A

Project Report

on

**Coronary Heart Disease Prediction System**

Developed by

**Priyam Sheta(IT123),**

**DEPARTMENT OF IT, DD UNIVERSITY.**

**Meet Shingala(IT124),**

**DEPARTMENT OF IT, DD UNIVERSITY.**

**Guided By**

**Prof. Deepak C. Vegda,**

**Department of Information Technology,**

**Faculty of Technology,**

**DD University**



**Department of Information Technology**

**Faculty of Technology, Dharmsinh Desai University**

**College Road, Nadiad-387001.**

**Candidate’s Declaration**

We declare that pre-final semester report entitled “Coronary Heart Disease prediction system” is our own work conducted under the supervision of the guide Prof. Deepak C. Vegda.

We further declare that to the best of our knowledge the report for B.Tech. VI semester does not contain part of the work which has been submitted either in this or any other university without proper citation.

Candidate’s Signature

Candidate’s Name

Student ID

Candidate’s Signature

Candidate’s Name

Student ID

## DHARMSINH DESAI UNIVERSITY

## NADIAD-387001, GUJARAT



## CERTIFICATE

This is to certify that the project entitled “**Coronary Heart Disease Prediction System”** is a report of the work carried out by

1) **Mr. Priyam Sheta**, Roll No: IT123, Student ID No: **18ITUOS075**

2) **Mr. Meet Shingala**, Roll No: IT124, Student ID No:**19ITUOD005**

of Department of Information Technology, semester VI, under the guidance and supervision of Prof. Deepak C. Vegda. They were involved in Project work during

academic year 2020-2021.

**Prof. Deepak C. Vegda**

(Project Guide)

Department of Information Technology,

Faculty of Technology,

Dharmsinh Desai University, Nadiad.

Date:

**Prof. Vipul Dabhi**

Head, Department of Information Technology,

Faculty of Technology,

Dharmsinh Desai University, Nadiad.

Date:

**ACKNOWLEDGEMENT**

It’s a great honor for us to sincerely thank many people without whose kindness and support this project may not have reach to the completion.

We have made great efforts in completing this Project on “**Coronary Heart Disease Prediction System”**.

However, it might not have been possible without our guide **Prof. Deepak C. Vegda**. Our sincere thanks to him. We are highly indebted to **Prof. Deepak C. Vegda** for his guidance and constant supervision as well as for providing necessary information regarding the project & also for his support, kind co-operation and encouragement which helped us in completion of this project. We would not forget to remember **Dr. Vipul Dabhi** for his encouragement and more over for his timely support and guidance until the completion of our project work.

Last but not the least we are also thankful to all friends & those who

have contributed directly or indirectly by giving their suggestions or even a advice for the completion of this project.

With sincere regards,

**Priyam Sheta**

**Meet Shingala**

**Table of content**

**Chapter 1:- Introduction 2**

1.1 Project Overview 2

1.2 Purpose 3

1.3 Scope 3

1.4 Objective 3

**Chapter 2:- Project Management 4**

2.1 Feasibility study 4

2.1.1 Technical Feasibility 4

2.1.2 Time Schedule Feasibility 4

2.1.3 Operational Feasibility 4

2.2 Project Planning 5

2.2.1 Project development Approach 5

2.3 Project Scheduling 5

**Chapter 3:- System requirement study 6**

3.1 Study of current system 6

3.2 Problems and weaknesses of the current system 6

3.3 User characteristics 6

3.4 Hardware and Software requirements 7

**Chapter 4:- System Analysis 8**

4.1 Functional requirements 8

4.2 Non-Functional requirements 8

4.2 Use-Case Diagram 9

**Chapter 5: Implementation planning 10**

5.1 Implementation environment 10

5.2 Installing necessary libraries 10

5.3 XAMPP and phpMyAdmin installation 11

**Chapter 6: Handling the data 12**

6.1 Removing the unnecessary data 12

6.2 Removing the NaN data 13

6.3 Avoiding overfitting 14

6.3.1 Balancing the distribution 14

6.3.2 Shuffling the data 15

6.4 Dividing the data 15

**Chapter 7: Post Model Training 17**

7.1 Getting a good test accuracy 17

7.2 Selecting Threshold value 18

7.2.1 What is Threshold value? 18

7.2.2 Decision Tree 18

**Chapter 8: Testing 21**

8.1 Introduction 21

8.2 Unit testing 21

8.3 Integration testing 21

8.4 Validation testing 21

8.5 GUI testing 22

8.6 Test cases 22

**Chapter 9: User Manual 23**

**Chapter 10: Limitation and Future Enhancements 29**

10.1 Limitations 29

10.2 Future Enhancements 29

**Chapter 11: Conclusion and Discussion 30**

11.1 Conclusion 30

11.2 Discussion 31

11.2.1 Self-Analysis 31

11.2.2 Summary 31

**Chapter 11: Bibliography** **32**

**Abstract**

We can implement website for predicting the coronary heart disease in the following 10 years. It can be used by the users that are frequently having a heart check-up and are a heart patient. After the Machine Learning model is ready, research of the amount a particular factor is contributing towards the disease is also done. The proposed solution is to develop a system where the users can easily by entering their details can know their heart condition and the history can also be checked.

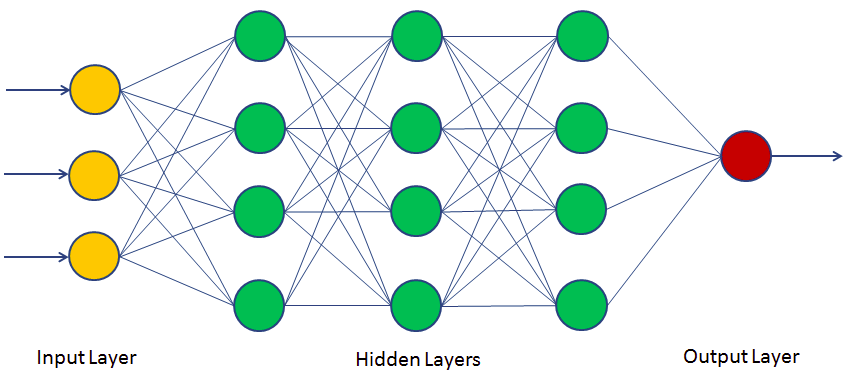
**Chapter 1:- Introduction**

**1.1 Project Overview**

🡪 We have created a website which will work with a Machine Learning model to predict the presence of the coronary heart disease. Our website will take the information of the user and will give the probability of the disease. If the user cannot afford the consultancy of the doctor, then he can check their heart conditions on this website. All the history of the user can also be stored and can be showed to the user on request. If the user is getting a high risk then all the doctors in his city is also suggested to him. An accurate model is trained by the Neural Network.

**What is a Neural Network?**

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature. Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria.



**1.2 Purpose**

🡪 The purpose of making this system is to facilitate the users. Heart diseases are increasingly continuously from the last decade. Among them the coronary heart disease is also increasing. Our heart gets pure blood and oxygen through arteries and veins. Due to the swelling of these veins, heart does not get enough amount of blood and oxygen. That is basically known as coronary heart disease. Different factors are involved in this disease for example, Cholesterol, Blood Pressure, Strokes and etc. We all are aware of the fact that the number of patients of age above 30 are more prominent to heart diseases. Also, since the availability of the doctors is very less and many people cannot afford the consultancy, few people do not get regular check-ups. Hence this system will help the users to check their heart conditions easily.

**1.3 Scope**

🡪 The use of the system is not limited to any particular group of users. Any user from anywhere can access it and their data is stored by the admin in the database. The user can freely check their previous check-ups and easily get the new prediction by the system.

**1.4 Objectives**

🡪 To give an accurate prediction of heart disease.

🡪To study the factors affecting the disease and their contribution towards getting the heart disease.

🡪 To provide an interactive web application for the users.

🡪 To give the users their history of their check-ups.

🡪 To suggest the doctors to the user having high risk.

**Chapter 2:- Project Management**

**2.1 Feasibility study**

**2.1.1 Technical feasibility**

🡪 A technical feasibility study evaluates the details of how you intend to build a system or solution. The technical feasibility means the study or evaluation of current equipment, existing software technology and the current knowledge that we are going to use to provide the solution or system. Technical analysis evaluates technical merits of the system at the same time collects additional information about performances, reliability, maintainability and productivity.

**2.1.2 Time schedule feasibility**

🡪 The project entirely depends on the time it takes to train the model with a satisfied accuracy. But the time allotted to us is enough to train the model and deploy it onto the website.

**2.1.3 Operational feasibility**

🡪 As mentioned above, all type of users can operate this system and check their heart condition. There are 2 types of user:

1) Common user 2) Admin

A database is developed to store the values entered by the user. All the values entered by the users will be stored in the database.

**2.2 Project Planning**

**2.2.1 Project development approach and justification**

🡪 For our project development the Iterative waterfall model is used. It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

This model divides the cycle into the phases mentioned below:

1. Feasibility Study

2. Requirement analysis and specification

3. Design

4. Coding and unit testing

5. Integration and system testing

6. Maintenance

**2.3 Project Scheduling**

🡪 Scheduling the project tasks is an important project planning activity. It involves deciding which task should be taken up and when. We will proceed in this way.

1. Finding all the tasks needed to complete the project.

2. Converting large tasks into smaller activities.

3. Determine dependencies among different activities

4. Establish most likely estimates for the time duration necessary to complete the activities.

**Chapter 3:- System Requirement study**

**3.1 Study of current system**

🡪 Actually, there is no system right now that is working on the values given by the user. There can be a system that works on the images of the heart but images cannot be true every time. Hence a system that worked on numbers will be very helpful to the users.

**3.2 Problems and weaknesses of current system**

🡪 The current system is working on images but it has been found that the prediction on images has many false negatives. It can be very daunting as the user feels that they are not having any disease but finally end up getting the disease. And the user has to have a scan of his heart for using this system and that can be costly every time when they want to check the disease. Hence the system we are going to prepare will overcome all the problems. The use of direct numbers will provide greater accuracy to the user.

**3.3 User characteristics**

🡪 There are 2 types of users in our system

1) Common user

2) Admin

User can enter the values of their heart and check their history. Admin can maintain the database.

**3.4 Hardware and software requirements**

🡪 Computer requirements are

RAM 4 GB, Storage 512SSD, I3 Processor.

🡪 Software requirement are

Python libraries mentioned below.

**Chapter 4:- System Analysis**

**4.1 Functional Requirements**

1) User can create the account on the website by signing up or can also login to the website if account is there.

2) User can check the history of their previous check-up. All the details with the date and time will be displayed in the history section.

3) User can know all the doctors in his city if he is getting a high risk of the heart disease.

4) User can check the probability of getting the coronary heart disease by entering their latest conditions.

5) User can know their BMI by entering their weight and height if they are not aware of their current BMI.

6) User can freely contact us in case of any problems.

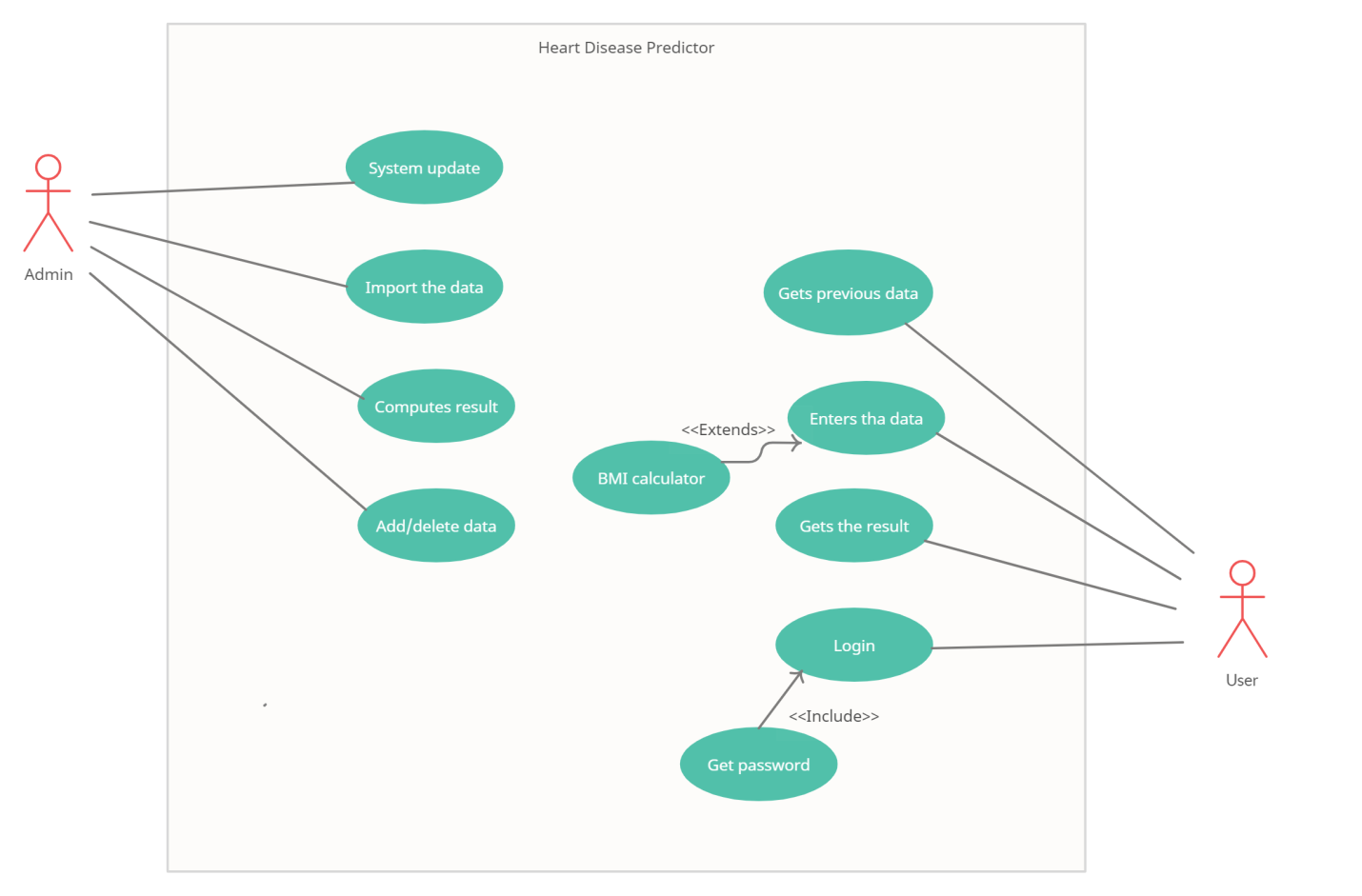
**4.2 Non Functional Requirements**

1) Users are requiring more and more accurate results and hence the prediction must be done accurately.

2) User must be able to use the interface at any place and hence it must be portable.

3) For prediction, always new features will be raised up. Hence the system must be maintained and updated regularly.

**4.3 Use-Case Diagram**

****

[4.3.1 Use Case Diagram]

**Chapter 5:- Implementation Planning**

**5.1 Implementation environment**

🡪 The project has been completely made on Jupyter notebook. Notebooks as well as the python 3 files can be easily created and maintained. This environment is chosen because every part of the code can be run easily and independently. No use of running the whole code is required.

**5.2 Installing Libraries**

🡪 For a Machine learning model to run and get deploy itself in the website needs many in-built libraries.

**Numpy and Pandas:**

**🡪** These 2 libraries are the most basic libraries. Numpy helps with working with the data by easily converting them into arrays, easily multiplying the arrays, etc. On the other hand, Pandas is used to load the dataset into the notebook.

**TensorFlow:**

It is a library that is used to create the layers of the neural network and finally running the model. It does the back-propagation of the neural network itself.

**Flask:**

Flask is used to deploy the trained Machine Learning model to the frontend website. All the user entered values are obtained in the flask file, prediction is done in the flask file, database is linked to the website and every routes are maintained in the flask.

**Pickle:**

Pickle is used for storing the parameters trained by the neural network and hence use that parameters in any file.

**Matplotlib and Seaborn:**

🡪 All the data comparison and the research analysis is done by using graphs that are easily done these 2 libraries. Final conclusions can be made by these libraries.

**SkLearn:**

🡪 This module helps to remove the NaN values in the data and hence cleans the data.

**SMOTE:**

🡪 This module is used to balance the unbalanced distribution of the data. It prevents the overfitting problem to occur.

**ROC Curve:**

🡪 This module is used to get us the true positives and the false positives and by which a good threshold value is selected.

**5.3 XAMPP and phpMyadmin installation**

🡪 The database that is used for the login system and the history section will be created on the XAMPP server and phpMyadmin. It is a very user friendly interface and the updation, deletion can be easily done on it.

**Chapter 6:- Handling the data**

**6.1 Removing the unnecessary data**

🡪 The features that are given in the data for the prediction must be a contributing feature towards the heart disease. But, eventually we grasped that out of all the columns, 1 column named “**education**” was not an important factor leading to a heart disease. Of course, the unnecessary data will be known after the research analysis but this column was one that was identified by us directly by seeing it.

So, finally the features available after that process are:

01) Gender

02) Age

03) Smoker?

04) Number of cigarette

05) BP medications?

06) Prevalent Stroke?

07) Prevalent Hypertensive?

08) Diabetes?

09) Cholesterol

10) Systolic BP

11) Diastolic BP

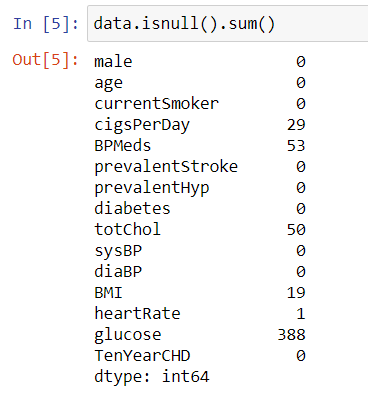
12) BMI

13) Heart rate

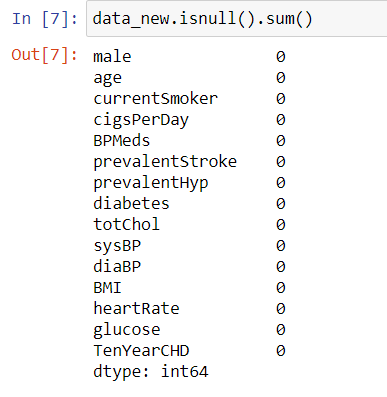
14) Glucose

**6.2 Removing the NaN data**

🡪 The NaN (not a number) data cannot be read by the machine learning model and hence there must be a solution to either remove that or converting them to a number that can be read. We can see the number of NaN values each feature is having.



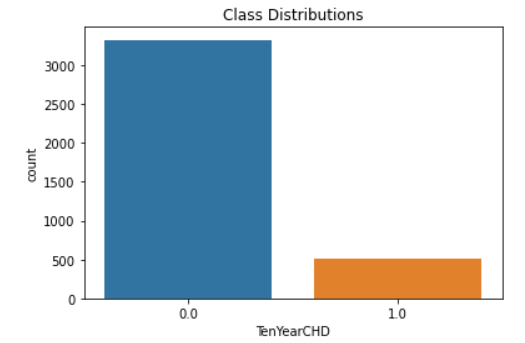
By the help of the SkLearn library as mentioned above we will fill all the NaN values with the average of all the values of that respective feature. Hence by doing so all NaN values are handled easily.



**6.3 Avoiding overfitting**

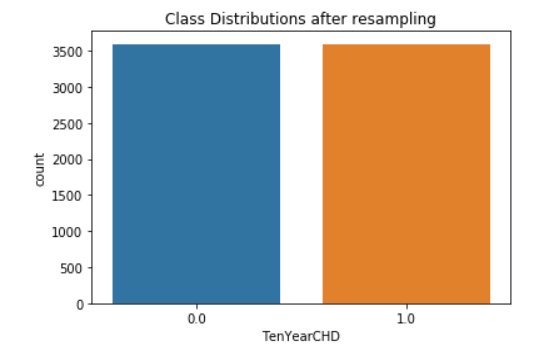
**6.3.1 Balancing the distribution**

🡪 The problem is detection that whether the user is having a coronary heart disease or not. But if there is a bias data in which the data of user not having disease is more, then it is possible that the final trained model will remain biased towards the data which is more in number. This is called overfitting. This can be seen in this diagram



It can be stated that this data is biased. Hence the library named SMOTE as mentioned above is used. SMOTE duplicates the data with the less number and makes it equal to the higher number data.

Hence the problem of overfitting can be prevented by using SMOTE. This is the new distribution after using SMOTE.



**6.3.2 Shuffling the data**

🡪 After the data was equally distributed, it was found that first half of the data contained a large number of negative labels. And the second half of the data contained a large number of positive labels. For a machine learning model to be trained accurately, it is necessary that we do not provide it with same data continuously. Hence shuffling of the data becomes very important here. After shuffling the data has become uniform and the problem of overfitting is avoided.

**6.4 Dividing the data**

🡪 Usually the Machine Learning model gives a good accuracy on the data it is being trained on. The real accuracy is measured by giving it a new data that is unseen by the model. Hence the dataset is divided in unequal 2 parts so that one of the part is used for training the model and the other part is used to test the accuracy on the unseen data. This can be done by a module named **“train-test-split”.** It is used to divide the data into 2 parts.



We have kept 80% of the data into training part and 20% of the part into testing phase. So our final data is:

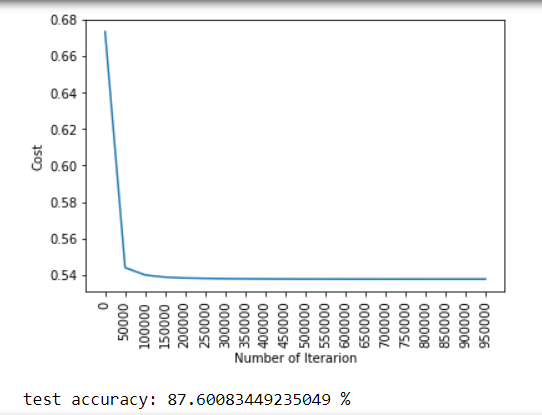
Training data : 5750 x 15

Testing data : 1438 x 15

**Chapter 7: Post Model Training**

**7.1 Getting a good test accuracy**

🡪 A machine learning model always gets a good accuracy on the data it has been trained on. But the ultimate goal of it is to get a satisfying accuracy on the data that is new it. The data that is new to the model is called the **“test data”** and the accuracy associated with it is called **“test accuracy”.** A model is only then considered as a good model when it has a good test accuracy. So in our case, we got a very good test accuracy on our model.



**7.2 Selecting Threshold value**

**7.2.1 What is a Threshold value?**

🡪 After the training of the model, we are getting the probability of the coronary heart disease. But we are not knowing that which probability can be considered as low and which can be considered as high. Hence we need a value that can be considered as the determining value i.e. under that value, the probability would be considered as **“Low”** and above that value, it would be considered as **“High”.**

That value is called the Threshold value.

**7.2.2 Decision Tree**

🡪 A decision tree is used to determine the Threshold value. A module named ROC\_Curve is used to get the **“True Positives”** and **“False Positives”** by which we will get a good threshold value.

🡪 We have here used 4 models to get a very accurate threshold value.

🡪 These are the 4 models with their respective ROC Scores.

1) Random Forest Classifier



2) Logistic Regression



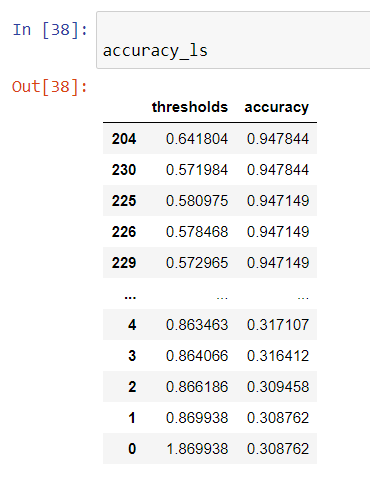
3) AdaBoost Classifier



4) KNeighbours classifier

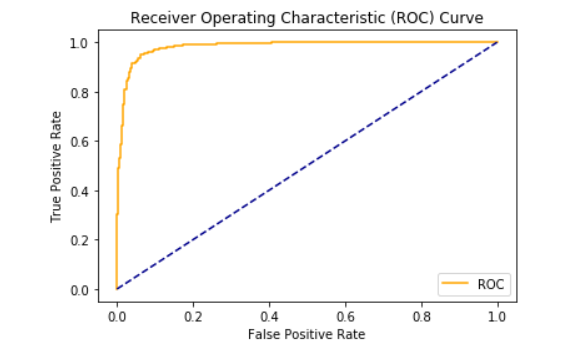


🡪 So by the help of these 4 models, we will get the threshold value.



**🡪 Hence we will select 0.57 as our threshold value.**

🡪 This is the ROC Curve:



**Chapter 8: Testing**

**8.1 Introduction**

**🡪**Software testing is the process of testing the functionality and correctness of software. Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free.

**8.2 Unit Testing**

🡪In this each module is tested individually. Criteria selected for identify unit test module is to identify module that has core functionality implementation. Module could be an individual or procedure. The following is a list of functions for unit testing that will be tested:

Get More than 90% of accuracy of Predictable Functionality work correctly.

**8.3 Integration Testing**

🡪 Integration testing integrates individual modules and tested as a group. Integration testing takes as it`s input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates and delivers as its output the integrated system for testing.

**8.4 Validation Testing**

🡪 The process of evaluating system during or at the end of the development process in to determine whether it satisfies specified requirements.

**8.5 GUI Testing**

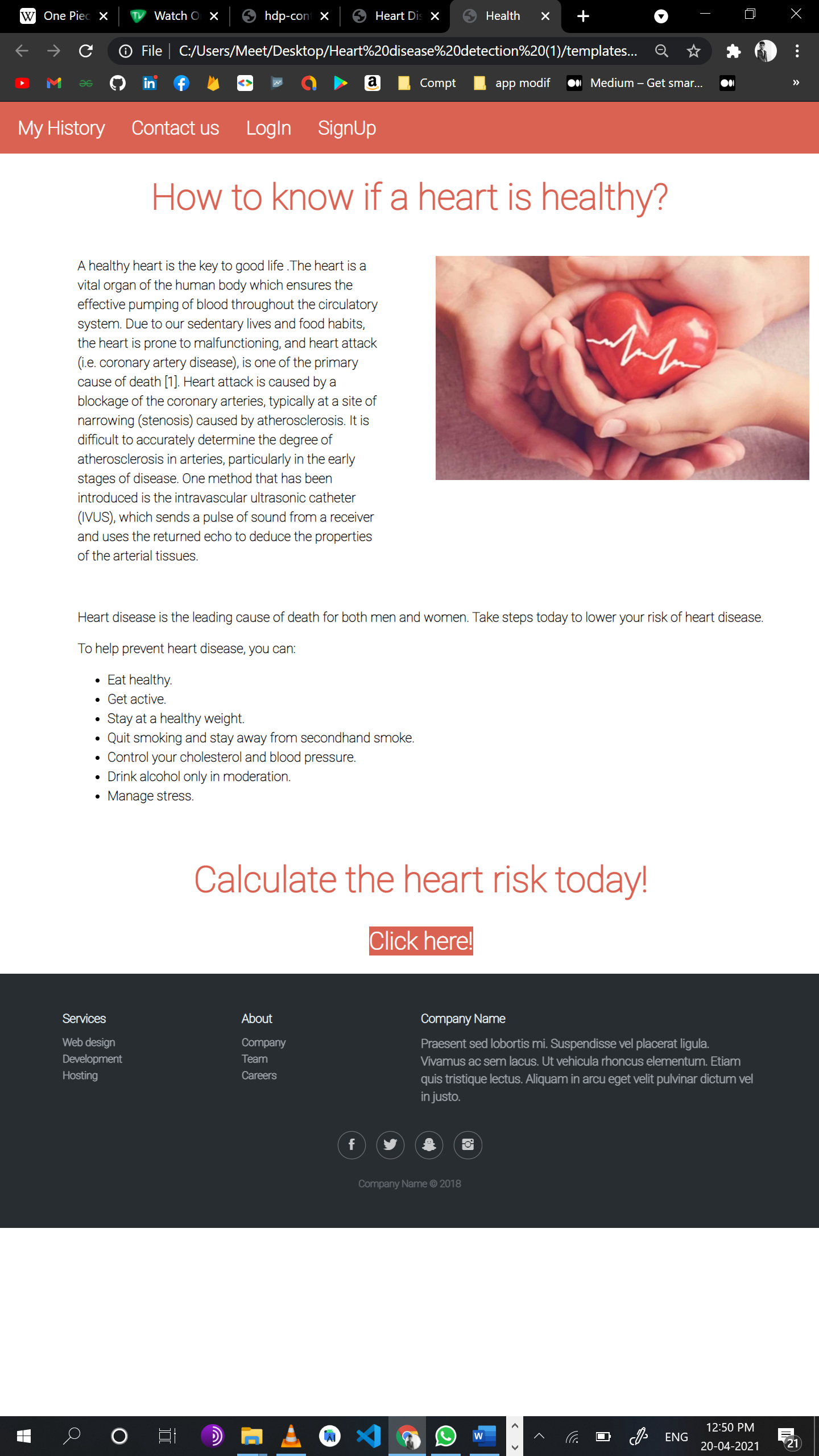
🡪 GUI Testing is the process of testing the system`s Graphical User Interface of the Application under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons and all type of bars – menu bar, dialog boxes and windows etc.

**8.6 Test Cases**

|  |  |
| --- | --- |
| **Functional Test Cases** | **Expected Output**  **Positive / Negative** |
| Verifying that the account is created and updated in the database successfully. | Positive |
| Verifying the working of the history section. | Positive |
| Verifying predicted output is more than 85%. | Positive |
| Verifying the working of the BMI calculator. | Positive |
| Verifying the working of all the alert boxes. | Positive |
| Verifying the working of the suggestions of the doctors. | Positive |

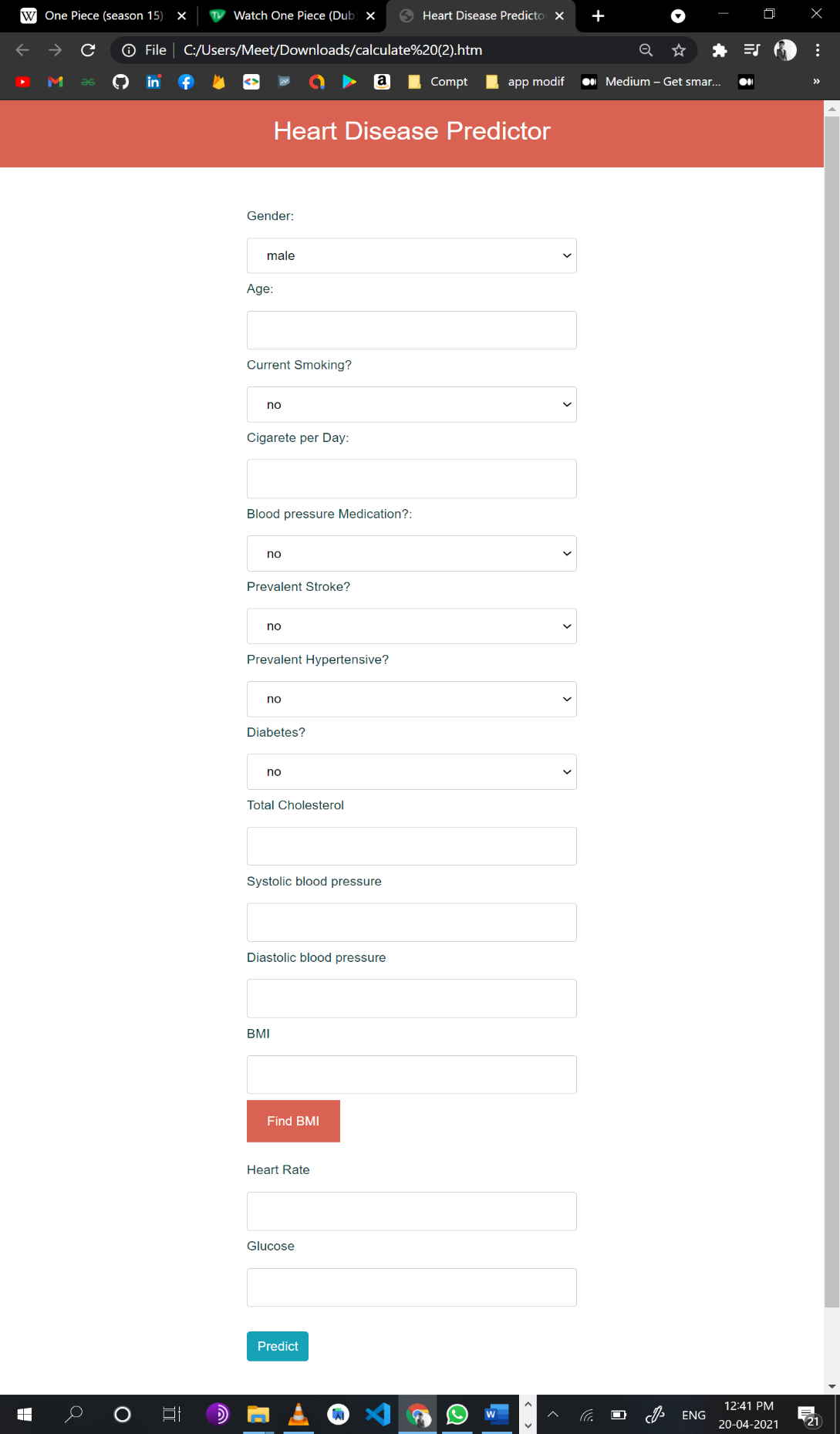
**Chapter 9: User Manual**

* **Home Page:**

****

**[Figure 9.1.1-Home page ]**

* **Prediction page:**

****

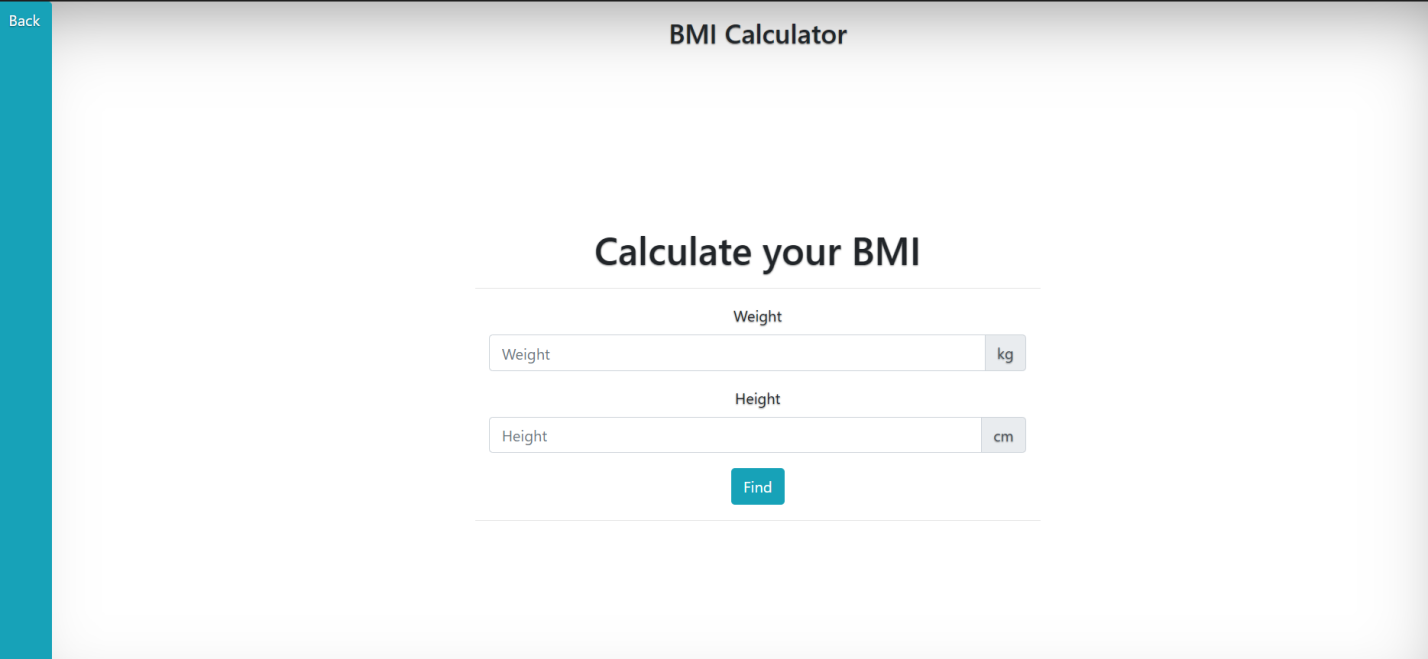
**[Figure 9.1.2-Prediction Page ]**

* **Predicting the values:**

****

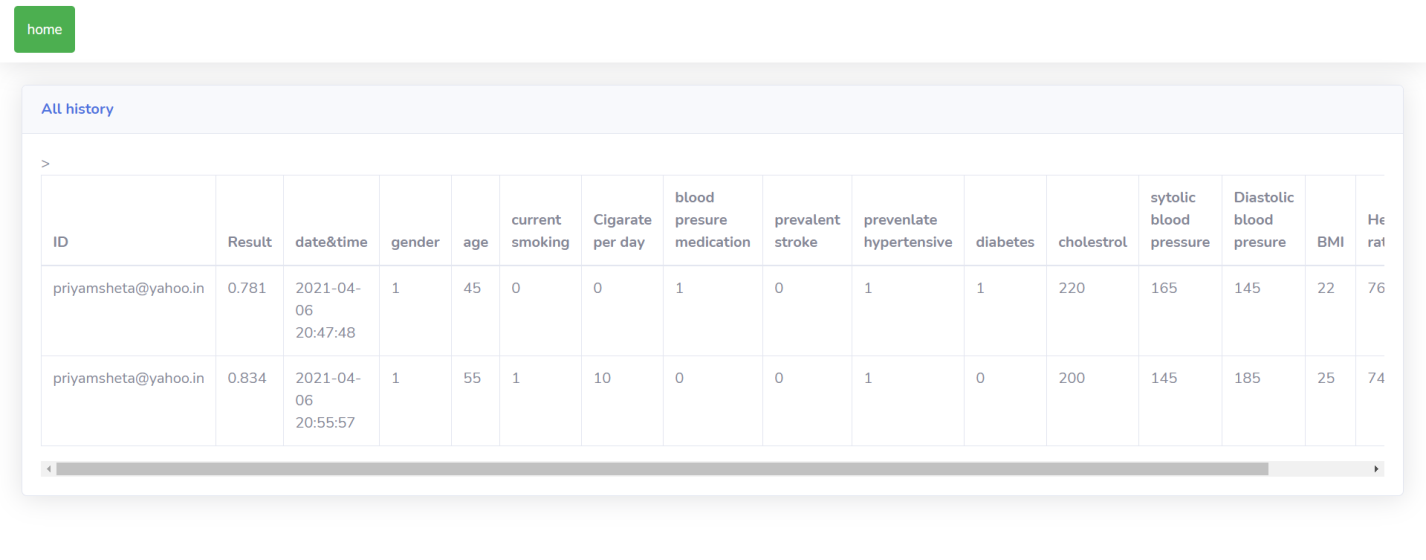
**[Figure 9.1.3- Predicting the values]**

* **BMI:**

****

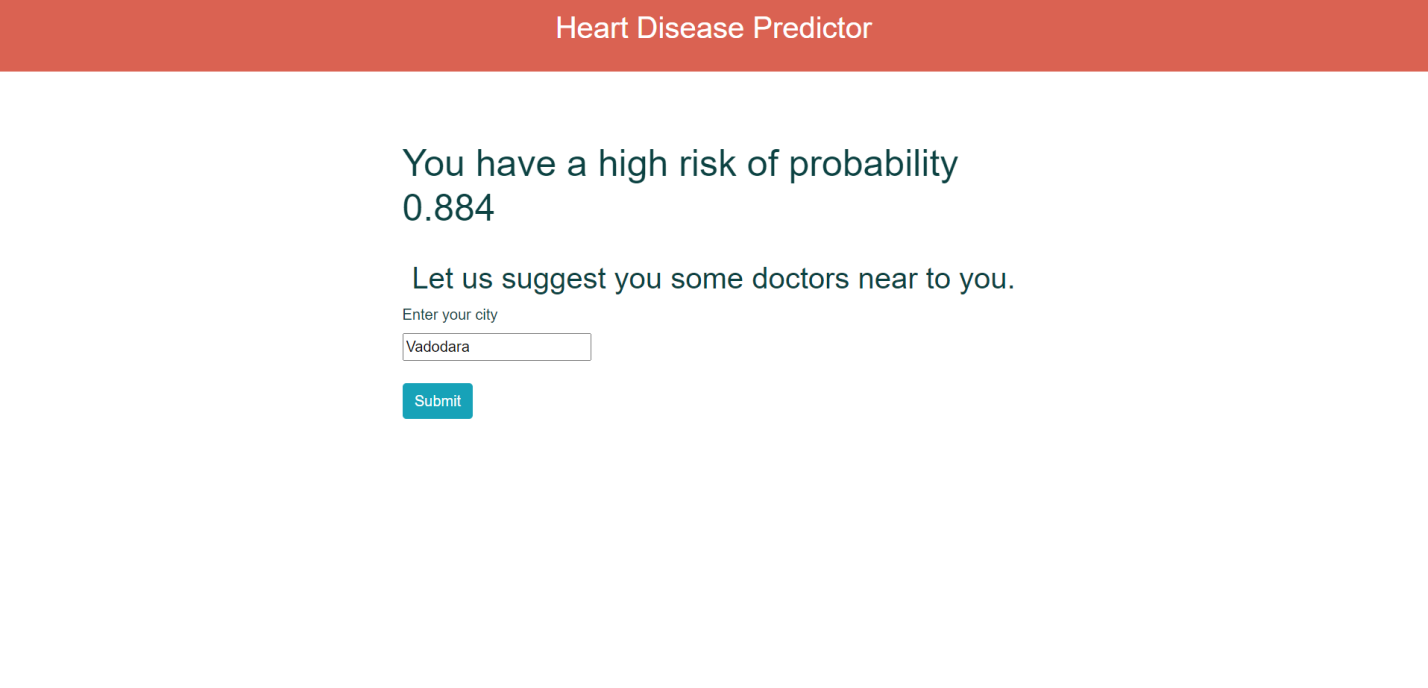
**[Figure 9.1.4- Predicting the values]**

* **History Section:**

****

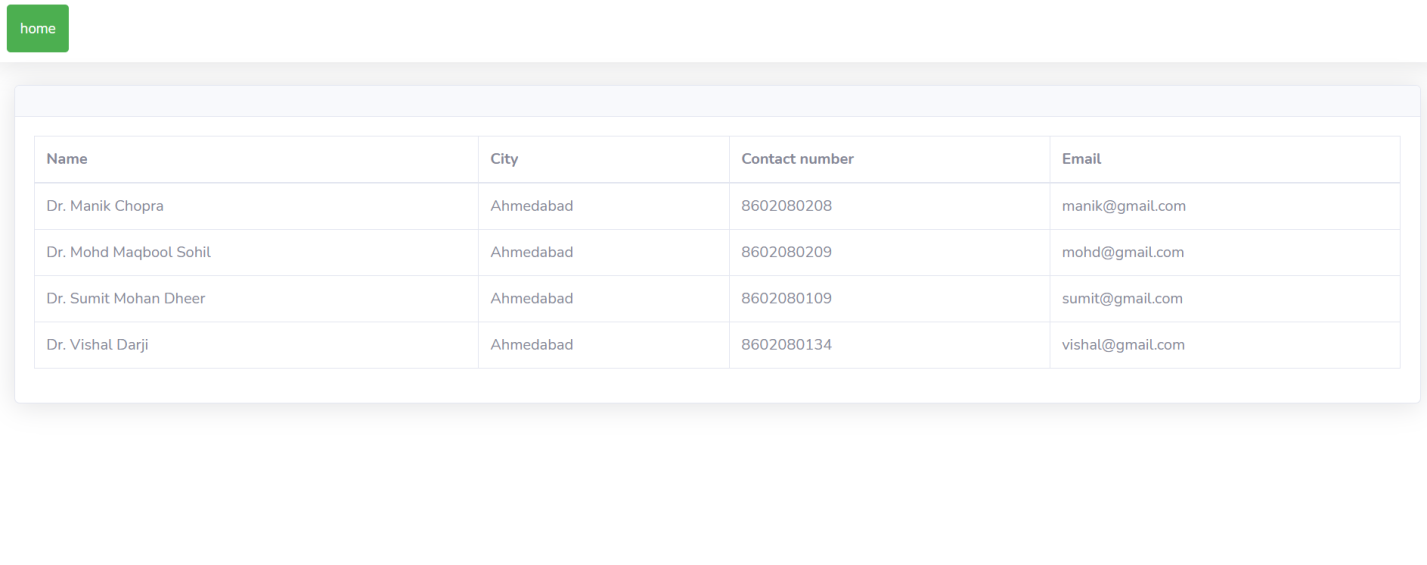
**[Figure 9.1.5-History Section]**

* **City Section:**

****

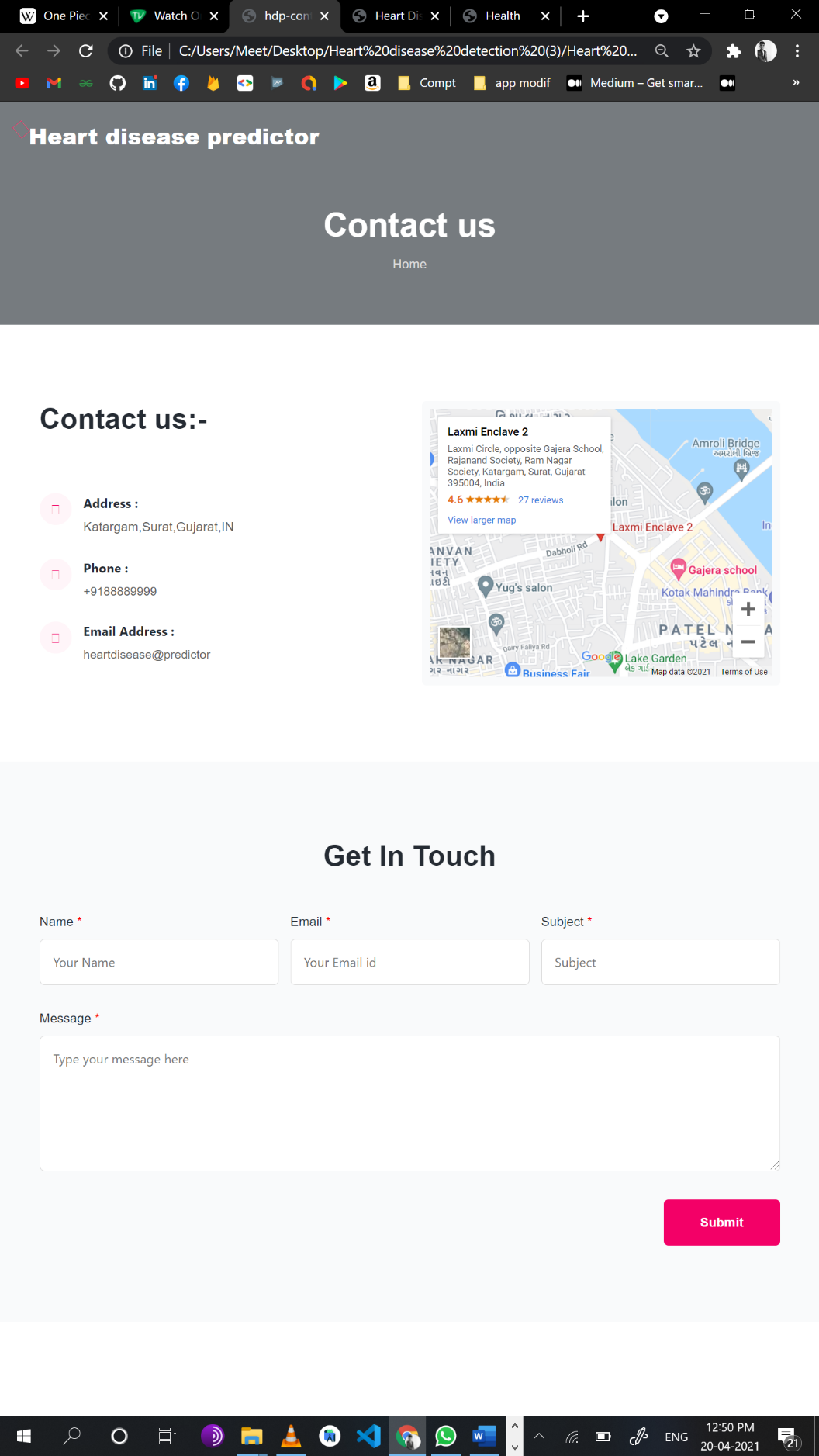
**[Figure 9.1.6-City Section ]**

* **Doctors Section:**

****

**[Figure 9.1.7-Doctor Section]**

* **Contact us Page**

****

**[Figure 9.1.8-Contact Page ]**

**Chapter 10: Limitations and Future Enhancements**

**10.1 Limitations**

🡪 We have only taken 14 features into consideration for predicting the coronary heart disease but it is possible that there are many other features that may play a role for this disease.

🡪 We are only predicting the probability of the disease but we are not suggesting the user the proper diet that must be followed with respect to their probability to avoid the disease.

**10.2 Future Enhancements**

🡪 We are thinking to involve all the features that play an important role in the disease so our users can benefit from the website.

🡪 We are also going to increase the security as no one hampers the model.

🡪 We know that more the data, more the accuracy and hence we will also work on increasing the data.

🡪 A proper meal will be also suggested by us so the user may not visit the doctor regularly.

🡪 We will also a chat section where users can directly chat with the doctors and the doctors can give suggestions to them according to their results.

**Chapter 11: Conclusion and Discussion**

**11.1 Conclusion**

🡪 It can be concluded that “Coronary heart disease prediction system” was a real world learning experience. We came to know many new concepts in the field of Machine Learning. This system can be used by all type of users and all the previous data is also stored by the system and hence the user can know whether his condition has improved. We came to know about many features that are affecting our heart and also the proportion in which they are affecting. Overall, it was a great experience learning all new technologies, working in a team and completing everything in the given time.

**11.2 Discussion**

**11.2.1 Self Analysis**

🡪 During the worktime of this project, we have analyzed many things. We observed the features used in the project and analyzed all that features. We can finally say that.

🡪 All attributes selected after the elimination process show P-values lower than 5% and thereby suggesting significant role in the Heart disease prediction.

🡪 Men seem to be more susceptible to heart disease than women. Increase in age, number of cigarettes smoked per day and systolic Blood Pressure also show increasing odds of having heart disease.

🡪 Total cholesterol shows no significant change in the odds of CHD. This could be due to the presence of good cholesterol (HDL) in the total cholesterol reading. Glucose too causes a very negligible change in odds (0.2%)

**11.2.2 Summary**

🡪 Our model gave us a very good accuracy and hence the users can use it without the fear of wrong results. It will give a very accurate probability of getting the coronary heart disease in the following 10 years. For checking the improvement of the users, there is a history section where all their previous values will get stored. Only authenticated users will be allowed to access the website. In case the user doesn’t know the BMI, there is an extra functionality that can help them find their BMI. Also, if the user is having high risk of the disease then some doctors near to them are also suggested by the website. In the present times, all type of diseases is increasing rapidly. So in such times, a website that will allow the users to know their heart conditions will be very beneficial to them.

**12. Bibliography**

1) <https://www.sciencedirect.com/science/article>

2)<https://www.researchgate.net/publication/326733163_Prediction_of_Heart_Disease_Using_Machine_Learning_Algorithms>

3) Book: Baldi, P. and Brunak, S. (2002). Bioinformatics: A Machine Learning Approach. Cambridge, MA: MIT Press.