Visualizing and Predicting Heart Diseases with an Interactive Dashboard

NALAIYA THIRAN PROJECT REPORT
2022

Submitted by

PRASHANTH L	**422519205031**
MUGILAN M	**422519205026**
DAVID M	**422519205010**
PRAVEENKUMAR S	**422519205034**

Team ID: PNT2022TMID29391

Content

TEAM ID : PNT2022TMID29391

VISUALIZING AND PREDICTING HEART DISEASES WITH AN INTERACTIVE DASHBOARD

1. Introduction

1.1 Project Overview

The leading cause of death in the developed world is heart disease. Therefore, there needs to be work done to help prevent the risks of having a heart attack or stroke. This project aims to create an interactive Dashboard using IBM Cognos Tool and dataset to predict which patients are most likely to suffer from a heart disease in the near future using the features given.

1.2 Purpose

Heart disease (HD) is a major cause of mortality in modern society. Medical diagnosis is an extremely important but complicated task that should be performed accurately and efficiently. Cardiovascular disease is difficult to detect due to several risk factors, including high blood pressure, cholesterol, and an abnormal pulse rate. Based on the analytics we can analyze which patients are most likely to suffer from heart disease in the near future and based on the patient details we will make decisions to cure them.

2. Literature Survey

2.1 Existing Problem

Healthcare industries generate enormous amount of data, so called big data that accommodates hidden knowledge or pattern for decision making. The huge volume of data is used to make decision which is more accurate than intuition. Exploratory Data Analysis (EDA) detects mistakes, finds appropriate data, checks assumptions and determines the correlation among the explanatory variables. In the context, EDA is considered as analysing data that excludes inferences and statistical modelling. Analytics is an essential technique for any profession as it forecast the future and hidden pattern. Data analytics is considered as a cost effective technology in the recent past and it plays an essential role in healthcare which includes new research findings, emergency situations and outbreaks of disease. The use of analytics in healthcare improves care by facilitating preventive care and EDA is a vital step while analysing data..

2.2 References

"Heart Disease Prediction using Exploratory Data Analysis" R. Indrakumari, T.Poongodi, Soumya Ranjan Jena

In this paper, the risk factors that causes heart disease is considered and predicted using K-means algorithm and the analysis is carried out using a publicly available data for heart disease. The dataset holds 209 records with 8 attributes such as age, chest pain type, blood pressure, blood glucose level, ECG in rest, heart rate and four types of chest pain. To predict the heart disease, K-means clustering algorithm is used along with data analytics and visualization tool. The paper discusses the pre-processing methods, classifier performances and evaluation metrics. In the result section, the visualized data shows that the prediction is accurate.

Prediction of heart disease at early stage using data mining and big data analytics: A survey N. K. Salma Banu, Suma Swamy

Several studies have been carried out for developing prediction model using individual technique and also by combining two or more techniques. This paper provides a quick and easy review and understanding of available prediction models using data mining from 2004 to 2016. The comparison shows the accuracy level of each model given by different researchers. Into practice.

2.3 Problem Statement Definition

Who does the problem affect?

People with unhealthy lifestyles, stress, depression, age above 40 and when their ancestors got heart disease (since heart disease is hereditary).

When does the issue occur?

The issue occurs for people with unhealthy lifestyles and age above 40. Where is the issue occurring? The issue is originating from an unhealthy lifestyle. It mostly occurs in the blood valves of the heart.

What would happen if we didn't solve the problem?

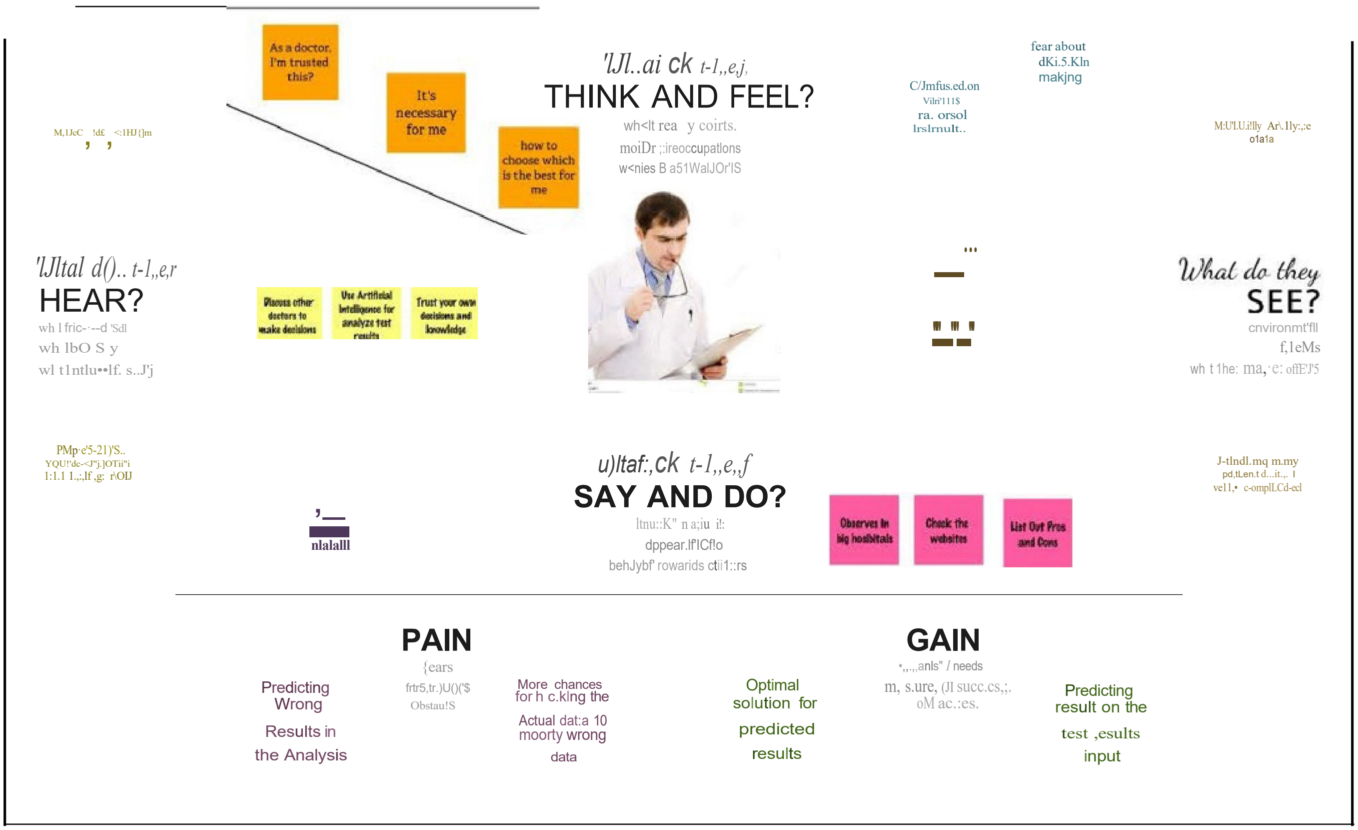
If we don't solve the problem, many people will die at a young age. The death rate due to heart disease will increase rapidly.

Why is it important to fix the problem?

We should predict the problem before giving treatment to the patients. As the problem is predicted early, we can solve it easily and early.

3. Ideation and Proposed Solution

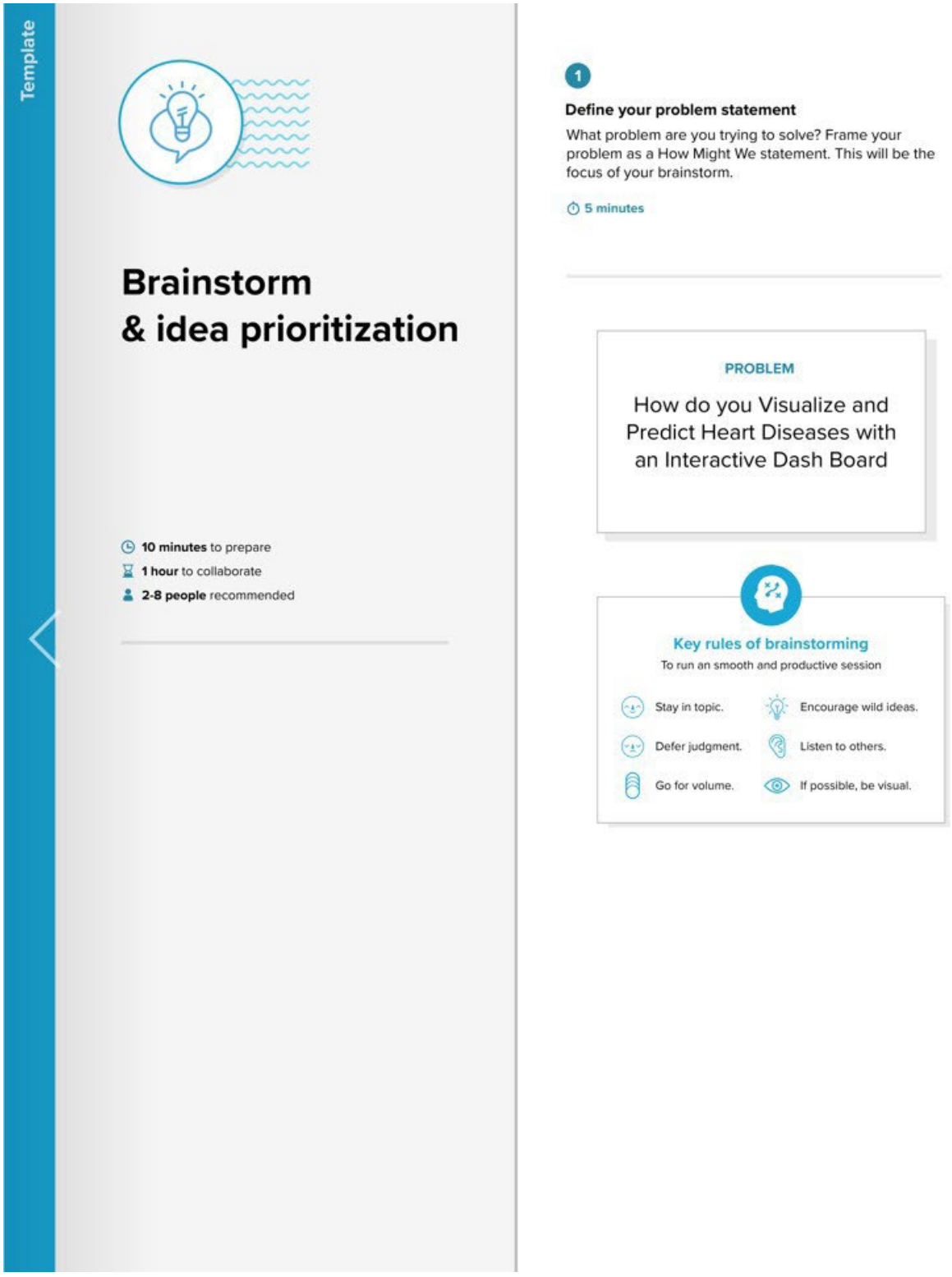
3.1 Empathy Map Canvas



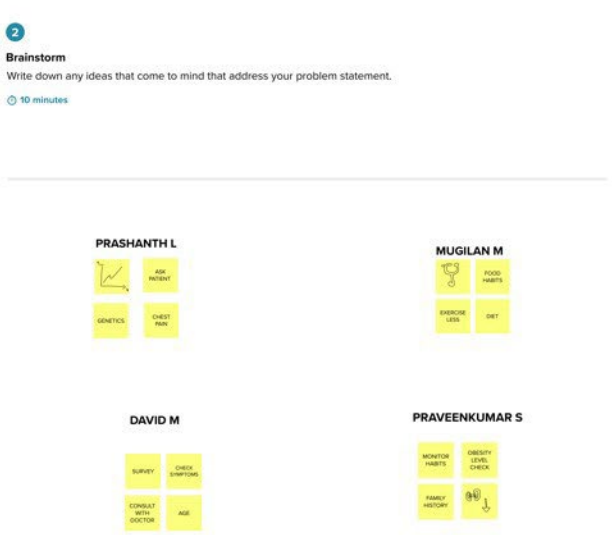
As a doctor. I'm trusted this?
It's necessary for me
how to choose which is the best for me
THINK AND FEEL?
HEAR?
Discuss other doctors to make decisions
Use Artificial Intelligence for analyze test results
Trust your own decisions and knowledge
What do they
SEE?
SAY AND DO?
Observes in big hospitals
Check the websites
List Out Pros and Cons
PAIN
Predicting Wrong Results in the Analysis
GAIN
Optimal solution for predicted results

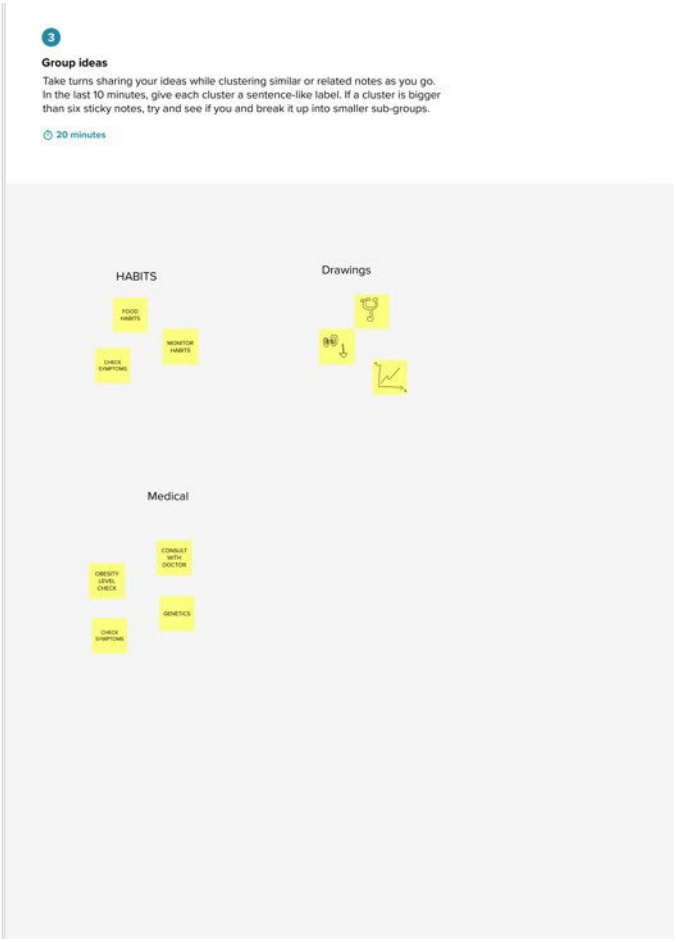
3.2 Ideation and Brainstorming

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

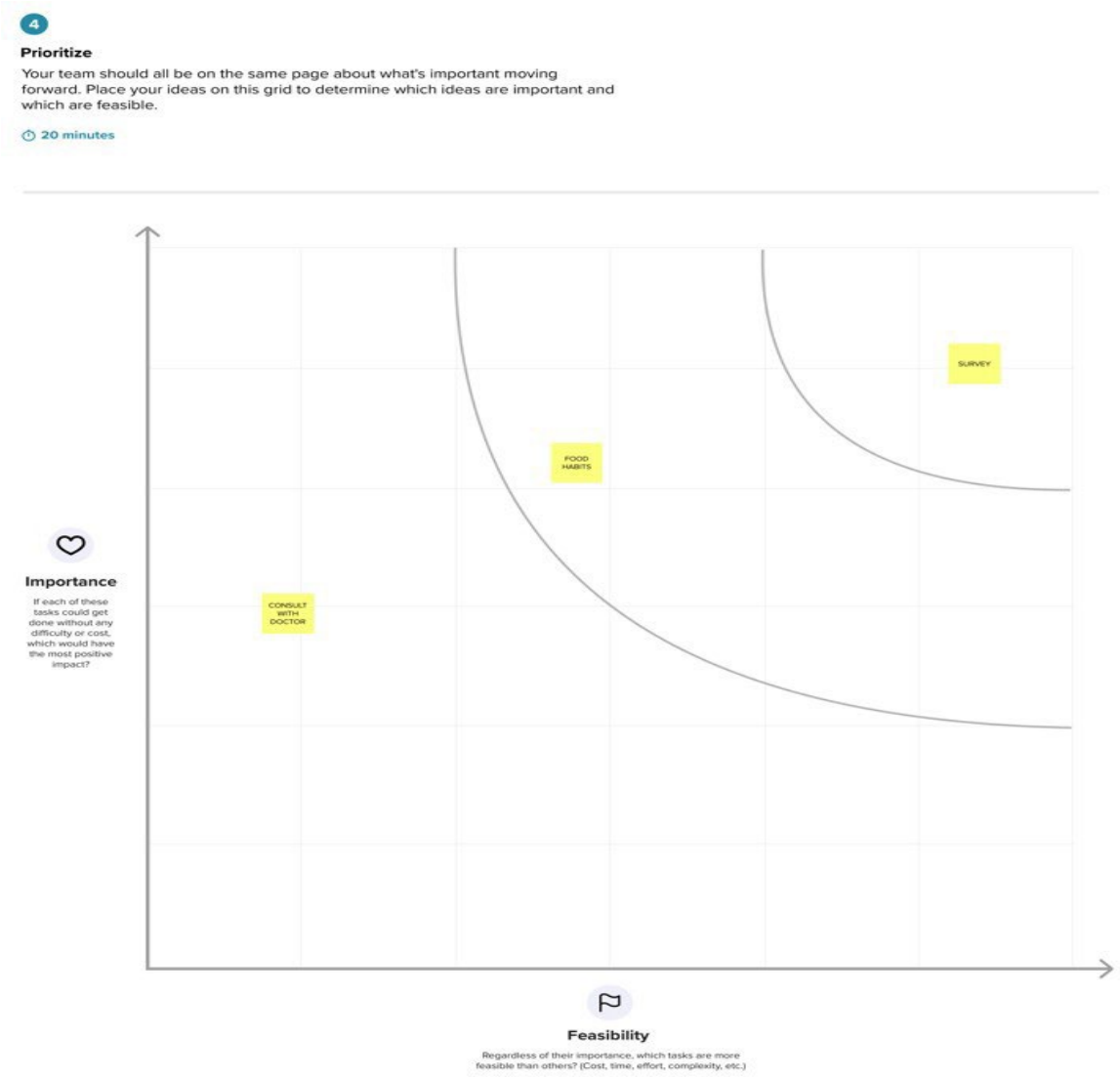


Step-2: Brainstorm, Idea Listing and Grouping





Step-3: Idea Prioritization



3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Predicting the heart disease of a patient by analyzing Past or historical records. Where patient can get cure Or take necessary treatment before the disease is affected.Also, to develop am interactive dashboard to predict the disease accurately with few tests and attributes.
2.	Idea / Solution description	Analyzing data and identifying the heart disease using Cognos analytics.

		To predict the heart disease at the beginning stage and provide treatment for speedy recovery
3.	Novelty / Uniqueness	Comparing other models the prediction will vary. But our model will predict accurately and give effective results
4.	Social Impact / Customer Satisfaction	By this project people can able to diagnose the heart disease at initial stage by themselves
5.	Business Model (Revenue Model)	By subscription technique, one user will be allowed to predict the disease.
6.	Scalability of the Solution	In future, some more health associated prediction will be added with the same interactive dashboard.

3.4 Problem Solution Fit

4. Requirement Analysis

4.1 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	Enables user to make registration for the application through Gmail
FR-2	User Confirmation	Once after registration, the user will get confirmation via Email
FR-3	Visualizing Data	User can visualize the trends on the heart disease through Dashboard created using IBM Cognos Analytics
FR-4	Generation Report	User can view his/her health report and can make decisions accordingly

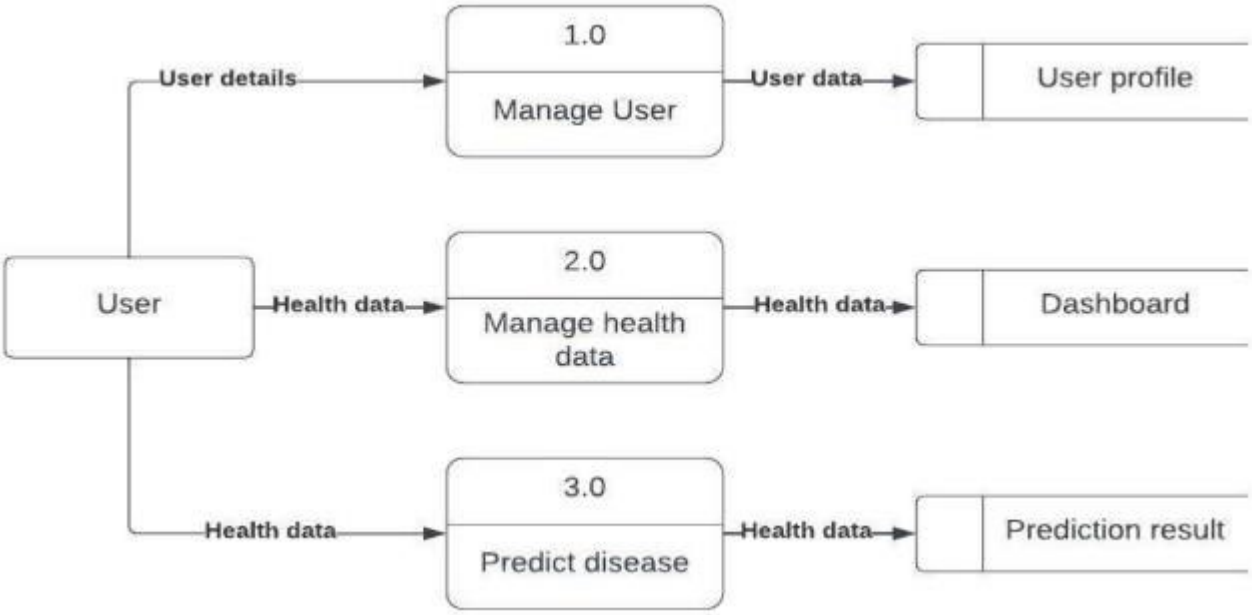
4.2 Non-Functional Requirement

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application will have a simple and userfriendly graphical interface. Users will be ableto understand and use all the features of the application easily. Any action has to be performed with just a few clicks
NFR-2	**Security**	For security of the application the technique known as database replication should be used so that all the important data should be kept safe. Incase of crash, the system should be able to backup and recover the data
NFR-3	**Reliability**	The application has to be consistent at every scenario and has to work without failure in any environment
NFR-4	**Performance**	Performance of the application depends on the response time and the speed of the data submission. The response time of the application is direct and faster which depends on the efficiency of implemented algorithm
NFR-5	**Availability**	The application has to be available 24 x 7 for users without any interruption
NFR-6	**Scalability**	The application can withstand the increase in the no. of users and has to be able to develop Higher versions

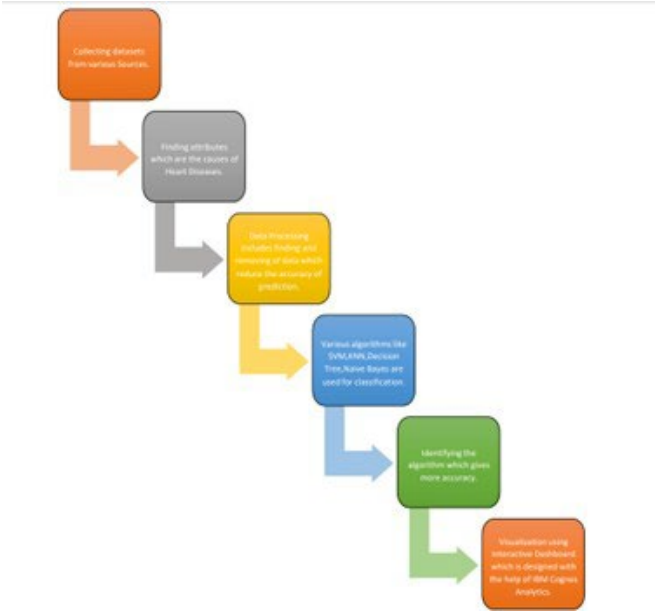
5. Project Design

5.1 Data Flow Diagram



5.2 Solution and Technical Architecture

5.3



6. Project Planning and Scheduling

6.1 Script Planning and Execution

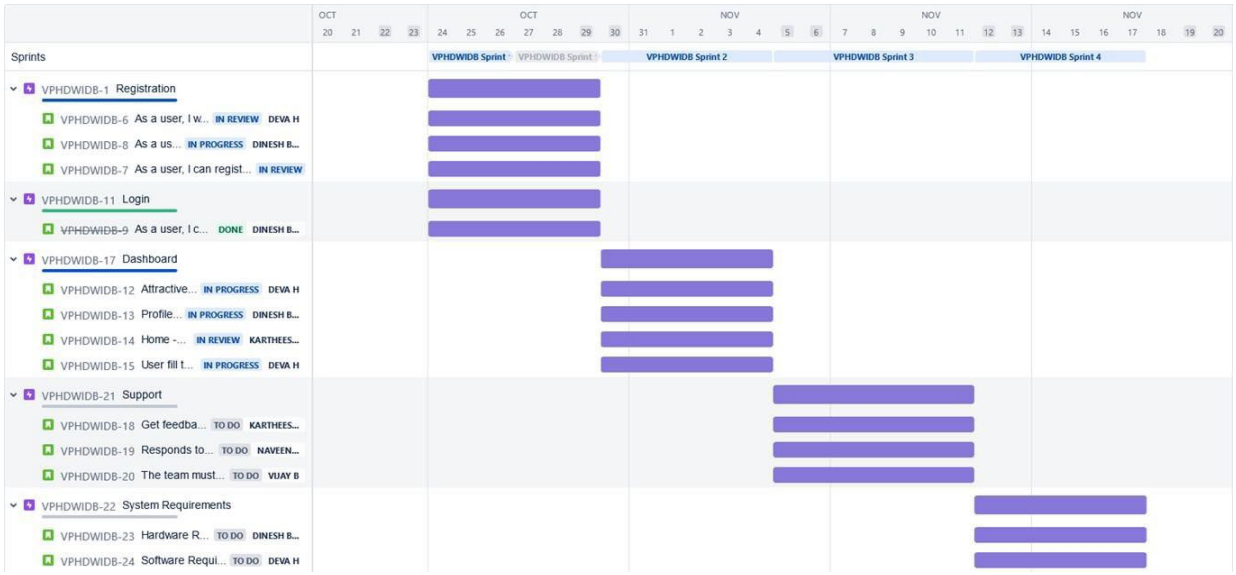
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	3	High	1
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	3	High	3
Sprint-1		USN-3	As a user, I can register for the application through Gmail	3	Medium	1
Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	6	High	5
Sprint-2	Dashboard	USN-5	Attractive dashboard For the Application	3	Medium	3
Sprint-2		USN-6	Profile - view & update your profile	5	Low	2
Sprint-2		USN-7	Home - Analyze your Heart problem	2	High	4
Sprint-2		USN-8	The user will have to fill in the below 13 fields for the system to predict a disease -Age in year -Gender -Chest pain Type -Fasting Blood Sugar -Resting Electrographic Results -Exercise Induced Angina -Trust Blood Pressure	7	High	2
Sprint-3	Support	USN-9	Get feedback from users	10	Medium	3
Sprint-3		USN-10	Responds to user queries via telephone,email etc.	3	Medium	2
Sprint-3		USN-11	The team must respond immediately to the queries based on the priority	5	High	5
Sprint-4	System Requirements	USN-12	Hardware Requirement 3. Laptop or PC • i5 processor system or higher • 4 GB RAM or higher • 128 GB ROM or higher 4. Mobile • (12.0 and above)	5	Low	2
Sprint-4		USN-13	Software Requirement 2. Laptop or PC	8	Medium	4

			• Windows 10 or higher • Android Studio			

6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	30 Oct 2022	04 Nov 2022	17	04 Nov 2022
Sprint-3	20	6 Days	05 Nov 2022	11 Nov 2022	18	11 Nov 2022
Sprint-4	20	6 Days	12 Nov 2022	17 Nov 2022	19	17 Nov 2022

6.3 Jira Report



7 Coding And Solutioning

7.1 Machine Learning

Learning which model is best for the given Dataset

```
data = {'Estimators':['Linear Regression',
                      'Logistic Regression',
                      'Gaussian Naive Bayes',
                      'K-Nearest Neighbor',
                      'Random Forest',
                      'Bagging Decision Tree',
                      'Hard coting classifier'],
        "Accuracy":[r2,
                    LogisticRegressionScore,
                    gauss_score,
                    KNC_accuracy,
                    rnd_clf_accuracy,
                    bag_clf_accuracy,
                    voting_clf_accuracy]
        }

data = pd.Dataframe(data)

data.sort_values('Accuracy', ascending='False')
```

	Estimators	Accuracy
0	Linear Regression	0.447140
5	Bagging Decision Tree	0.772727
6	Hard coting classifier	0.772727
3	K-Nearest Neighbor	0.795455
4	Random Forest	0.795455
1	Logistic Regression	0.818182
2	Gaussian Naive Bayes	0.818182

From the Above results we can conclude that Logistic Regression and Gaussian Naive Bayes gives Highest accuracy than others for this particular data set . In our convinient we choose the Logistic Regression

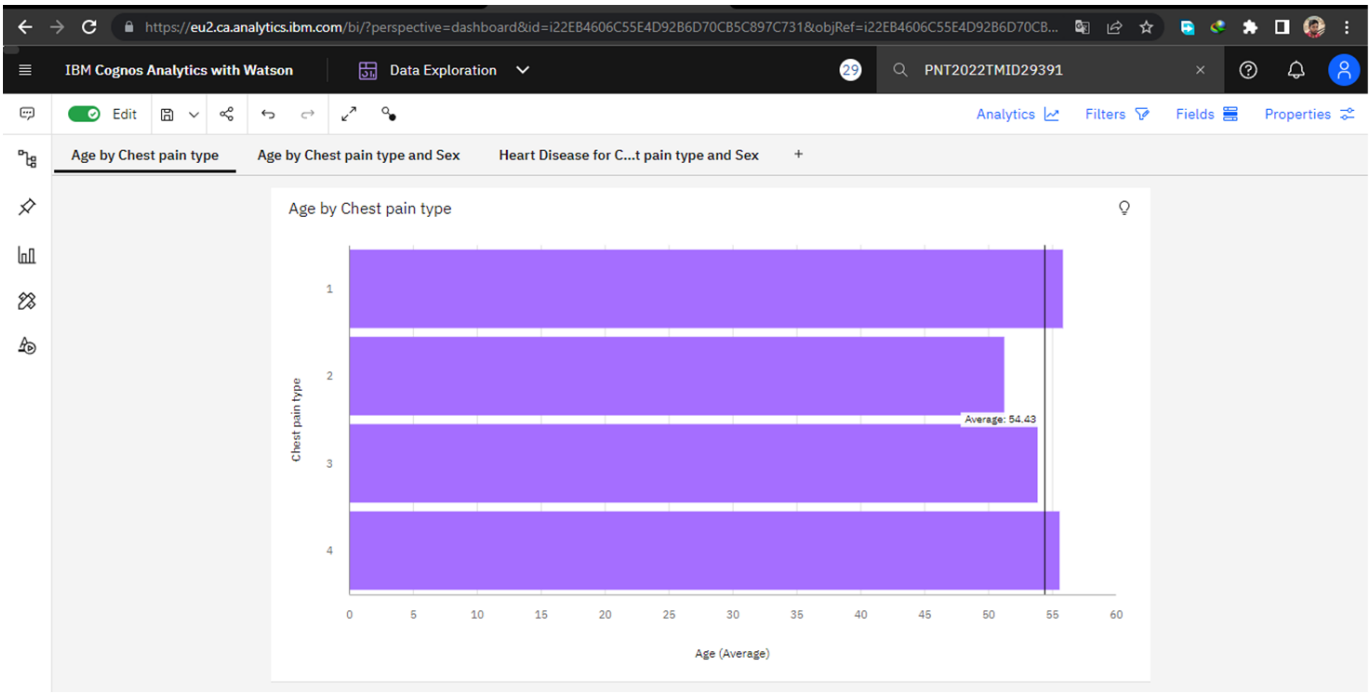
Comparing it with the accuracy gotten from Decision Tree:

```
TP=cm[0][0] #cm=Confusion Matrix
TN=cm[1][1]
FN=cm[1][0]
FP=cm[0][1]
print('Testing Accuracy for Decision Tree:',(TP+TN)/(TP+TN+FN
+FP))
print('Testing Sensitivity for Decision Tree:',(TP/(TP+FN)))
print('Testing Specificity for Decision Tree:',(TN/(TN+FP)))
print('Testing Precision for Decision Tree:',(TP/(TP+FP)))
```

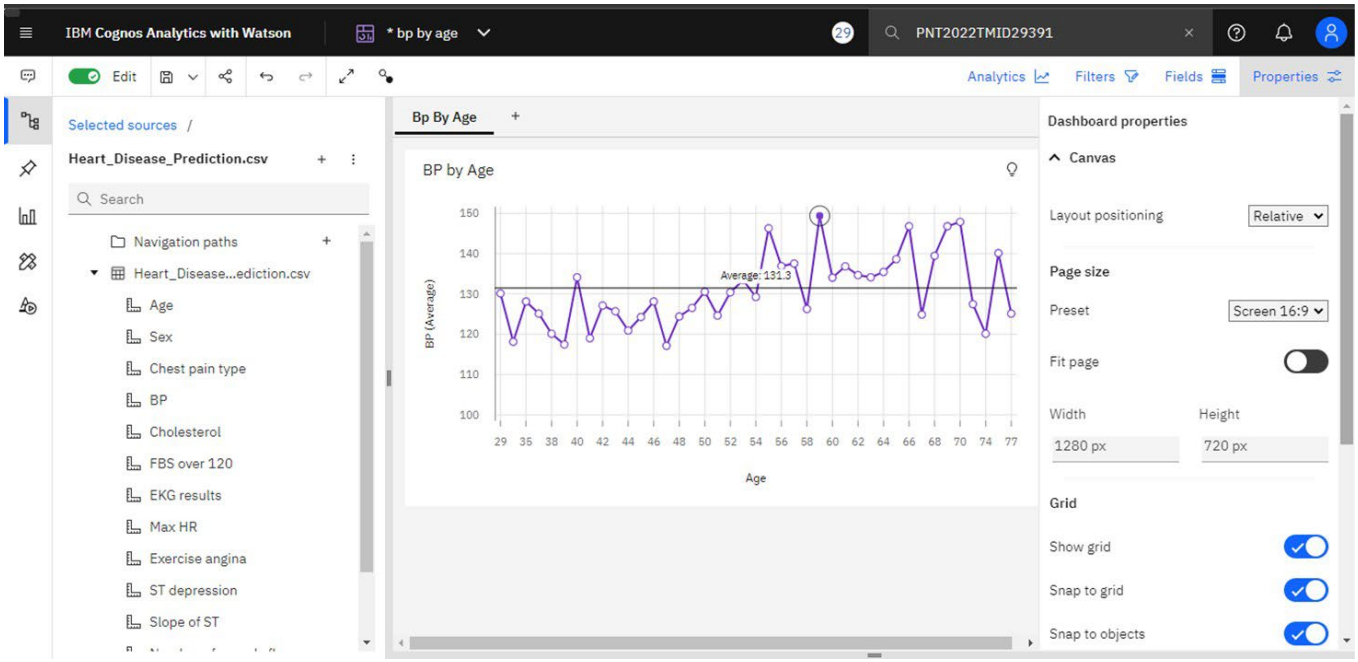
```
Testing Accuracy for Decision Tree: 0.9264705882352942
Testing Sensitivity for Decision Tree: 0.8888888888888888
Testing Specificity for Decision Tree: 1.0
Testing Precision for Decision Tree: 1.0
```

7.2 Dashboard

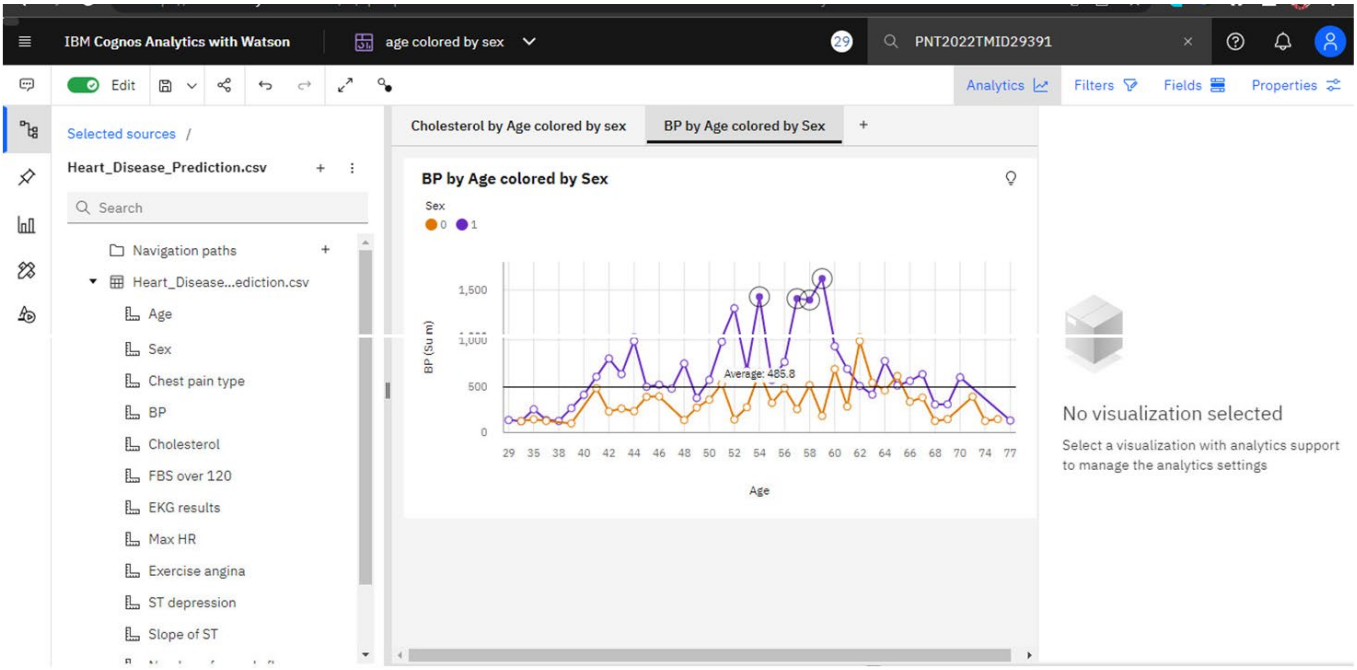
Age by Chest Pain Type



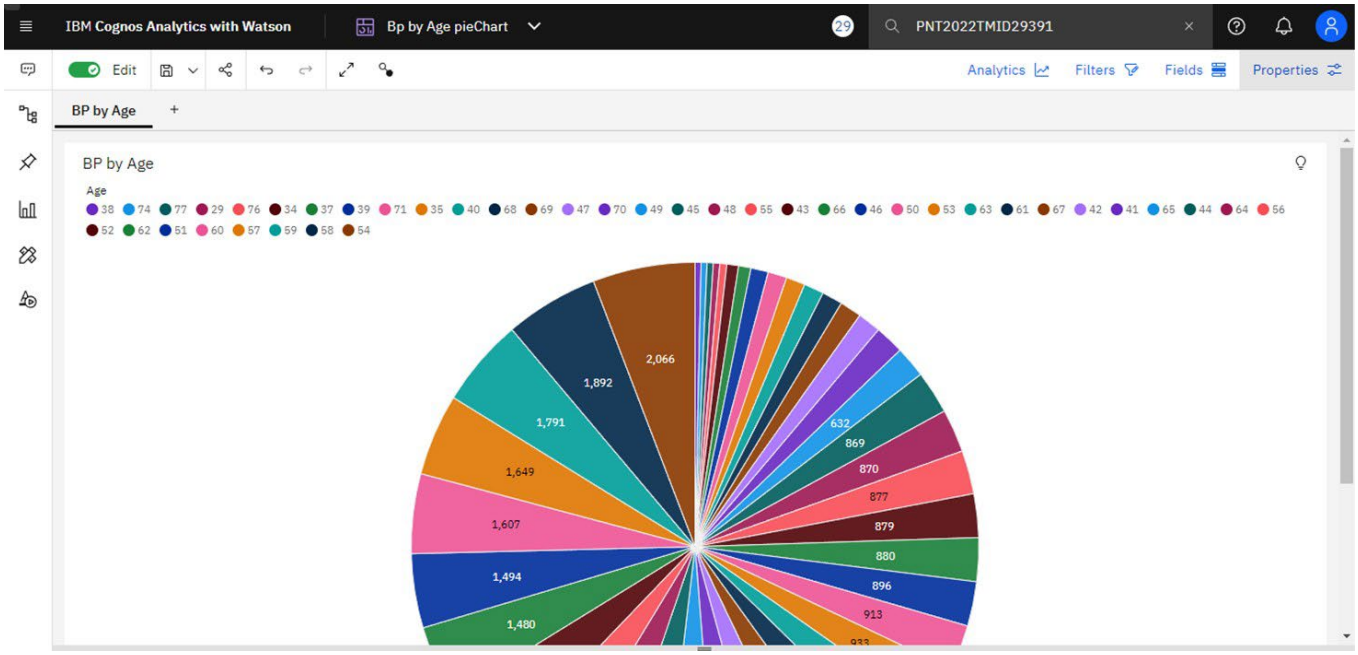
BP by Age Line



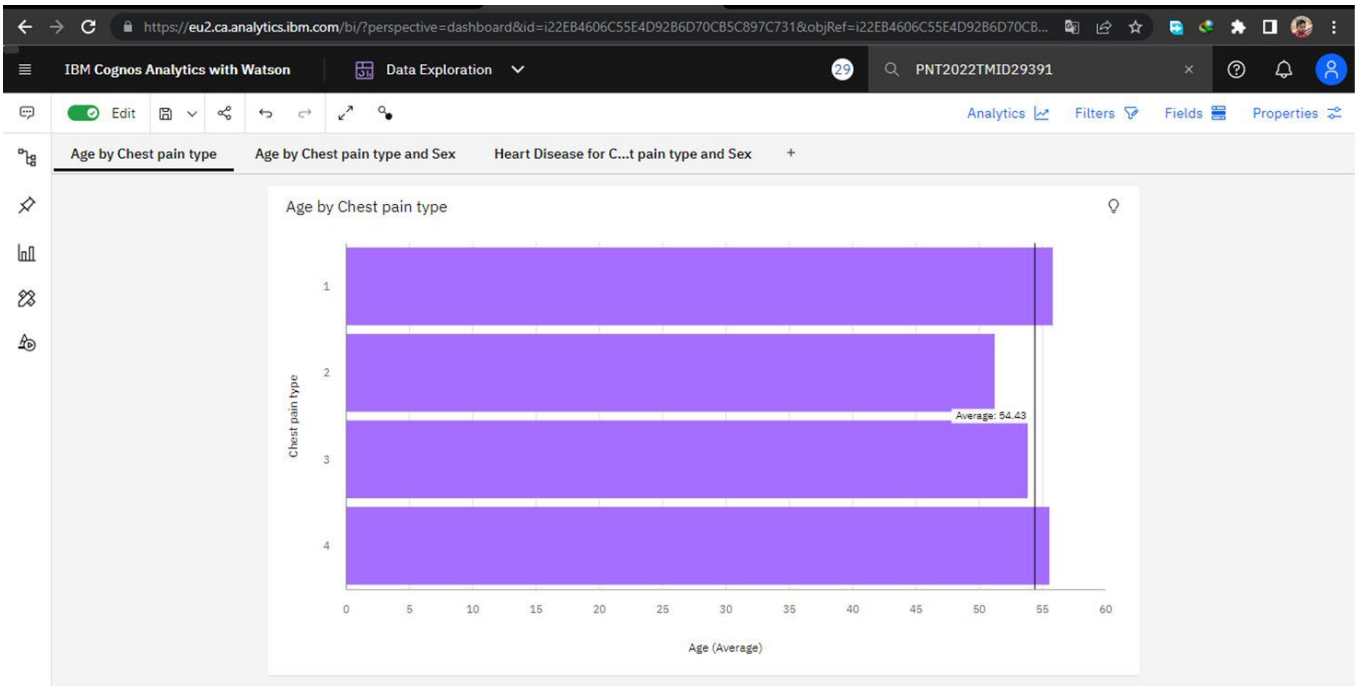
BP by Age coloured By sex



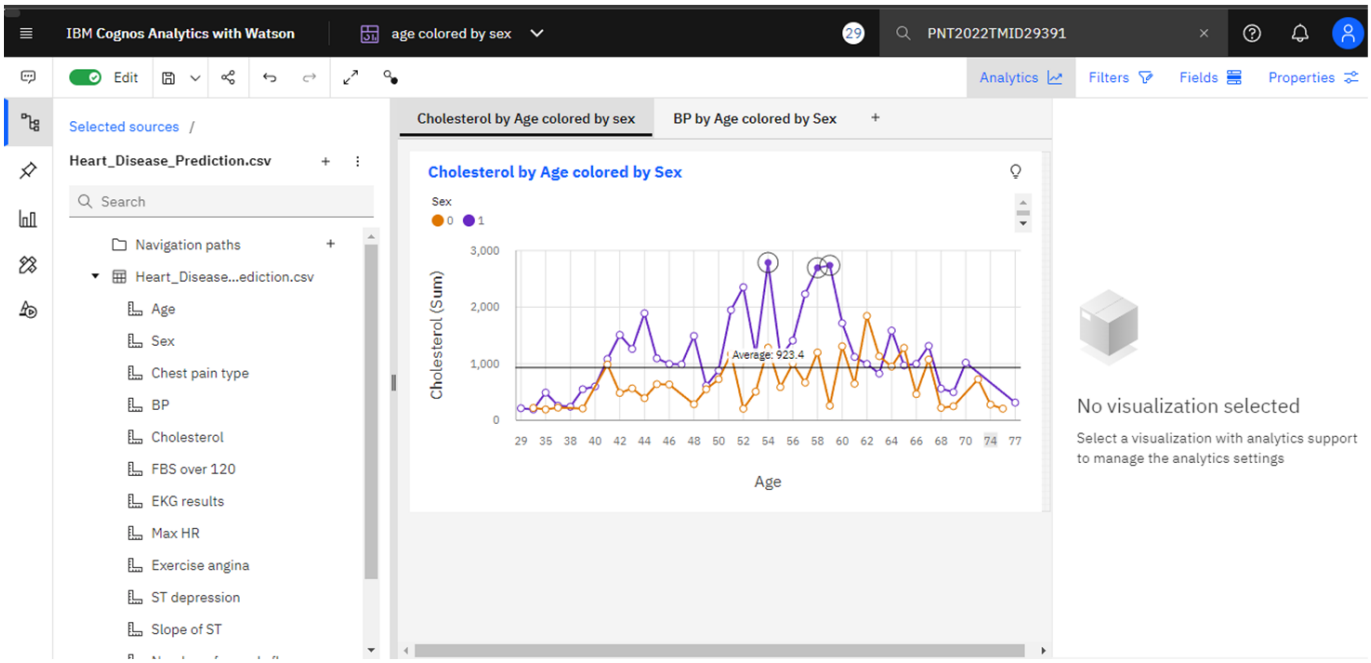
Bp by Age Pie-chart



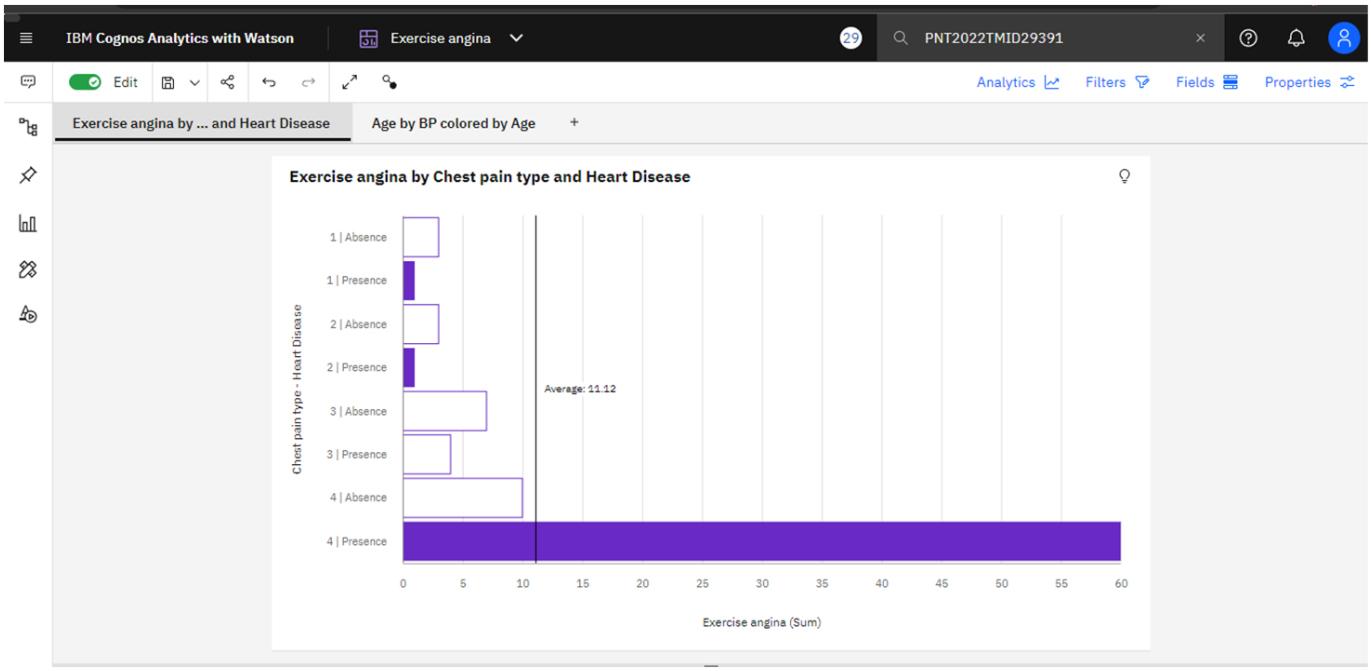
Age by chest pain type and sex



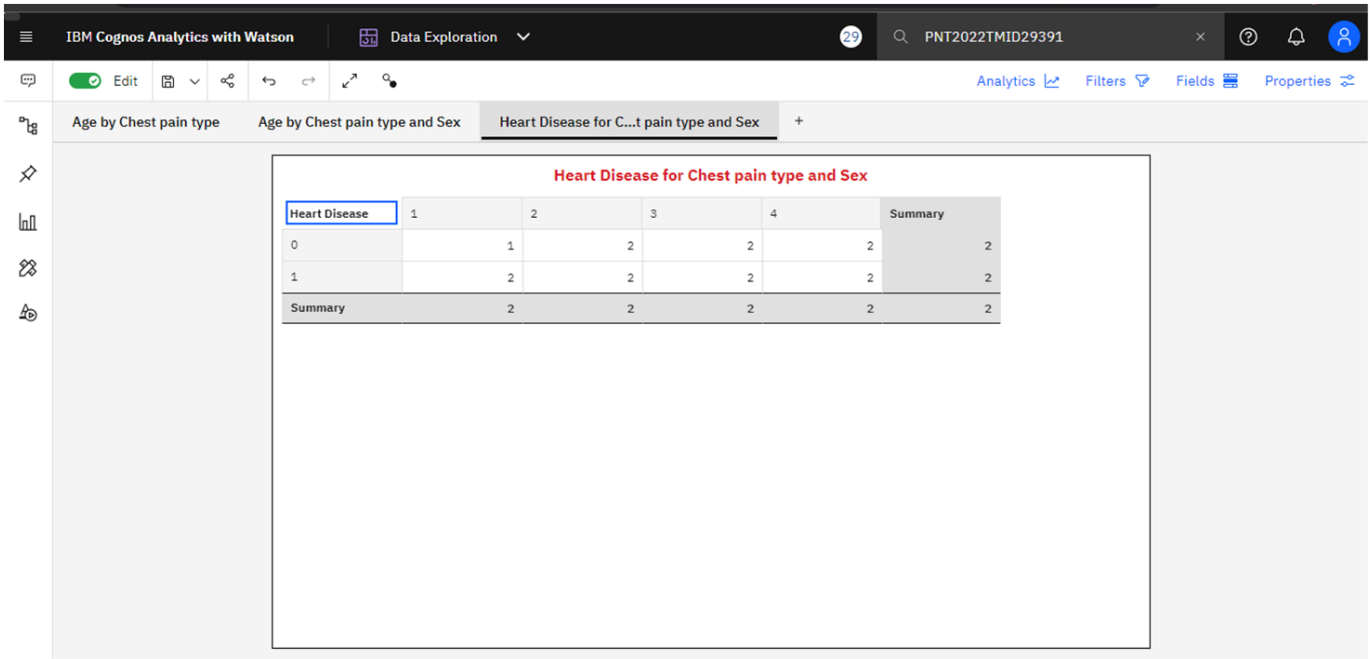
Cholestrol by age cloured by sex



Exercise angina by Chest pain type and Heart Diseases

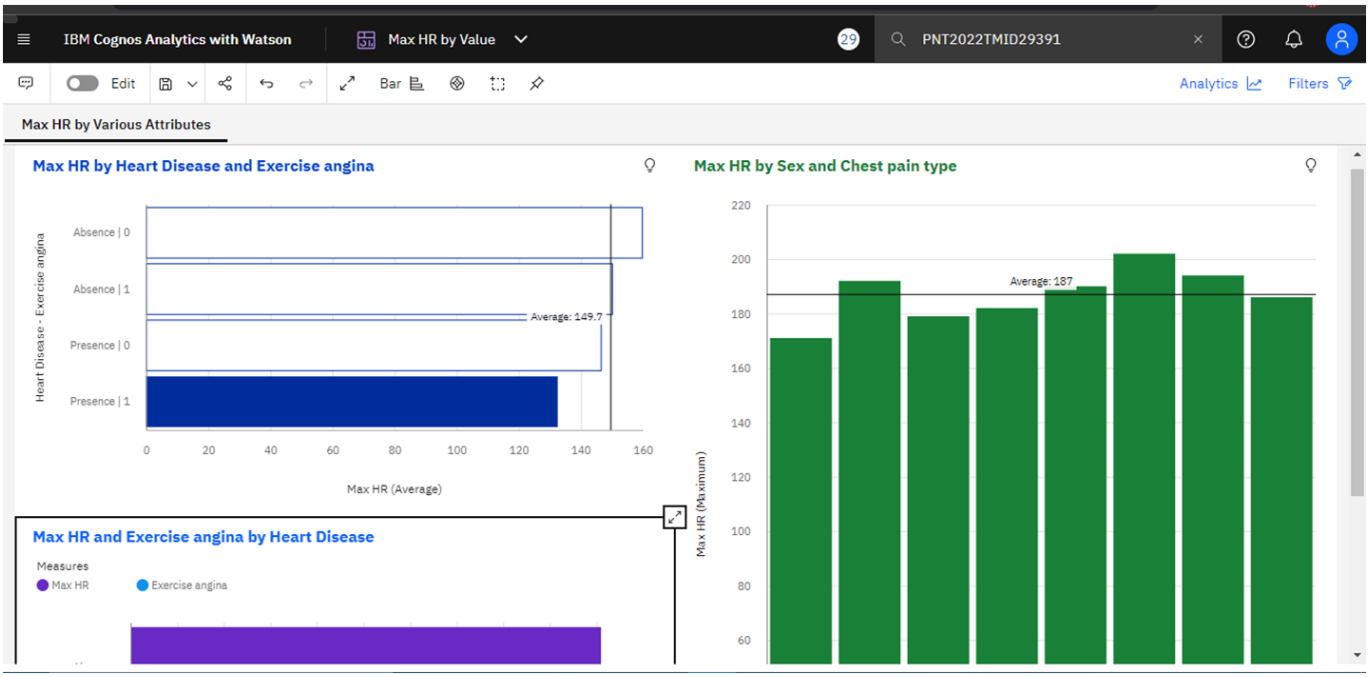


Heart Diseases for chest pain type and sex

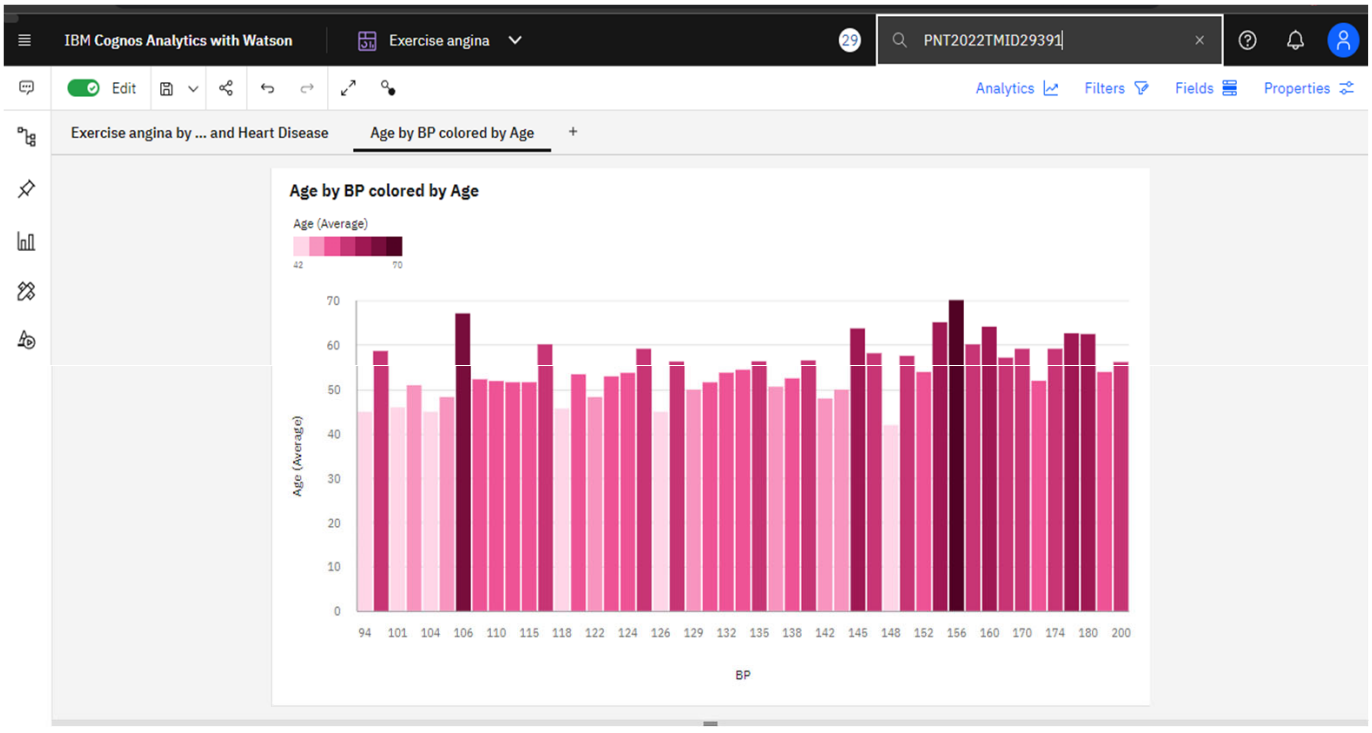


Heart Disease	1	2	3	4	Summary
0	1	2	2	2	2
1	2	2	2	2	2
Summary	2	2	2	2	2

Showing Different Types Visuals



Age by BP coloured by Age



8. Testing

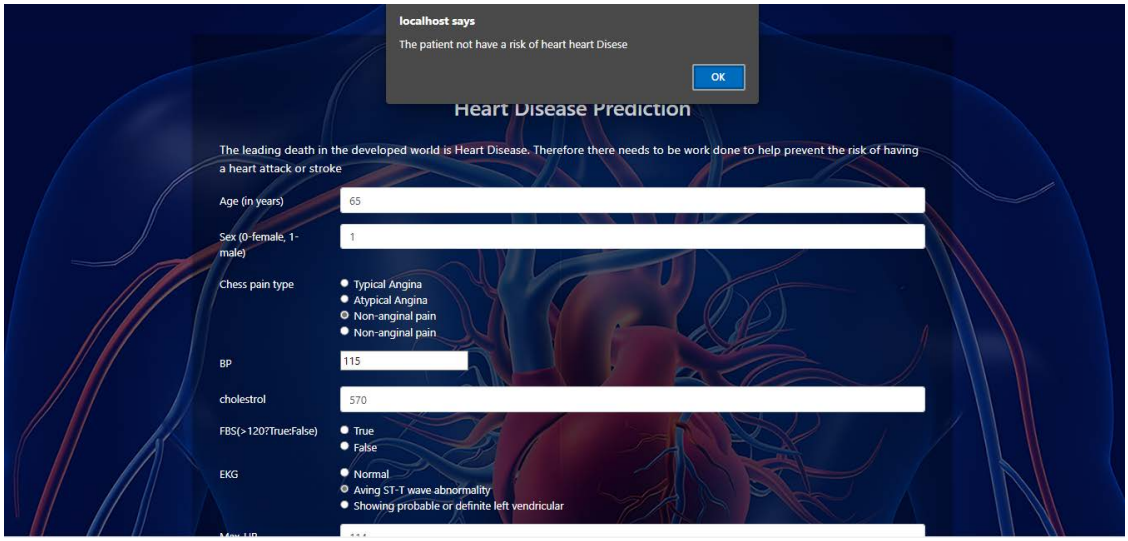
8.1 Test Cases

Testing the data model for various input values.

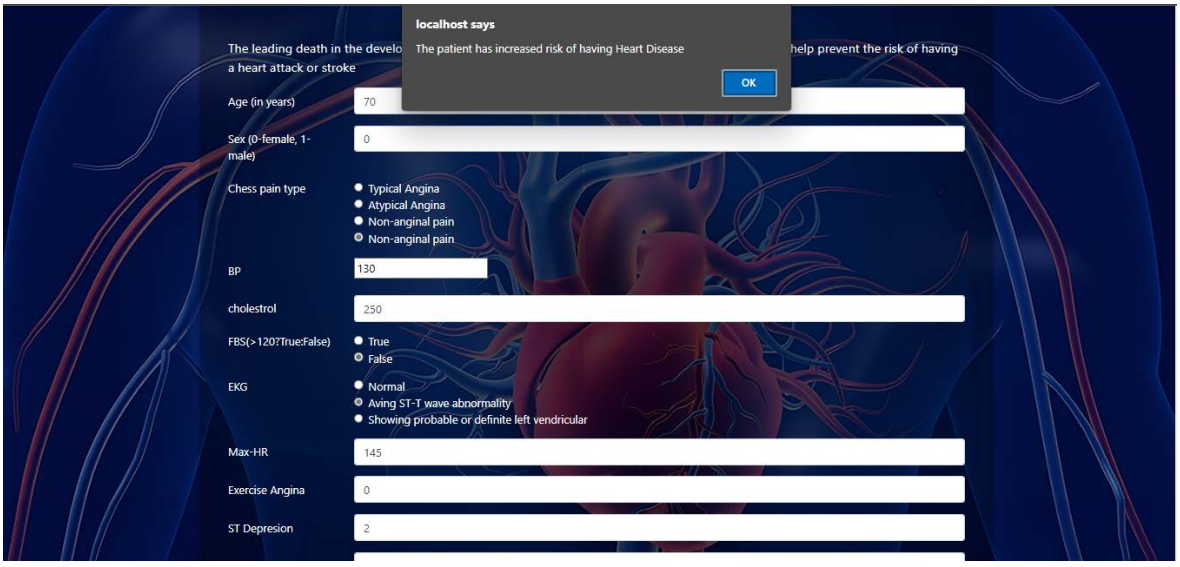
```
In [ ]: from sklearn.metrics import accuracy_score
        input=(63,1,3,145,200,150,98,0,0,0,0,0,0)
        input_as_numpy=np.asarray(input)
        input_reshaped=input_as_numpy.reshape(1,-1)
        pre1=tree_model.predict(input_reshaped)
        print(pre1)
        a1 = accuracy_score(pre1,model1.predict(input_reshaped)) * 100
        print(a1)

        ['Absence']
        100.0

In [ ]: from sklearn.metrics import accuracy_score
        input=(70,1,4,130,322,0,2,109,0,2.4,2,3,3)
        input_as_numpy=np.asarray(input)
        input_reshaped=input_as_numpy.reshape(1,-1)
        pre1=tree_model.predict(input_reshaped)
        print(pre1)
        a1 = accuracy_score(pre1,model1.predict(input_reshaped)) * 100
        print(a1)

        ['Presence']
        100.0
```

8.2 User acceptance Testing

Testing a case where user not have heart disease



Testing a case where user does have heart disease



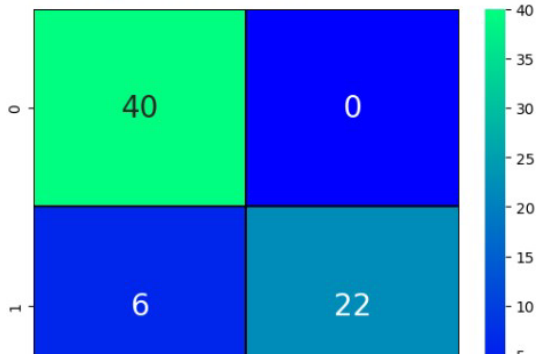
9. Result

9.1 Performance Metrics

The confusion matrix below shows the performance metrics of the machine learning model.

```
from sklearn.model_selection import RandomizedSearchCV
from sklearn.tree import DecisionTreeClassifier

tree_model = DecisionTreeClassifier(max_depth=5,criterion='entropy')
cv_scores = cross_val_score(tree_model, x, y, cv=10, scoring='accuracy')
m=tree_model.fit(x, y)
prediction=m.predict(X_test)
cm= confusion_matrix(y_test,prediction)
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3, linecolor='black',annot_kws={"size": 20})
print(classification_report(y_test, prediction))

TP=cm[0][0]
TN=cm[1][1]
FN=cm[1][0]
FP=cm[0][1]
print('Testing Accuracy for Decision Tree:',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Decision Tree:',(TP/(TP+FN)))
print('Testing Specificity for Decision Tree:',(TN/(TN+FP)))
print('Testing Precision for Decision Tree:',(TP/(TP+FP)))
```

```
              precision    recall  f1-score   support

      Absence       0.87      1.00      0.93        40
     Presence       1.00      0.79      0.88        28

     accuracy                           0.91        68
    macro avg       0.93      0.89      0.91        68
 weighted avg       0.92      0.91      0.91        68

Testing Accuracy for Decision Tree: 0.9117647058823529
Testing Sensitivity for Decision Tree: 0.8695652173913043
Testing Specificity for Decision Tree: 1.0
Testing Precision for Decision Tree: 1.0
```



10. Advantages Disadvantages

Advantages:

- This is one of the fastest ways to determine if a person is likely to suffer from a heart disease or not.
- Useful for medical practitioners to easily classify their patients.
- User Friendly
- Easy to understand
- Secure
- Dashboard provides insightful informations

Disadvantages:

- Needs work
- Users need to know all the fields
- Does Not take null value as input
- Does not provide suggestions to user

11. Conclusion

Complications of heart disease include heart attack and stroke. You can reduce the risk of complications with early diagnosis and treatment. So the suggestion that we get from the website might help save patients. It is always to get treated in the early stages of heart disease.

12. Future Scope

Like the saying goes “Prevention is better than cure”. We have to look into methods to prevent heart diseases altogether other than just predicting it in early stages.
To use this website we need to take a lot of tests beforehand. So it would be better if we require less attributes and still give an effective result

13. Appendix

Source Code:
https://github.com/IBM-EPBL/IBM-Project-22109-1659805164/tree/main/Visualizing%20and%20Predicting%20Heart%20Diseases/Final%20deliverables/Source%20Code

Demo Video link: