.
NFR-3	**Reliability**	The app must be reliable and strong in giving the functionalities. The movements must be made unmistakable by the app when a customer has revealed a couple of enhancements.
NFR-4	**Performance**	Response Time and Net Processing Time is Fast
NFR-5	**Availability**	The Application will be available up to 99% of the time.
NFR-6	**Scalability**	The Application is scalable.

CHAPTER-5

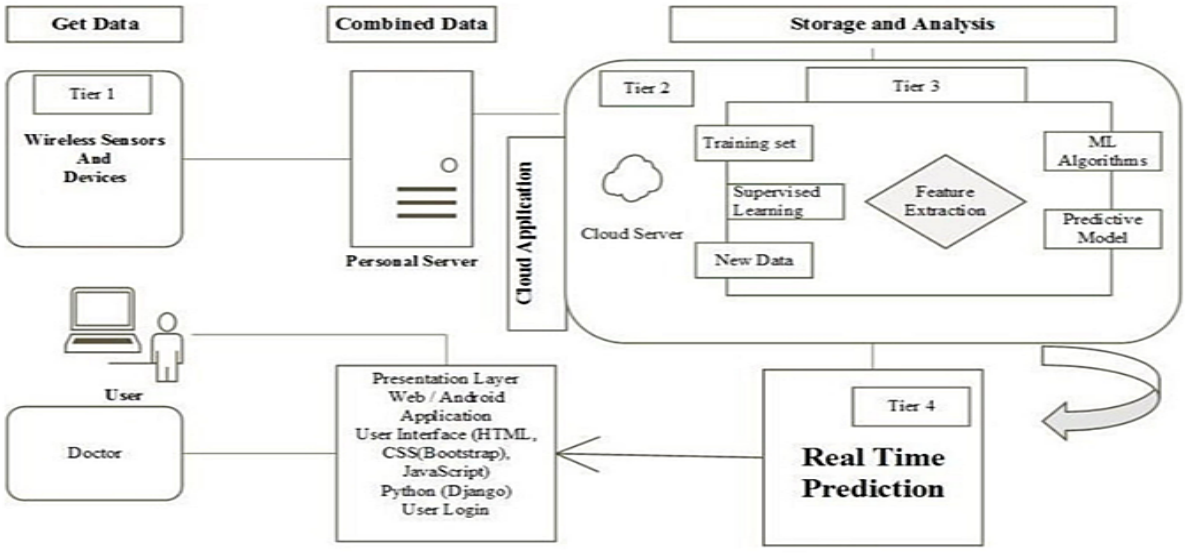
PROJECT DESIGN

a. DATA FLOW DIAGRAM

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



b. SOLUTION AND TECHNICAL ARCHITECTURE



Heart disease is the main cause of death in the developed world.Therefore, efforts must be made to reduce the likelihood of suffering a heart attack or stroke. Using the provided attributes, this dataset can identify which patients are most likely to have a heart condition in the near future. One of the leading causes of morbidity and mortality among the globalpopulation is heartdisease. One of the most crucial topics in the clinical data analysis subsection is the prediction of cardiovascular disease. In the healthcare sector, there is an enormous amount of data. The vast amount of unprocessed healthcare data is transformed via data mining into knowledge that may be used to make forecasts and educated judgments. The dataset consists of 270 individual's data. There are 14 columnsin the dataset, which are described below

IMPORT DATASET

read_csv() is used to read the CSV data with the pandas package, and then with the sklearn package we can work with some models for the prediction process

USER STORIES:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-1
Customer Care Executive	helpline	USN-5	As a user, I can use customer care number to solve my queries	I can solve my queries using the number	Medium	Sprint-2

CHAPTER-6
PROJECT PLANNING & SCHEDULING

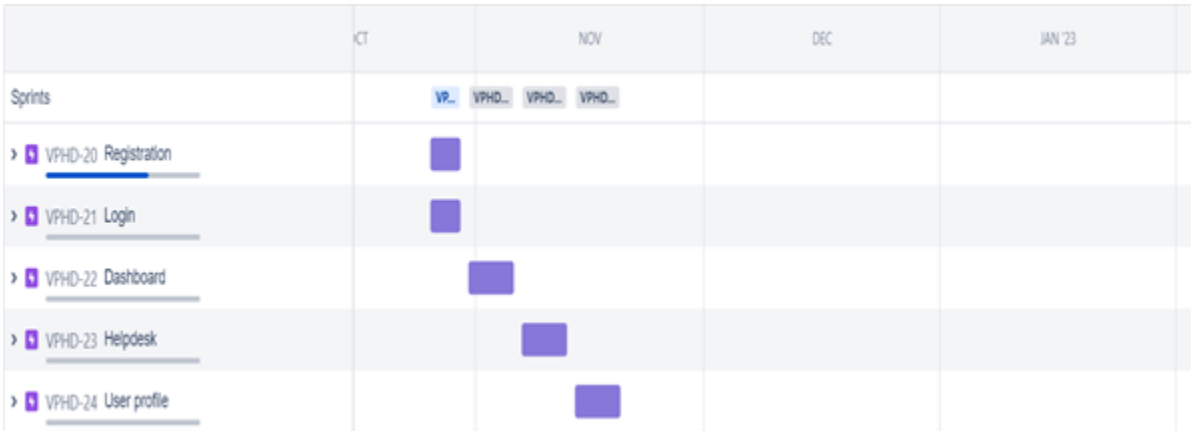
a. SPRINT PLANNINGAND ESTIMATION

SPRINT	FUNCTIONAL REQUIREMENT (EPIC)	USER STORY NUMBER	USER STORY / TASK	STORY POINTS	PRIORITY	TEAM MEMBERS
Sprint-1	Registration	USN-1	As a user, I can register forthe application by entering my email, password, and confirming my password.	10	High	R Naveena J B Hussaina Parveen
Sprint-1	Data Uploading	USN-2	As a user, I will be uploading my data into the cognos analytics	10	High	V Susmitha S M Musthabushira
Sprint-2	Data Analysis	USN-3	As a user, I will be performing analysis on the data foe making predictions	5	High	S M Musthabushira R Naveena
Sprint-2	Dashboard	USN-4	As a user, I will be making visualizations and interacting dashboards from the data	10	High	J B Hussaina Parveen V Susmitha
Sprint-3	Story	USN-5	As a user, I will be making stories from the data and the dashboards	20	High	R Naveena V Susmitha
Sprint-4	Report	USN-6	As a user, I will be making a report from the analysis and dashboards	20	High	S M Musthabushira J B Hussaina Parveen

b. SPRINT DELIVERY AND SCHEDULE

SPRINT	TOTAL STORY POINTS	DURATION	SPRINT START DATE	SPRINT ENDDATE (PLANNED)	STORY POINTS COMPLETED (AS ON PLANNED END DATE)	SPRINT RELEASE DATE (ACTUAL)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

c. REPORTS FROM JIRA



CHAPTER-7

CODING AND SOLUTIONING

a. FEATURES 1:

FRONT END:

- **CSS**
- **BOOTSTRAP**

FRONT END CODE:

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;50
0;600&display=swap');

 *{

font-family: 'Poppins', sansserif;

margin:0;

padding:0;

box-sizing: border-box;

outline: none;

border:none;text-

decoration: none;

}
 .container{

min-height:100vh;

display: flex;
```

```
align-items:
center; justify-
content: center
padding:20px;
padding-bottom:
60px;
}
 .container .content{
text-align: center;
}
.container .content
h3{ font-size: 30px;
color:#333;
}
.container .content h3
span{ background:
crimson; color:#fff;
border-radius: 5px;
padding:0 15px;
}
.container .contenth1{
font-size: 50px;
color:#333;
}

.container .content h1

span{ color:crimson;

}

.container .content p{

font-size: 25px;
```

```
margin-bottom: 20px;
}
.container .content
.btn{ display: inline-
block; padding:10px
30px; font-size: 20px;
background: #333;
color:#fff;
margin:0 5px;
 text-transform:
capitalize;
}
 .container .content
.btn:hover{
background: crimson;
}

.form-container{ min-

height: 100vh;

display: flex;

align-items: center;

justify-content:

center; padding:20px;

padding-bottom:

60px; background:

#eee;

}
```

```
.form-container form{
padding:20px;
 border-radius: 5px;
box-shadow: 0 5px
10px rgba(0,0,0,.1);
background: #fff;
text-align: center;
width: 500px;
}
.form-container form
h3{
font-size: 30px;
text-transform:
uppercase; margin-
bottom: 10px;
color:#333;
}
 .form-container form
input,
 .form-container form
select{
width: 100%;
padding:10px 15px;
font-size: 17px;
margin:8px 0;
background: #eee;
border-radius: 5px;
}
.form-container form
select option{
background: #fff}
 .form-container form
```

```
.form-btn{
background: #fbd0d9;
color:crimson;
text-transform:
capitalize; font-size:
20px;
cursor: pointer;
}
 .form-container form
.form-
btn:hover{backgroun
d: crimson;
color:#fff;
 }
 .form-container form
p{ margin-top: 10px;
font-size: 20px;
color:#333;
}
 .form-container form
p a{ color:crimson;
}
 .form-container form
.error-msg{
margin:10px 0;
display: block;
 background:
crimsocolor:#fff;
 border-radius: 5px;
font-size: 20px;
padding:10px;
}
```

```
 function
myfunction(){
 var x
=document.getEleme
ntById("pass");
if(x.type ===
"password"){x.type
"text";
}
 else{
 x.type = "password";
}
}
 function validate(){
 var password =
document.getElement
ById("pass");var
length =
document.getElement
ById("length");
if(password.value.len
gth >= 8){
alert("Login
Successful");
window.location.repla
ce("heart .html");
return false;
}
 else{
n;alert("Login
Failed");
 }
```

```
 }
 function page(){
window.location.repla
ce("Landingpage.html
")
}
window.watsonAssist
antChatOptions = {
 integrationID:
"3f5d7446-04cb-
4796-8496-
1351f8d8acfd", // The
ID of this
integration.
region: "au-syd", //
The region your
integration is hosted
in. serviceInstanceID:
"58af1a9a-a26e-4a48-
ae6f-0881b0be4c0d",
// The ID of
your service instance.
 onLoad:
function(instance) {
instance.render(); }
 };
setTimeout(function)

const
t=document.createEle
ment('script');
```

```
t.src="https://web-
chat.global.assistant.
watson.appdomain.cl
oud/versions/" +
(window.watsonAssis
tantChatOptions.clien
tVersion || 'latest') +
"/WatsonAssistantCha
tEntry.js";
document.head.appen
dChild(t);
 });
 function
myfunction(){
 var x
=document.getEleme
ntById("pass");if(x.ty
pe === "password"){
x.type = "text";

}

else{

x.type = "password";
 }
}
function validate(){
varpassword=docume
nt.getElementById("p
ass");var
length=document.get
ElementById("length
```

```
");
if(password.value.len
gth >= 8){
alert("Login
Successful");
window.location.repla
ce("heart .html");
return false;
}
 else{

alert("Login Failed");

}

}

function page(){
window.location.repla
ce("Landingpage.html
")
}
```

SCREENSHOTS

b. FEATURE 2:

BACK END CODE:

```
#!/usr/bin/env python
# coding: utf-8
# In[2]:


import pandas as pd
import numpy as np
import
matplotlib.pyplot as
plt import seaborn as
sns
import warnings
warnings.filterwarnin
gs('ignore') # In[4]:
df=pd.read_csv('Dow
nloads/Heart_Disease
_Prediction.csv') #
In[5]:
df.head()
```

```
 # In[6]:
df.isnull().sum() #
In[7]:
print(df.info()) #
In[9]:
plt.figure(figsize=(20,
10))
sns.heatmap(df.corr(),
annot=True,
cmap='terrain') #
In[10]:
sns.pairplot(data=df)
# In[11]:
df.hist(figsize=(10,12
), layout=(5,4)); #
In[13]:
df.plot(kind='box',
subplots=True,
layout=(6,3),
figsize=(10,10))plt.sh
ow()
# In[19]:
sns.catplot(data=df,
x='Sex', y='Age',
hue='Heart Disease',
palette='tab10') #
In[20]:
sns.barplot(data=df,
x='Sex',
y='Cholesterol',
hue='Heart
Disease',palette='sprin
```

```
g')
# In[21]:
df['Sex'].value_counts
() # In[25]:
gen.plot(kind='bar',
stacked='True',
color=['green','blue'],g
rid=False) # In[ ]:
# In[ ]:
# In[22]:
 df['Chest pain
type'].value_counts()
# In[23]:
sns.countplot(x='Che
st pain type',
hue='Heart Disease' ,
data=df,palette='rocke
t')
 # In[24]:
gen =
pd.crosstab(df['Sex'],
df['Heart Disease'])
print(gen)
#!/usr/bin/env python
# coding: utf-8 #
In[1]:
import pandas as pd
import numpy as np
import
matplotlib.pyplot as
plt import seaborn as
sns
```

```
import warnings
warnings.filterwarnin
gs('ignore') # In[2]:
df=pd.read_csv('Dow
nloads/Heart_Disease
_Prediction.csv') #
In[3]:
df.head()
# In[4]:
 df.isnull().sum() #
In[5]:
print(df.info()) #
In[6]:
plt.figure(figsize=(20,
10))
sns.heatmap(df.corr(),
annot=True,
cmap='terrain') #
In[7]:
sns.pairplot(data=df)
# In[8]:
df.hist(figsize=(10,12
), layout=(5,4)); #
In[9]:
df.plot(kind='box',
subplots=True,
layout=(6,3),
figsize=(10,10))plt.sh
ow()
# In[10]:
```

```
sns.catplot(data=df,
x='Sex', y='Age',
hue='Heart Disease',
palette='tab10') #
In[11]:
sns.barplot(data=df,
x='Sex',
y='Cholesterol',
hue='Heart
Disease',palette='sprin
g')
 # In[12]:
df['Sex'].value_counts
() # In[13]:
df['Chest pain
type'].value_counts()
# In[14]:
sns.countplot(x='Che
st pain type',
hue='Heart Disease' ,
data=df,palette='rocke
t')
 # In[15]:
 gen =
pd.crosstab(df['Sex'],
df['Heart
Disease'])print(gen)
# In[16]:
 gen.plot(kind='bar',
stacked='True',
color=['green','blue'],g
rid=False) # In[17]:
from
sklearn.model_selecti
on
importtrain_test_split
 from
sklearn.preprocessing
importStandardScaler
```

```
StandardScaler =
StandardScaler()
columns_to_scale=['A
ge', 'EKG results',
'Cholesterol',
'Thallium', 'Number
of vessels fluro']
 df[columns_to_scale]
=
StandardScaler.fit_tra
nsform(df[columns_t
o_scale]) # In[18]:
df.head()
 # In[19]:
 from
sklearn.model_selecti
on import
train_test_split from
sklearn.preprocessing
import
StandardScaler
StandardScaler =
StandardScaler()
columns_to_scale=['A
ge', 'EKG results',
'Cholesterol',
'Thallium', 'Number
of vessels fluro']
df[columns_to_scale=
StandardScaler.fit_tra
nsform(df[columns_t
o_scale]) # In[20]:
df.head()
 # In[21]:
 x=df.drop(['Heart
Disease'], axis=1)
y=df['Heart Disease']
# In[22]:
 x_train, x_test,
```

```
y_train,
y_test=train_test_split
(x,y,test_size=0.3,
random_state=40)
 # In[23]:
 print('x_train-',
x_train.size)
print('x_test-',
x_test.size)
print('y_train-',
y_train.size)
print('x_test-',
x_test.size) # In[24]:
from
sklearn.linear_model
import
LogisticRegressionlr=
LogisticRegression()
model1=lr.fit(x_train,
y_train)
prediction1=model
1.predict(x_test)
In[25]:
from sklearn.metrics
import
confusion_matrix
cm=confusion_matrix
(y_test,prediction1)
cm
# In[26]:
 sns.heatmap(cm,
annot=True,cmap='Bu
Pu') # In[27]:
TP=cm[0][0]
 TN=cm[1][1]
 FN=cm[1][0]
 FP=cm[0][1]
 print('Testing
Accuracy:',
```

```
(TP+TN+FN)/(TP+T
N+FN+FP)) # In[28]:
from sklearn.metrics
import
accuracy_score
accuracy_score(y_test
,prediction1)
l=accuracy_score(y_t
est,prediction1)
# In[29]:
from sklearn.metrics
import
classification_report
print(classification_re
port(y_test,
prediction1)) #
In[30]:
importpandas as pd
 from sklearnimport
neighbors,metrics
 from
sklearn.model_selecti
on import
train_test_split from
sklearn.neighbors
import
KNeighborsClassifier
import numpy as np
importpickle
 from
sklearn.ensemble
import
RandomForestClassifi
erimport pandas as pd
from sklearn.tree
import
DecisionTreeClassifie
rimport seaborn as
sns
```

```
import
matplotlib.pyplot as
plt # In[31]:
from sklearn.metrics
import
accuracy_score #
In[32]:
dataset =
pd.read_csv("Downlo
ads/Heart_Disease_Pr
ediction.csv") #
In[33]:
KX =
dataset[['Age','Sex','C
hest pain
type','BP','Cholesterol'
,'FBS over 120','EKG
results','Max
HR','Exercise
angina','ST
depression','Slope of
ST','Number of
vessels
fluro','Thallium']].val
ues
# In[34]:
 KY = dataset[['Heart
Disease']].values #
In[35]:
KX
# In[36]:
 KY = KY.flatten()
print(KY)
# In[37]:
 KX_train , KX_test ,
KY_train , KY_test =
train_test_split(KX,K
Y,test_size=0.2,rando
m_state=4)
```

```
# In[38]:
 kn
KNeighborsClassifier
(n_neighbors = 20)
knn.fit(KX_train,
KY_train)
print(knn.score(KX_t
est, KY_test))
# In[39]:
pickle.dump(knn,ope
n('heart_knn_model.s
av','wb')) # In[40]:
predict_knn =
knn.predict(KX_test)
 accuracy_knn =
metrics.accuracy_scor
e(KY_test,predict_kn
n)# In[41]:
predict_knn # In[42]:
accuracy_knn #
In[43]:
k=accuracy_knn #
In[45]:
import csv
import pandas as pd
import numpy as np
from
sklearn.naive_bayes
import GaussianNB
from
sklearn.model_selecti
on import
train_test_split from
sklearn importmetrics
from sklearn.metrics
import
confusion_matrix,
f1_score, roc_curve,
auc import
```

```
matplotlib.pyplot as
plt
from itertools import
cycle from scipy
import interp
# In[46]:
 df =
pd.read_csv('Downlo
ads/Heart_Disease_Pr
ediction.csv', header =
None) # In[47]:
training_x=df.iloc[1:d
f.shape[0],0:13] #
In[48]:
training_y=df.iloc[1:d
f.shape[0],13:14] #
In[49]:
nx=np.array(training_
x)
ny=np.array(training_
y) # In[52]:
for z in range(5):
print("\nTest Train
Split no. ",z+1,"\n")
nx_train,nx_test,ny_tr
ain,ny_test =
train_test_split(nx,ny,
test_size=0.25,rando
m_state=None) #
Gaussian function of
sklearn
gnb = GaussianNB()
gnb.fit(nx_train,
ny_train.ravel())
ny_pred =
gnb.predict(nx_test)
# In[61]:

print("\n Naive Bayes
```

```
model accuracy(in
%):",
metrics.accuracy_scor
e(ny_test, ny_pred))
# In[62]:
n=metrics.accuracy_s
core(ny_test,
ny_pred) # In[64]:
import pandas as pd
from sklearnimport
neighbors,metrics
from
sklearn.model_selecti
on import
train_test_split from
sklearn.neighbors
import
KNeighborsClassifier
import numpy as np
importpickle

from
sklearn.ensemble
import
RandomForestClassifi
erimport pandas as pd
from sklearn.tree
import
DecisionTreeClassifie
rimport seaborn as
sns
import
matplotlib.pyplot as
plt # In[65]:
from sklearn.metrics
import
accuracy_score #
In[67]:
dataset =
```

```
pd.read_csv("Downlo
ads/Heart_Disease_Pr
ediction.csv") #
In[69]:
DX =
dataset[['Age','Sex','ch
est pain
type','BP','Cholesterol'
,'FBS over 120','EKG
results','Max
HR','Exercise
angina','ST
depression','Slope of
ST','Number of
vessels
fluro','Thallium']].val
ues
# In[70]:
dy = dataset[['Heart
Disease']].valueE #
In[71]:
DX
 # In[72]:
 dy = dy.flatten()
print(dy)
# In[73]:
 DX_train , DX_test
dy_train , dy_test =
train_test_split(DX,d
y,test_size=0.2,rando
m_state=4)
 # In[74]:
 from sklearn.tree
import
DecisionTreeClassifie
rmax_accuracy = 0
# In[75]:

or x in range(200):
```

```
dt =
DecisionTreeClassifie
r(random_state=x)
dt.fit(DX_train,dy_tra
in)
dy_pred_dt =
dt.predict(DX_test)
 current_accuracy =
round(accuracy_score
(dy_pred_dt,dy_test)*
100,2)
if(current_accuracy>
max_accuracy):
max_accuracy =
current_accuracy
best_x = x # In[85]:
dt =
DecisionTreeClassifie
r(random_state=best_
x)dt.fit(DX_train,dy_t
rain)
dy_pred_dt =
dt.predict(DX_test) #
In[88]:
score_dt =
(accuracy_score(dy_p
red_dt,dy_test)) #
In[89]:
print("The accuracy
score achieved using
Decision Tree is:
"+str(score_dt)) #
In[90]:
d=(accuracy_score(dy
_pred_dt,dy_test)) #
In[91]:
print('Logistic
Regression :',l)
print('KNN :',k)
```

```
print('Naive Bayes
:',n) print('Decision
Tree :' ,d) # In[93]:
print('Logistic
Regression
:',l*100,'%')
print('KNN
:',k*100,'%')
print('Naive Bayes
:',n*100,'%')
print('Decision Tree :'
,d*100,'%') # In[ ]:
import js2py
import os
from flask import
Flask,request,jsonify,
json, Response,
make_response,
render_template
from flask_pymongo
import PyMongo
from flask_bcrypt
import Bcrypt
from flask_cors
import CORS import
db
js2py.run_file(bot.js)
app = Flask( name )
bcrypt = Bcrypt(app)
CORS(app)@app.rout
e('/')
@app.route("/test")
def test():
return "Connected to
the database!" class
UserAuthUtil:
@app.route("/",
methods=['GET']) def
hello_world():
```

```
return "Working"
@app.route("/login",
methods=['POST'])
def login_user():
try:
if request.method ==
'POST': form_data =
request.get_json()ema
il =
form_data['email']
password =
form_data['password']
if(email != '' and
password != ''):
data =
list(db.users.find({'em
ail': email}))
if(len(data) == 0):
return
Response(status=404,
response=json.dumps
({'message': 'user
does not exist'}),
mimetype='applicatio
n/json')
else:
data = data[0]
if(bcrypt.check_pass
word_hash(data['pass
word'], password)):
#token
=jwt.encode({'email':
email},
app.config['SECRET
_KEY'])
return
make_response(jsonif
y({'message':'User
logged in
```

```
successfully'}), 201)
else:
return
Response(status=402,
response=json.dumps
({'message': 'Invalid
password'}),
mimetype='applicatio
n/json')
else:
return
Response(status=400,
response=json.dumps
({'message': 'Bad
request'}),
mimetype='applicatio
n/json')
else:
return
Response(status=401,
response=json.dumps
({'message': 'invalid
request type'}),
mimetype='applicatio
n/json')
except Exception as
Ex:
print('\n\n\n********
*****************
********') print(Ex)
print('**************
*****************
***\n\n\n')
return
Response(response=js
on.dumps({'message':
"Internal Server
error"}), status=500,
mimetype="applicatio
```

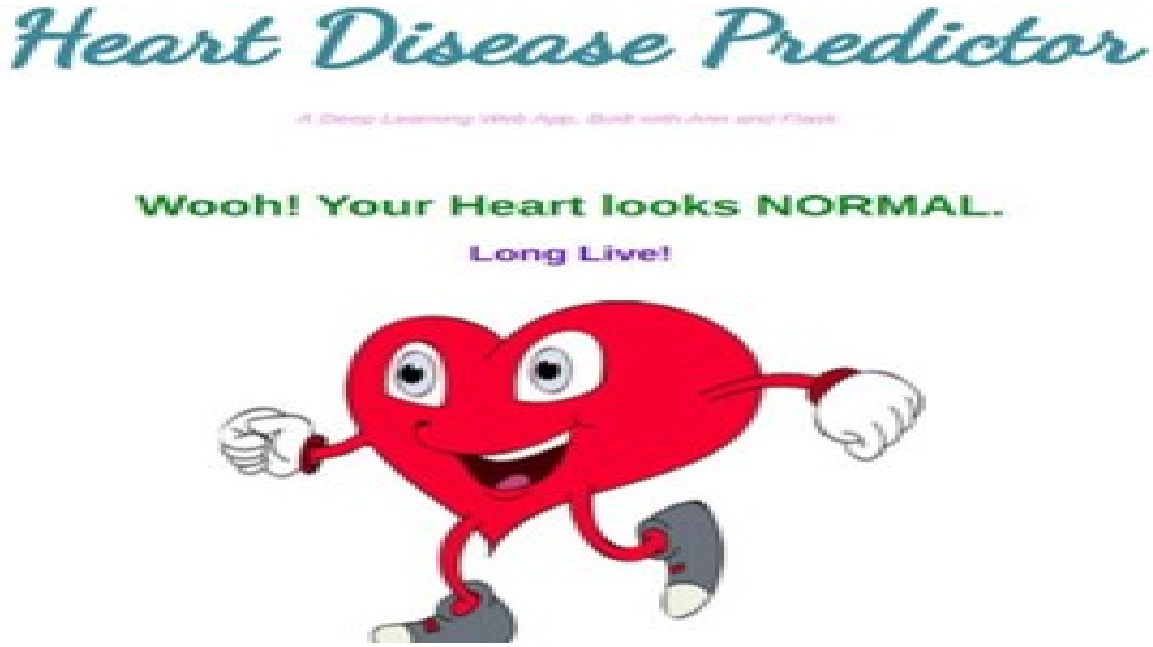
```
n/json")
@app.route("/register
", methods=['POST'])
def register_user():
try:
if request.method ==
"POST":user_details
= request.get_json()
full_name =
user_details["fullNam
e"] email =
user_details["email"]
password =
user_details["passwor
d"]
password_hash =
bcrypt.generate_pass
word_hash(password)
.decode('utf-8')
if (full_name != '' and
email != '' and
password_hash != ''):
db.users.insert_one({'
fullName':full_name,'
email':email,'passwor
d':
password_hash})
return
Response(response=js
on.dumps({'message':
'User created
successfully'}),
status=200,
mimetype="applicatio
n/json")
else:
return
Response(status=400,
response=json.dumps
```

```
({'message':
'Pleaseenter your
details'}),
mimetype='applicatio
n/json')
else:
return
Response(status=400,
response=json.dumps
({'message': 'Bad
request'}),
mimetype='applicatio
n/json')
except Exception as
Ex:
print('\n\n\n********
*****************
********') print(Ex)
                    print('**********************************
                                        *\n\n\n')

                        return Response(response=json.dumps({'message':
            "Internal ServerError"}), status=500,
            mimetype="application/json")

if____________name__
== '___main___':
app.run(port=8000)
```

SCREENSHOTS:



CHAPTER-8

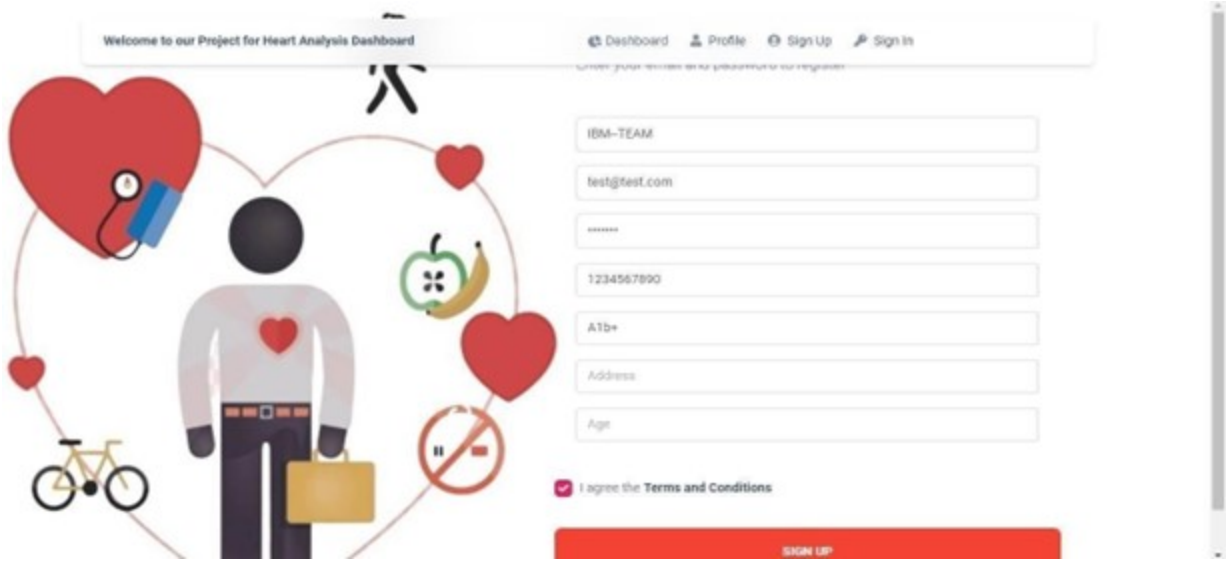
TESTING

a. TEST CASES

Software testing is a criticalelement of softwarequality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructiverather than constructive. A strategy for software testingintegrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advanceand conducted systematically. The underlying motivation of program testingis to affirm software qualitywith methods that can economically and effectively applyto both strategic to both large and small-scale systems.

LOGIN

Can't log into the application by entering email/PhoneNumber & password



CHAPTER 9

RESULTS

a. PERFORMANCE METRICS

Performance Analysis is the one used for predicting the algorithm based on various metricssuch as accuracy, precision, recall/F1-Score, etc. Performance Analysis aims at comparing the accuracy and performance of machine learning models. The metrics is evaluated withfour measures.

TP [True Positive]

If the positive input (from dataset) is given to the classifier, it gives positive output (predicted value). It predicts the total true positive cases identified correctly. True positive values will be 1(True) for heart failured patients. The predicted value will be value 1 if the true positive value is 1.

TN [True Negative]

If the negative input (from dataset) is given to the classifier, it gives negative output (predicted value). It predicts the total true negative cases identified correctly. The true negative values will be 0(False) for heart failured patients. The predicted value will be1(True) if the true negative value is 0(false).

FP [False Positive]

If the negative input (from dataset) is given to the classifier, it gives positive output (predicted value). It predicts the total false positive cases identified incorrectly. The false positivevalues will be 0 for heart failuredpatients. The predictedvalue will be 1 if the falsepositive value is 0.

FN [False Negative]

If the positive input (from dataset) is given to the classifier, it gives negative output (predicted value). It predicts the total false negative cases identified incorrectly. The false negative values will be 0 for heart failured patients. The predicted value will be 0 if the values predicted are false.

The important metricsfor performance analysisare accuracy, recall (F1-Score) and

precision. These above measures are used to define the metric.

Accuracy (A)

Accuracy is a performance metric that has the correct predictions for the test data. It gives the percentage of correct predictions for testing the data. In machine learning, accuracy is calculated using the formula as shown in equation (1),

(TP+TN)/(TP+TN+FP+FN)

recall (R) or F1-Score

Recall metric is used to predict the number of correct samples (all samples identified as positive). It is the fraction of values (results) returned to the total number of values that can be returned. It is the ratio betweentrue positives and all the actual positives. It measures the correctly identified positive samples out of all the actual positive samples. The recall is calculated using the formula as shown in equation (2),

TP/(TP+FN)

Precision (P)

Precision which is also called as “positive predictive value”. It gives the percentage of true positives. It is the ratio between true positives and all the predicted positives. It measures the correctly identified positive samples out of all positively predictedsamples. The precision is calculated using the formulaas shown in equation (3),

TP/(TP+FP)

CHAPTER-10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES

- Increased accuracyfor effective heart disease diagnosis.
- Handles roughest(enormous) amount of data using random forest algorithmandfeature selection.
- Reduce the time complexity of doctors.
- Cost effective for patients.

DISADVANTAGES

- Prediction of cardiovascular diseaseresults is not accurate.
- Data mining techniques do not help to provideeffective decision making.
- Cannot handle enormousdatasets for patientrecords

CHAPTER-11

CONCLUSION

we proposed a method for heart diseaseprediction using machinelearning techniques, these results showed a great accuracy standardfor producing a better estimation result. By introducing new proposed Random forest classification, we find the problem of prediction rate without equipmentand propose an approach to estimate the heart rate and condition. Sample resultsof heartrate are to be taken at different stagesof the same subjects, wefind the information from the above input via ML Techniques. Firstly, we introduced a support vector classifier based on datasets.

CHAPTER-12

FUTURESCOPE

The future scope of this system aims at giving more sophisticated prediction models, risk calculation tools and featureextraction tools for other clinical risks. Here the scope of the project is that integration of clinical decision support with computer-based patient records could reduce medical errors, enhance patient safety, decrease unwanted practice variation, and improve patient outcome. This suggestion is promising as data modeling and analysis tools, e.g., data mining, have the potential to generate a knowledge- rich environment which can help to significantly improve the quality of clinical decisions

CHAPTER 13

APPENDIX

SOURCE CODE: https://github.com/IBM-EPBL/IBM-Project-42740-1660708289/tree/main/Final%20Deliverables/Final%20Code

GITHUB AND PROJECT DEMO LINK:

GITHUB: https://github.com/IBM-EPBL/IBM-Project-42740-1660708289

PROJECT DEMO LINK:

https://drive.google.com/file/d/15Gr5ScYhGsxAj9fuj5BlRmsCj4w1rw16/view?usp=drivesdk