

VISUALIZING AND PREDICTING HEART DISEASES WITH AN INTERACTIVE DASHBOARD

PROJECT REPORT

- Submitted by

Sidharth B (195002114)

Pandiyarasan D (195002078)

Alagappan S (195002301)

Kavinraj S (195002303)

TABLE OF THE CONTENT

CHAPTER	CONTENTS	PAGE NO
1	INTRODUCTION	
	1.1 PROJECT OVERVIEW	1
	1.2 PURPOSE	2
2	LITERATURE SURVEY	
	2.1 EXISTING PROBLEMS	3
	2.2 REFERENCES	3
3	IDEATION AND PROPOSED SOLUTION	
	3.1 EMPATHY MAP	5
	3.2 IDEATION AND BRAINSTORMING	5
	3.3 PROPOSED SOLUTION	6
	3.4 PROBLEM SOLUTION FIT	7
4	REQUIREMENT ANALYSIS	
	4.1 FUNCTIONAL REQUIREMENTS	8
	4.2 NON-FUNCTIONAL REQUIREMENTS	9

5	PROJECT DESIGN	
	5.1 DATA FLOW DIAGRAMS	10
	5.2 SOLUTION AND TECHNICAL ARCHITECTURE	11
	5.3 USER STORIES	11
6	PROJECT PLANNING AND SCHEDULING	
	6.1 SPRINT PLANNING AND ESTIMATION	12
	6.2 SPRINT DELIVERY SCHEDULE	13
	6.3 REPORTS FROM JIRA	13
7	CODING AND SOLUTIONING	
	7.1 FEATURE 1: PREDICTION MODEL	15
	7.2 FEATURE 2: DASHBOARD	23
	7.3 FEATURE 3: LOGIN	25
	7.1.4 FEATURE 4: SIGNUP	28
	7.2 DATABASE SCHEMA	31

8	TESTING	
	8.1 TEST CASES	31
	8.2 USER ACCEPTANCE TESTING	32
9	RESULTS	
	9.1 PERFORMANCE METRICS	32
10	ADVANTAGES AND DISADVANTAGES	33
11	CONCLUSION	33
12	FUTURE SCOPE	33
13	APPENDIX	
	APPENDIX A1– SCREENSHOTS	34

CHAPTER 1

INTRODUCTION:

1.1 : PROJECT OVERVIEW

The terms "heart disease" and "cardiovascular disease" are frequently used interchangeably. Heart disease is a general term that covers a wide range of heart related medical conditions. The irregular health state that directly affects the heart and all its components is characterized by these medical conditions. To forecast cardiac disease, this study discusses various data mining, big data, and machine learning techniques. Building an important model for the medical system to forecast heart disease or cardiovascular illness requires the use of data mining and machine learning. Our application helps the user in finding out if they have heart disease or not. They can find out by entering details such as their heart rate, cholesterol, blood pressure etc. A dashboard is also attached along with the results for better understanding where they can compare their blood pressure and similar metrics with other users. This project focuses on Random Forest Classifier. The accuracy of our project is 87% for which is better than most other systems in terms of achieving accuracy quickly.

1.2 : PURPOSE

This project's goal is to determine, depending on the patient's medical characteristics—such as gender, age, chest pain, fasting blood sugar level, etc.—whether they are likely to be diagnosed with any cardiovascular heart illnesses. The leading cause of death in the developed world is heart disease. Heart disease cases are rising quickly every day, thus it's crucial and worrisome to predict any potential illnesses in advance. This diagnosis is a challenging task that requires accuracy and efficiency. Therefore, there needs to be work done to help prevent the risks of having a heart attack or stroke. It is the main factor in adult deaths. By using a person's medical history, our initiative can identify those who are most likely to be diagnosed with a cardiac condition. It can assist in identifying disease with less medical tests and effective therapies, so that patients can be treated appropriately. It can identify anyone who is experiencing any heart disease symptoms, such as chest pain or high blood pressure. Around the world, machine learning is applied in many different fields. There is no exception in the healthcare sector. Machine learning may be crucial in determining whether locomotor disorders, heart illnesses, and other conditions are present or absent. If foreseen well in advance, such information can offer valuable insights to doctors, who can then customise their diagnosis and course of care for each patient.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

A quiet significant amount of works related to the diagnosis of Heart disease using Machine Learning algorithms have been made. An efficient heart disease prediction has been made by using various algorithms some of them include Logistic Regression, KNN, Random Forest Classifier etc. It can be seen in results that each algorithm has its strength to register the defined objectives. The model incorporating IHDPS had the ability to calculate the decision boundary using the previous and new model of machine learning and deep learning. It facilitated the important and the most basic factors/knowledge such as family history connected with any heart disease. But the accuracy that was obtained in such IHDPS model was far more less than the new upcoming model such as detecting coronary heart disease using artificial neural network and other algorithms of machine and deep learning.

2.2 REFERENCES

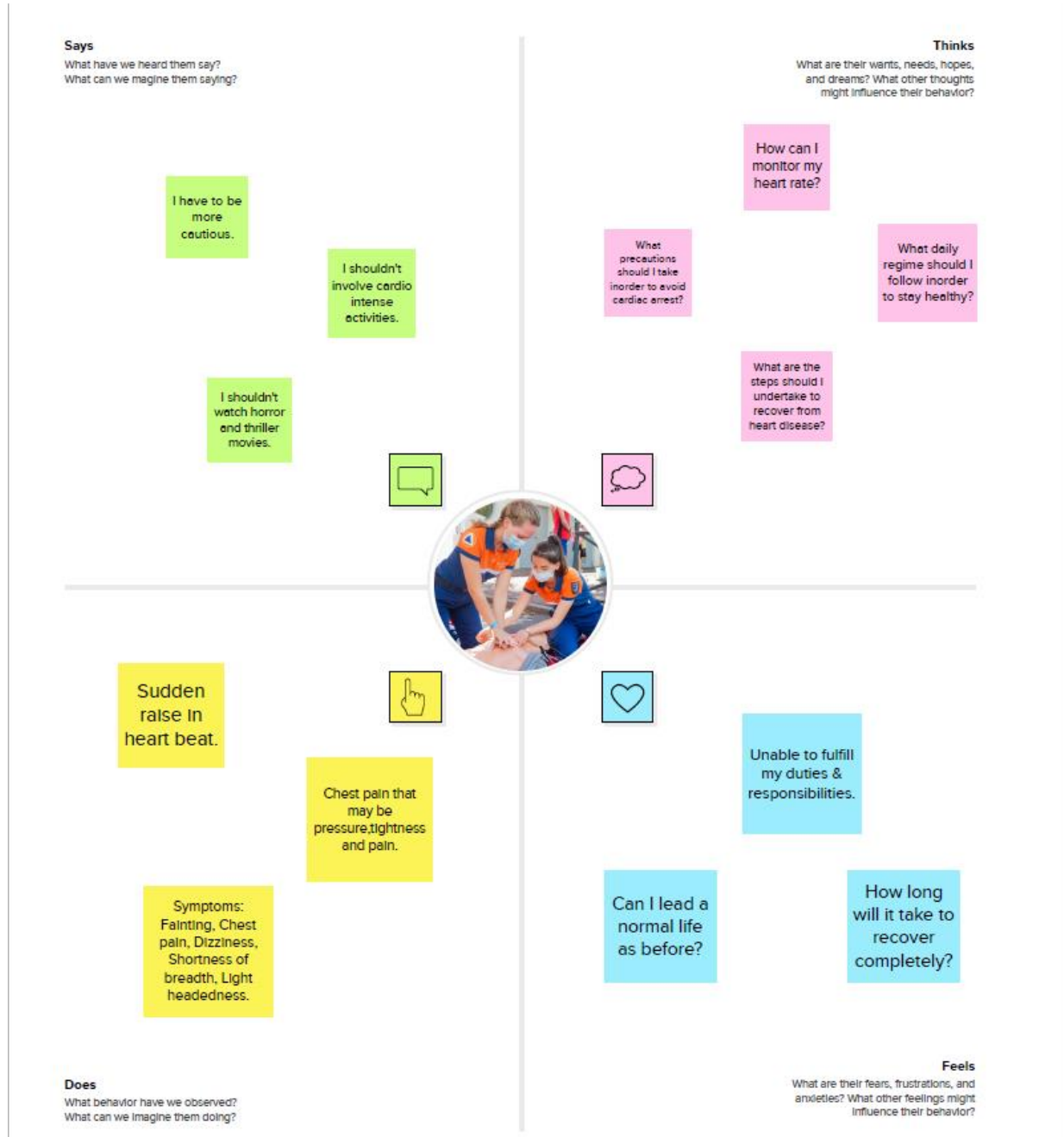
- [1] Ali, Liaqat, et al, "An optimized stacked support vector machines based expert system for the effective prediction of heart failure." IEEE Access 7 (2019): 54007-54014. www.ijcrt.org © 2020 IJCRT | Volume 8, Issue 8 August 2020 | ISSN: 2320-2882 IJCRT2008170 International Journal of Creative Research Thoughts (IJCRT) www.ijcrt.org 1606
- [2] Mohan, Senthilkumar, Chandrasegar Thirumalai, and Gautam Srivastava, "Effective heart disease prediction using hybrid machine learning techniques." IEEE Access 7 (2019): 81542-81554.

- [3] Purushottam, Kanak Saxena and Richa Sharma, "Efficient heart disease prediction system." *Procedia Computer Science* 85 (2016): 962-969.
- [4] Singh, Yeshvendra K., Nikhil Sinha, and Sanjay K. Singh, "Heart Disease Prediction System Using Random Forest", *International Conference on Advances in Computing and Data Sciences*. Springer, Singapore, 2016.
- [5] Santhana Krishnan. J, Geetha S., "Prediction of Heart Disease Using Machine Learning Algorithms", 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)
- [6] Bo Jin ,Chao Che, Zhen Liu, Shulong Zhang, Xiaomeng Yin, And Xiaopeng Wei, "Predicting the Risk of Heart Failure With EHR Sequential Data Modeling", *IEEE Access* 2018.
- [7]Aakash Chauhan , Aditya Jain , Purushottam Sharma , Vikas Deep, "Heart Disease Prediction using Evolutionary Rule Learning", "International Conference on "Computational Intelligence and Communication Technology" (CICT 2018).
- [8] Boshra Bahrami, Mirsaeid Hosseini Shirvani, "Prediction and Diagnosis of Heart Disease by Data Mining Techniques", *Journal of Multidisciplinary Engineering Science and Technology (JMEST)* ISSN: 3159-0040 Vol. 2 Issue 2, February–2015.
- [9] M.Satish, D Sridhar, "Prediction of Heart Disease in Data Mining Technique", *International Journal of Computer Trends & Technology (IJCTT)*, 2015.
- [10] Lokanath Sarangi, Mihir Narayan Mohanty, Srikanta Pattnaik, "An Intelligent Decision Support System for Cardiac Disease Detection", *IJCTA*, International Press 2015.

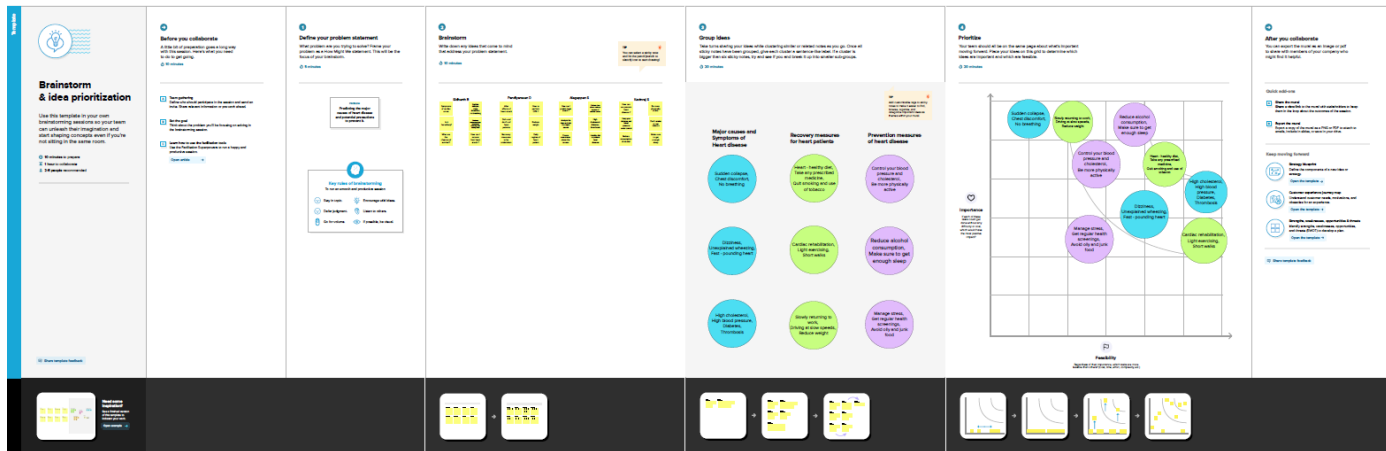
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING



3.3 PROPOSED SOLUTION

Our application helps the user in finding out if they have heart disease or not. They can find out by entering details such as their heart rate, cholesterol, blood pressure etc. A dashboard is also attached along with the results for better understanding where they can compare their blood pressure and similar metrics with other users. Our application has one of the smoothest user interfaces on the internet making it easy for the user to find their needs quickly and efficiently. And the tool utilizes the best machine learning algorithms for better prediction. There are separate sections for viewing treatment options, warning signs of cardiac arrest, risk factors and causes of various types of heart diseases.

Proposed Solution Template:

S. No	Parameter	Description
1.	Problem statement (Problem to be solved)	<p>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.</p> <p>Globally, nearly 18.6 million people died of heart diseases in 2019, the latest year for which worldwide statistics are calculated. That reflects a 17.1% increase over the past decade. There were more than 523.2 million cases of heart disease in 2019, an increase of 26.6% compared with 2010</p> <p>To predict which patients are most likely to suffer from a heart disease in the near future using the features given so that they can take educated, planned steps for the next phase of treatment.</p>
2.	Idea / Solution description	<p>In this project, we plan to build an interactive dashboard for understanding and visualising heart diseases using data analytics platform , in which we classify a person as prone to heart disease or not by considering various factors like age, sex blood pressure, cholesterol , maximum heart rate achieved etc.</p> <p>Further using machine learning algorithms like K Neighbors Classifier, in which we predict whether a person is prone to having a heart disease or not.</p>
3.	Novelty / Uniqueness	<p>Identifying heart diseases till now is in the phase of only research papers and no further working models to predict with high accuracy.</p> <p>So a model with better accuracy is aimed since false predictions results in unwanted fear and treatments.</p> <p>Finding the heart diseases in the early stages by predicting all possible outcomes in such a way by visualising the data obtained in the forms of graphs and charts to educate the user easily and effectively</p>

4.	Social Impact / Customer Satisfaction	<p>Provide users a reliable prediction to clear the doubt if they actually are suffering from any heart related issues or not</p> <p>This also helps reduce time of doctors and expenses such as transport, consultation fees, and other accompanying medical test charges for patients</p>
5.	Business Model (Revenue Model)	<p>This project would be a not-for-profit initiative with the sole purpose of it being a helping aid for people in need</p> <p>Heart specialists and hospitals can use the interactive dashboard to keep track of patient health and receive notifications regarding the same</p>
6.	Scalability	<p>Initially this model is focused on a small number of users in the development phase once the user traffic increases larger number of samples from users can also act as the input to the training model and as the number of users increases the platform can be moved to cloud for higher storage and performance and along with google ads we can add user subscription for a year/month and also doctors can pay money for suggesting their healthcare at the top of the list under contact details where the profiles of the healthcare providers will be thoroughly authenticated</p>

3.4 PROBLEM SOLUTION FIT

Problem-Solution Fit simply means that we have found a problem with our customer and that the solution we have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why. The purpose is to solve complex problems in a way that fits the state of your customers and succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS <p>Doctors, Medical professionals and patients or people who want to know Whether they are prone to heart disease all come under individual users</p>	6. CUSTOMER CONSTRAINTS CC <p>Doctor's availability, Patients may have a constrained expenditure in travelling, CT scan and Fees. Patient may not have the time to the Doctors</p>	5. AVAILABLE SOLUTIONS AS <p>In the available solution: A Health care provider will examine you and ask about your personal and family medical history. Many different tests are used to diagnose heart disease. Besides blood tests and a chest X-ray, tests to diagnose heart disease can include:</p> <ul style="list-style-type: none"> • Electrocardiogram (ECG OR EKG) • Holter Monitoring • Echocardiogram • Cardiac catheterization • CT or MRI Scan 	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P <p>The goal of our heart disease prediction project is to determine if a patient should be diagnosed with heart disease or not. Patients susceptible to heart disease would face the difficulty of visiting the hospital each and every time</p>	9. PROBLEM ROOT CAUSE RC <p>Users are lazy or maybe due to other commitments are not able to meet the doctors at right time. This system will be more effective in that manner</p>	7. BEHAVIOUR BE <p>Given necessary data, users expect accurate analysis, visualizations and prediction whether there is a possibility of occurrence of heart disease or not.</p>	
Focus on J&P, tip into BE, understand RC	3. TRIGGERS TR <p>The system is user friendly and the user can find out the problem by just feeding in the necessary data instead of waiting for a doctor's appointment.</p>	10. YOUR SOLUTION SL <p>A system that provides visualization and prediction whether a person has heart disease. The user feeds in the necessary data required and system outputs the possibility of the person having a heart disease.</p>	8. CHANNELS of BEHAVIOUR CH <p>8.1 OFFLINE Easily can communicate with other patients and doctors and know about what happening in the cause of disease easily. For medical institutions, a group of professionals are involved and through word-of-mouth, other Medical Institutions and individuals will become aware of this application</p> <p>8.2 ONLINE The system can be accessed through modern-day browser like Chrome, Safari, Firefox, etc.</p>	Focus on J&P, tip into BE, understand RC
	4. EMOTIONS: BEFORE / AFTER EM <p>BEFORE: User doesn't have more technology right Now. Also, patient want to visit doctors frequently and thus would be annoyed and uncertain whether there is heart disease or not.</p> <p>AFTER: Users feel much more comfortable with system due to it ease of access and user-friendly features and can attain instant result</p>			

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

- Users have to register.
- Function to view the homepage by the user.
- Function to display information related to heart diseases on the website.
- Function to provide textboxes to enter medical results.
- Function to predict heart disease using ML model.
- Function to display visualizations of the final results.
- Function to provide dashboard to user.

4.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional Requirements:

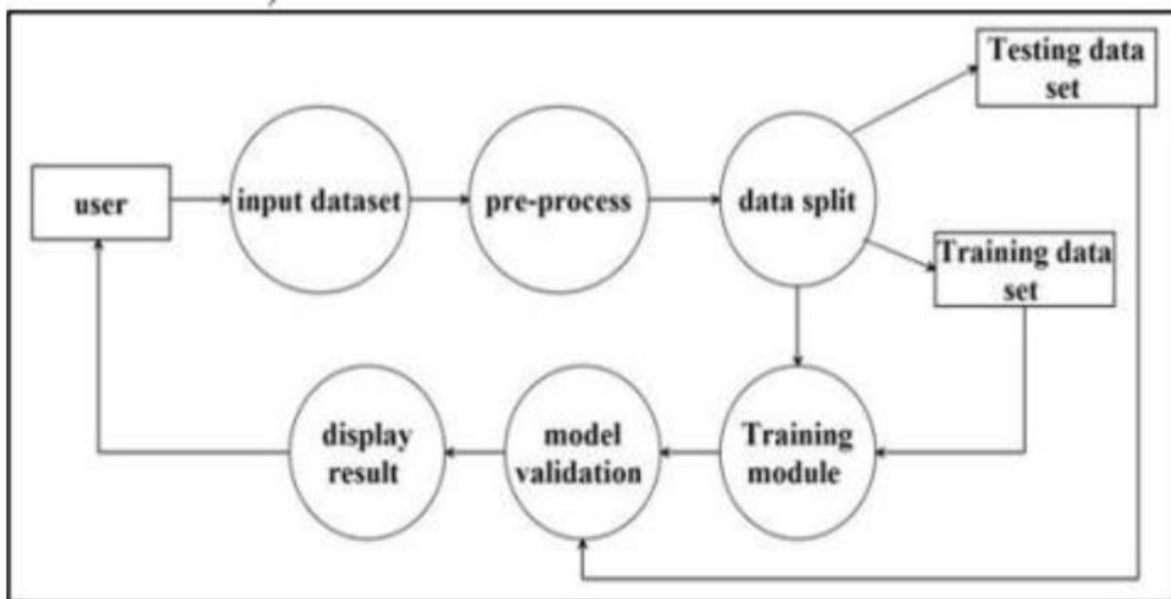
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Reliability	The structure must be reliable and strong in giving the functionalities. The progressions made by the Programmer must be Project pioneer and in addition the Test designer
NFR-2	Performance	The structure should not capitulate when various users would use everything the while. It should allow brisk accessibility to each and every piece of its users.
NFR-3	Maintability	The system watching and upkeep should be fundamental and focus in its approach. There should not be an excess of occupations running on diverse machines to avoid lapsing.

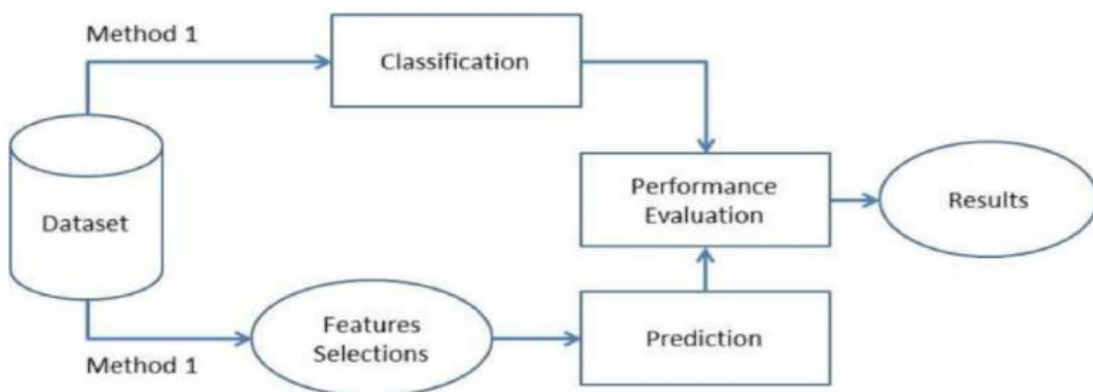
CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Patient)	Registration	USN-1	As a user, I can register by entering my email, phone number, Date of birth, password, and confirm password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive the confirmation message in my email once I have registered or OTP will be sent.	I receive confirmation email & click confirm. or by entering the OTP received	High	Sprint-1
		USN-3	As a user, I can register through Gmail		Medium	Sprint-1
	Login	USN-4	As a user, I can log in by entering email & password		High	Sprint-1
	Forgot Password	USN-5	As a user, if I forgot my password, by clicking forgot password an OTP is sent to	By entering the OTP sent via phone number or email.	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			my number or email,			
	Data collection	USN-6	As a user, I can upload the input data set to diagnose.		High	Sprint-1
Admin	Login	USN-1	As an admin, I can login by using email id and password.		High	Sprint-1
	Data collection	USN-2	As an admin, I can upload the data set to train the machine.		High	Sprint-1

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & 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 for the application by entering my email, password, and confirming my password.	5	High	Alagappan S, Kavindra S
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	3	High	Sidharth B, Pandiyarasan D
Sprint-1		USN-3	As a user, I can register for the application through GMAIL	2	Low	Pandiyarasan D, Alagappan S
Sprint-1		USN-4	As a user, I can register for the application through Google	2	Medium	Sidharth B
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	3	High	Sidharth B, Pandiyarasan D, Alagappan S
Sprint-2	Dashboard	USN-6	As a User, I can view my complete medical analysis & accuracy and prediction of heart disease in a dashboard	5	High	Sidharth B, Pandiyarasan D, Alagappan S, Kavindra S
Sprint-2	User entry	USN-7	As a User, I can enter my personal details for analysis	3	High	Pandiyarasan D
Sprint-2		USN-8	As a User, I can enter my medical records & symptoms	3	High	Sidharth B, Alagappan S,
Sprint-3	User profile	USN-9	As a user, I can update the health details of users.	5	High	Sidharth B
Sprint-3	Helpdesk	USN-10	As a user, I can post my queries & view the frequently asked question (FAQ)	5	High	Alagappan S
Sprint-3		USN-11	As an admin, I can view the user queries	3	High	Sidharth B, Kavindra S
Sprint-4	Rating	USN-12	As a user, I can rate the app and give feedback	2	Low	Kavindra S
Sprint-4	User profile	USN-13	As an admin, I can update the health details of users.	5	High	Sidharth B
Sprint-4		USN-14	As an admin, I can add or delete users.	3	High	Alagappan S
Sprint-4		USN-15	As an admin, I can manage the user details.	3	High	Pandiyarasan D

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	13	6 Days	24 Oct 2022	29 Oct 2022	13	29 Oct 2022
Sprint-2	13	6 Days	31 Oct 2022	05 Nov 2022	13	05 Nov 2022
Sprint-3	13	6 Days	07 Nov 2022	12 Nov 2022	13	12 Nov 2022
Sprint-4	13	6 Days	14 Nov 2022	19 Nov 2022	13	19 Nov 2022

6.3 REPORTS FROM JIRA



Projects / Visualizing and Predicting Heart Diseases

Backlog

S
PD
A
+

Epic ▾

[Insights](#)

VPHD Sprint 3 30 Oct – 5 Nov (3 issues)

0 0 0
Complete sprint
⋮

■ VPHD-9 Website - Connect frontend & backend

IN PROGRESS ▾
PD

■ VPHD-10 Dashboard - Visualizations for Heart Disease Prediction

DONE ▾
A

■ VPHD-11 Interactive Dashboard for Heart Disease Prediction

TO DO ▾
S

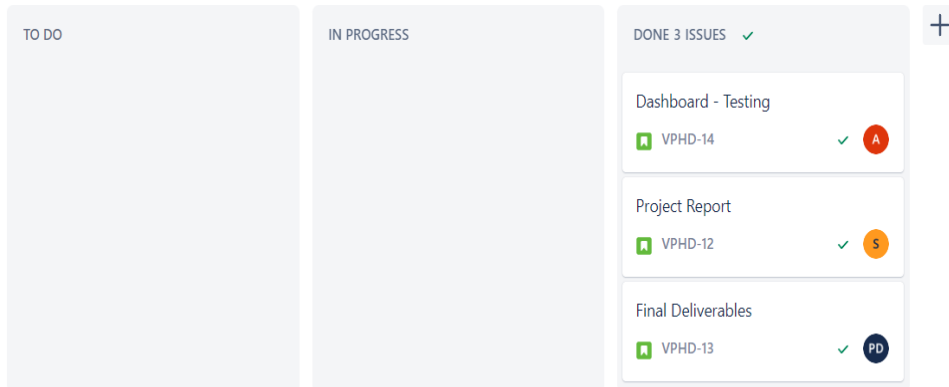
+ Create issue

VPHD Sprint 4

⚡ ☆ ⌚ 0 days remaining [Complete sprint](#) ⋮

S
A
PD
+

GROUP BY [None](#) [Insights](#)



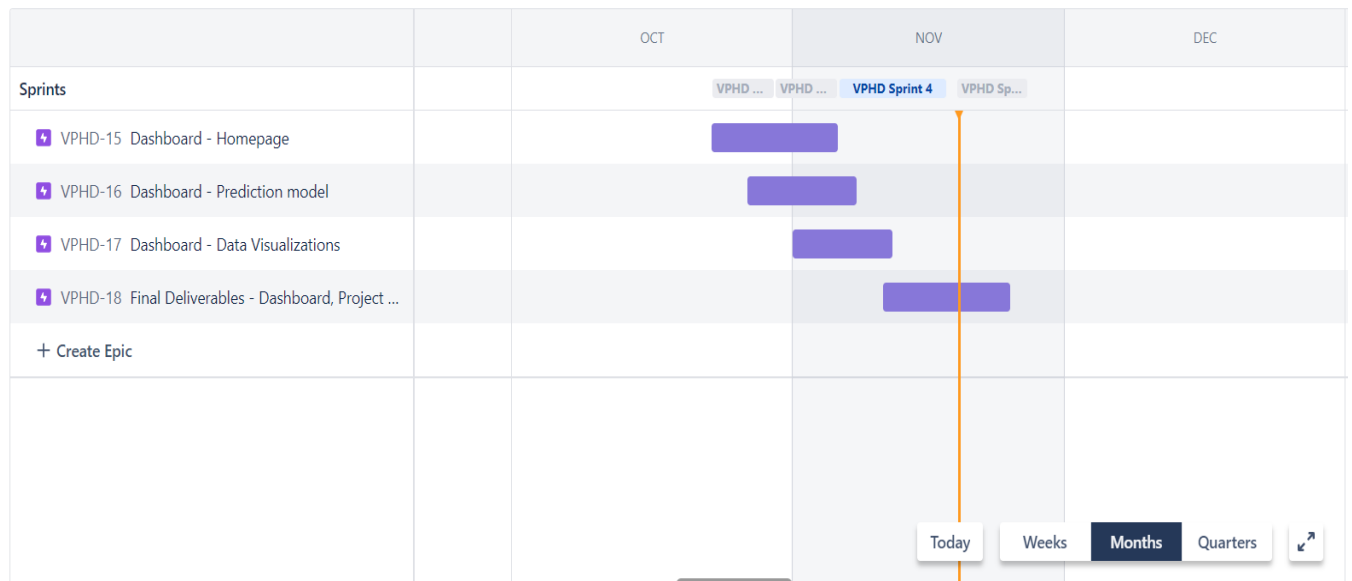
Roadmap

📢 Give feedback [Share](#) [Export](#) ⋮

S
+

Status category [▼](#) Epic [▼](#)

[View settings](#)



CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

Prediction Model: When applied to a nonlinear data set, the random forest technique performs better than the decision tree. The collection of decision trees known as a random forest was produced by several root nodes. The random forest algorithm can achieve more accuracy quickly and produce expected results.

Algorithm:

Step 1: Input the required details

Step 2: The model processes the input with the help of random forest algorithm

Step 3: The results are displayed

Code:

```
import axios from 'axios';

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

import './FormPageCommons.css';

function PredictionPage() {

  const [age, setAge] = useState("");

  const [sex, setSex] = useState("");

  const [chestPainType, setChestPainType] = useState("");

  const [bp, setBp] = useState("");

  const [cholesterol, setCholesterol] = useState("");
```

```

const [fbs, setFbs] = useState("");

wconst [ekg, setEkg] = useState("");

const [maxHr, setMaxHr] = useState("");

const [angina, setAngina] = useState("");

const [stDepression, setStDepression] = useState("");

const [slopeOfSt, setSlopeOfSt] = useState("");

const [fluro, setFluro] = useState("");

const [thallium, setThallium] = useState("");

const navigate = useNavigate();

function predictResult(event) {

  event.preventDefault();

  if (age && sex && chestPainType && bp && cholesterol && fbs && ekg && maxHr &&
  angina && stDepression && slopeOfSt && fluro && thallium) {

    const      predictionParams      =      { age,sex,chest_pain_type:
chestPainType,bp,cholesterol,fbs_over_120:      fbs,      ekg_results:      ekg,max_hr:
maxHr,exercise_angina:  angina,  st_depression:  stDepression,slope_of_st:  slopeOfSt,
number_of_vessels_fluro: fluro, thallium,};

    const headers = {

      Authorization: `Bearer ${sessionStorage.getItem('token')}` ,

    };

    axios.post('http://127.0.0.1:8000/predict', predictionParams, { headers })

    .then((response) => {

      predictionParams.prediction = response.data.prediction;

      [predictionParams.date, predictionParams.time] = new Date().toLocaleString().split(' ');

      sessionStorage.setItem('prediction', JSON.stringify(predictionParams));

      navigate('/results');

```

```

    }) // eslint-disable-line no-console

    .catch((ex) => console.log(ex)); // eslint-disable-line no-console
  }
}

return (

  <div className="main-container">

    <form className="main-form" onSubmit={predictResult}>

      <div className="title">Enter your details</div>

      <div className="form-input">

        <p>Age</p>

        <input type="number" name="Age" id="text-input" value={age} onChange={(e) =>
setAge(e.target.value)} />

      </div>

      <div className="form-input">

        <p>Sex</p>

        <select

          name="sex"

          id="sel-input"

          value={sex}

          onChange={(e) => setSex(e.target.value)}

        >

          <option value="-1">--Select Value--</option>

          <option value="1">Male</option>

          <option value="0">Female</option>

        </select>

```

```

</div>

<div className="form-input">

  <p>Chest Pain Type</p>

  <select

    name="chest-pain"

    id="sel-input"

    value={chestPainType}

    onChange={(e) => setChestPainType(e.target.value)}

    >

      <option value="-1">--Select Value--</option>

      <option value="1">Typical Angina</option>

      <option value="2">Atypical Angina</option>

      <option value="3">Non Anginal Pain</option>

      <option value="4">Asymptomatic Pain</option>

    </select>

  </div>

  <div className="form-input">

    <p>Blood Pressure</p>

    <input type="number" className="form__field" name="bp" id="text-input"
value={bp} onChange={(e) => setBp(e.target.value)} />

  </div>

  <div className="form-input">

    <p>Cholesterol</p>

    <input type="number" name="cholesterol" id="text-input" value={cholesterol}
onChange={(e) => setCholesterol(e.target.value)} />

```

```

</div>

<div className="form-input">

  <p>Fasting Blood Sugar Over 120</p>

  <select

    name="fbs"

    id="sel-input"

    value={fbs}

    onChange={(e) => setFbs(e.target.value)}

  >

    <option value="-1">--Select Value--</option>

    <option value="1">Yes</option>

    <option value="0">No</option>

  </select>

</div>

<div className="form-input">

  <p>EKG Results</p>

  <select

    name="ekg"

    id="sel-input"

    value={ekg}

    onChange={(e) => setEkg(e.target.value)}

  >

    <option value="-1">--Select Value--</option>

    <option value="0">Normal</option>

```


`<option value="1">Having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of greater than 0.05 mV)</option>`

`<option value="2">Showing probable or definite left ventricular hypertrophy by Estes`'` criteria</option>`

`</select>`

`</div>`

`<div className="form-input">`

`<p>Maximum Heart Rate</p>`

`<input type="number" name="mhr" id="text-input" value={maxHr} onChange={(e) => setMaxHr(e.target.value)} />`

`</div>`

`<div className="form-input">`

`<p>Exercise Induced Angina</p>`

`<select`

`name="angina"`

`id="sel-input"`

`value={angina}`

`onChange={(e) => setAngina(e.target.value)}`

`>`

`<option value="-1">--SelectValue--</option>`

`<option value="1">Yes</option>`

`<option value="0">No</option>`

`</select>`

`</div>`

`<div className="form-input">`

```

    <p>ST Depression</p>

    <input    type="number"    name="stdep"    id="text-input"    value={stDepression}
onChange={ (e) => setStDepression(e.target.value)} />

</div>

<div className="form-input">

    <p>Slope of ST</p>

    <select

        name="slope-st"

        id="sel-input"

        value={slopeOfSt}

        onChange={ (e) => setSlopeOfSt(e.target.value)}

    >

        <option value="-1">--Select Value--</option>

        <option value="1">Upsloping</option>

        <option value="2">Flat</option>

        <option value="3">Downsloping</option>

    </select>

</div>

<div className="form-input">

    <p>Number of major vessels colored by Flouroscopy</p>

    <select

        name="fluro"

        id="sel-input"

        value={fluro}

        onChange={ (e) => setFluro(e.target.value)}

```

```

>

<option value="-1">--Select Value--</option>

<option value="0">0</option>

<option value="1">1</option>

<option value="2">2</option>

<option value="3">3</option>

</select>

</div>

<div className="form-input">

  <p>Thallium</p>

  <select

    name="thallium"

    id="sel-input"

    value={ thallium }

    onChange={ (e) => setThallium(e.target.value)}

  >

    <option value="-1">--Select Value--</option>

    <option value="3">Normal</option>

    <option value="6">Fixed Defect</option>

    <option value="7">Reversible Defect</option>

  </select>

</div>

  <button type="text" className="submit">Submit</button>

</form>

```

```

    </div>

  );
}

export default PredictionPage;

```

7.2 FEATURE 2

Dashboard: Our application helps the user in finding out if they have heart disease or not. They can find out by entering details such as their heart rate, cholesterol, blood pressure etc. A dashboard is also attached along with the results for better understanding where they can compare their blood pressure and similar metrics with other users.

Code:

```

import Sidebar from "../../components/sidebar/Sidebar"

import "../home.scss"

import "../Cards.css"

const DashboardHome = () => {

  return (

    <div className="home">

      <Sidebar />

      <div className="homeContainer">

        <h1>Welcome to your Dashboard!</h1>

        <br />

        <br />

        <br />

```

```

<h3>Check out your email. Your results have been sent there.</h3>

<br />

<br />

<h3>Here you can check out different kind of visualizations to get a general idea about
the factors increasing causes of getting a cardiac arrest.</h3>

<br />

<div class="row">

  <div class="col-md-4 col-xl-3">

    <div class="card bg-c-blue order-card">

      <div class="card-block">

        <h2      class="text-right"><i      class="fa      fa-cart-plus      f-
left"></i><span>87%</span></h2>

        <p class="m-b-0">Accuracy</p>

      </div>

    </div>

  </div>

  <div class="col-md-4 col-xl-3">

    <div class="card bg-c-green order-card">

      <div class="card-block">

        <h2 class="text-right"><i class="fa fa-cart-plus f-left"></i><span>5</span></h2>

        <p class="m-b-0">Visualization Types</p>

      </div>

    </div>

  </div>

  <div class="col-md-4 col-xl-3">

```

```

<div class="card bg-c-yellow order-card">

  <div class="card-block">

    <h2 class="text-right"><i class="fa fa-cart-plus f-left"></i><span>Random Forest
Classifier</span></h2>

    <p class="m-b-0">ML Model</p>

  </div>

</div>

</div>

</div>

</div>

</div>

</div>

)
}

export default DashboardHome

```

7.3 FEATURE 3

Login

Algorithm :

1. Input the credentials (email and password).
2. If already logged in user is taken to home page
3. Else, check for validity of credentials
4. If wrong credentials entered , notification is displayed to user and user stays in login page.
5. On correct credentials, user is taken to home page.

Code:

```
import axios from 'axios';

import React, { useState } from 'react';

import './FormPageCommons.css';

import { useNavigate } from 'react-router-dom';

const Login = () => {

  const [email, setEmail] = useState("");

  const [password, setPassword] = useState("");

  const navigate = useNavigate();

  function loginUser(event) {

    event.preventDefault();

    const userDetails = {

      email, password,

    };

    // console.log(userDetails); // eslint-disable-line no-console

    if (userDetails && userDetails.email && userDetails.password) {

      axios.post('http://127.0.0.1:8000/login', userDetails)

        .then((response) => {

          sessionStorage.setItem('token', response.data.token);

          navigate('/predict');

        })

        .catch((ex) => {

          // console.log(ex); // eslint-disable-line no-console

          // const error = JSON.parse(ex);
```

```

    if (ex.response && ex.response.status && ex.response.status === 404) {

        alert('User not found'); // eslint-disable-line no-alert

    } else {

        console.log(ex); // eslint-disable-line no-console

    }

});

} else {

    alert('Please enter valid credentials'); // eslint-disable-line no-alert

}

}

return (

    <div className="main-container">

        <form className="main-form" onSubmit={loginUser}>

            <div className="form-input">

                <p>e-mail</p>

                <input type="email" name="email" id="email" value={email} onChange={(e) =>
setEmail(e.target.value)} />

            </div>

            <div className="form-input">

                <p>Password</p>

                <input type="password" name="password" id="password" value={password}
onChange={(e) => setPassword(e.target.value)} />

            </div>

            <div className="button-container">

                <input type="submit" />

```



```

        </div>

    </form>

</div>

);

};

export default Login;

```

7.4 FEATURE 4

Signup

Algorithm:

1. Input the signup form fields (name , email , password , re-enter password).
2. All credentials are validated.
3. Website checks whether the given email exists in the database.
4. If already registered, notification is displayed. Or else, the user is taken to the login page.

Code:

```

import axios from 'axios';

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

import { Link } from 'react-router-dom';

const Register = () => {

    const [name, setName] = useState("");

```

```

const [email, setEmail] = useState("");

const [password, setPassword] = useState("");

const [confPassword, setConfPassword] = useState("");

const navigate = useNavigate();

function registerUser(event) {

  event.preventDefault();

  const userDetails = {

    fullName: name, email, password,

  };

  if (userDetails && userDetails.fullName && userDetails.password && userDetails.email)
  {

    // console.log('Hi'); // eslint-disable-line no-console

    axios.post('http://127.0.0.1:8000/register', userDetails)

      .then(() => {

        // console.log(response.json); // eslint-disable-line no-console

        navigate('/login');

      })

      .catch((ex) => console.log(ex)); // eslint-disable-line no-console

  }

}

return (

  <div className="main-container">

    <form className="login-form" onSubmit={registerUser}>

      <div className="form-input">

        <p>Name</p>

```

```

    <input type="text" name="text" id="text" value={name} onChange={(e) =>
setName(e.target.value)} />

</div>

<div className="form-input">

    <p>e-mail</p>

    <input type="email" name="email" id="email" value={email} onChange={(e) =>
setEmail(e.target.value)} />

</div>

<div className="form-input">

    <p>Password</p>

    <input type="password" name="password" id="password" value={password}
onChange={(e) => setPassword(e.target.value)} />

</div>

<div className="form-input">

    <p>Confirm Password</p>

    <input type="password" name="password" id="conf-password" value={confPassword}
onChange={(e) => setConfPassword(e.target.value)} />

</div>

<div className="button-container">

    <input type="submit" />

    <p>Already have an account?</p>

    <Link to="/login">

        <button type="button">Login</button>

    </Link>

</div>

```

```
    </form>

  </div>

);

};

export default Register;
```

7.5 DATABASE SCHEMA

NoSQL databases like MongoDB offer high performance, high availability, and easy scalability. MongoDB is a document-oriented database which stores data in JSON-like documents with dynamic schema. It means you can store your records without worrying about the data structure such as the number of fields or types of fields to store values. MongoDB documents are similar to JSON objects. Details like name, e-mail, password of the registered user are stored so that when the user tries to login, authentication takes place and the user is logged in.

CHAPTER 8

TESTING

8.1 TEST CASES

Testcase 1: Logging in with registered login details.

Testcase 2: Logging in with invalid login details.

Testcase 3: Registering with existing user's details.

Testcase 4: Entering wrong values while filling medical related details.

Testcase 5: Producing visualisations for given input.

8.2 USER ACCEPTANCE TESTING

Test case ID	Feature Type	Component	Test Scenario	Pre-Requirement	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	Stable internet connection, Compatible browser, Login credentials	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Signup popup displayed or not	https://shopnizer.com/	Login/Signup popup should display	Working as expected	Pass	Login page displayed successfully	N	1
LoginPage_TC_002	UI	Home Page	Verify the UI elements in Login/Signup popup	Proper code for UI elements, Elements position, Buttons and Textbox response	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Signup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link	https://shopnizer.com/	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.Last password? Recovery password link	Working as expected	Fail	Elements are displayed successfully but recovery password button is not present	N	2
LoginPage_TC_003	Functional	Home page	Verify user is able to log into application with Valid credentials	User credentials, Database with credentials of existing users	1.Enter URL(https://shopnizer.com/) and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123	User should navigate to user account homepage	Working as expected	Pass	Logged in successfully	N	3

Test case ID	Feature Type	Component	Test Scenario	Pre-Requirement	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with invalid credentials	User credentials, Database with credentials of existing users	1.Enter URL(https://shopnizer.com/) and click go 2.Click on My Account dropdown button 3.Enter invalid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123	Application should show "incorrect email or password" validation message.	Working as expected	Fail	Login failed due to incorrect login details or user not registered	N	4
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with invalid credentials	User credentials, Database with credentials of existing users	1.Enter URL(https://shopnizer.com/) and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter invalid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123678686786876876	Application should show "incorrect email or password" validation message.	Working as expected	Fail	Login failed due to incorrect login details or user not registered	N	5
LoginPage_TC_005	Functional	Login page	Verify user is able to log into application with invalid credentials	User credentials, Database with credentials of existing users	1.Enter URL(https://shopnizer.com/) and click go 2.Click on My Account dropdown button 3.Enter invalid username/email in Email text box 4.Enter invalid password in password text box	Username: chalam@gmail.com password: Testing123678686786876876	Application should show "incorrect email or password" validation message.	Working as expected	Fail	Login failed due to incorrect login details or user not registered	N	6

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

1. Hours worked: 50 hours
2. Stick to Timelines: 100%
3. Stay within budget: 100%
4. Consistency of the product: 85%
5. Efficiency of the product: 85%
6. Quality of the product: 85%

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES:

- Smooth User Interface
- Accuracy is achieved quickly

DISADVANTAGES:

- Random forest can be used for both classification and regression tasks, but it is not more suitable for Regression tasks.

CHAPTER 11

CONCLUSION

This overview of the project conveys the idea that numerous methods have been investigated for diagnosing cardiovascular disease. Big data, machine learning, and data mining can be used to great success to analyse the prediction model with the highest degree of accuracy. The primary goal of this project is to diagnose cardiovascular disease or heart disease utilizing a variety of techniques and procedures to obtain a prognosis.

CHAPTER 12

FUTURE SCOPE

A future update shall comprise of section for viewing renowned cardiologists and scan centres in their city. The obtained output can be further processed and sent to smart devices to provide necessary assistance. Constant monitoring can provide necessary data to recommend to consult a doctor in case of an emergency.

CHAPTER 13

APPENDIX

PROJECT DEMONSTRATION LINK:

https://drive.google.com/file/d/1CDdldB9PRdRVz5q4toE1WhzUFduw0rlz/view?usp=share_link

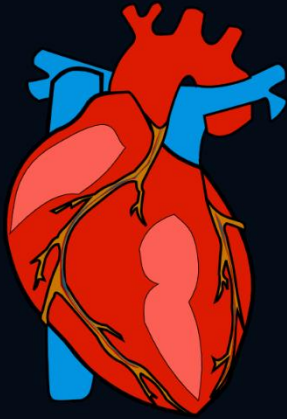
GITHUB LINK: <https://github.com/IBM-EPBL/IBM-Project-20354-1659717940/tree/main>

APPENDIX A1: SCREENSHOTS

Visualizing and Predicting Heart Diseases with an Interactive Dash Board

This Tool provides you the option of diagnosing whether you have heart disease or not without visiting a hospital.

[Get Started](#)



Causes of heart diseases.

Smoking

Smoking is a major risk factor for coronary heart disease. Both nicotine and carbon monoxide (from the smoke) put a strain on the heart by making it work faster. They also increase your risk of blood clots. Other chemicals in cigarette smoke can damage the lining of your coronary arteries, leading to furring of the arteries. Smoking significantly increases your risk of developing heart disease.

High blood pressure

High blood pressure (hypertension) puts a strain on your heart and can lead to CHD.

High cholesterol

Cholesterol is a fat made by the liver from the saturated fat in your diet. It is essential for healthy cells, but too much in the blood can lead to CHD.

Lack of regular exercise

If you are inactive, fatty deposits can build up in your arteries. If the arteries that supply blood to your heart become blocked, it can lead to a heart attack. If the arteries that supply blood to your brain are affected it can cause a stroke. Read about the physical activity guidelines for adults aged 19 to 64.

Diabetes

A high blood sugar level may lead to diabetes, which can more than double your risk of developing CHD. Diabetes can lead to CHD because it may cause the lining of blood vessels to become thicker, which can restrict blood flow.



"Welcome You"

Name

e-mail

Password

Confirm Password

Already have an account?

Enter your details

Age

Sex

Chest Pain Type

Blood Pressure

Cholesterol

Fasting Blood Sugar Over 120

EKG Results

Maximum Heart Rate

Exercise Induced Angina

ST Depression

Slope of ST

Number of major vessels colored by Flouroscopy



"Welcome You"

Based on the parameters provided, our ML model has predicted the Presence of heart disease

Do you want to save the results?

[Save](#)
[Dashboard](#)

Welcome to your Dashboard!

Check out your email. Your results have been sent there.

Here you can check out different kind of visualizations to get a general idea about the factors increasing causes of getting a cardiac arrest.

The visualizations have been made with the help of Tableau

85%
Accuracy

5
Visualization Types

Random Forest Classifier
ML Model

Cholesterol by Age

Over all the ages, the average of cholesterol is 249.7

The average values of Cholesterol range from 196, occurring when Age is 34, to 304, when Age is 77.

