## Visualizing and Predicting Heart Diseases With an Interactivate Dashboard

Submitted By Rahul S (737819CSR151)

Poorva Kowshik V S (737819CSR134)

Pravin Kumar P (737819CSR143)

Rithani A M (737819CSR162)

# Project Report

#### INTRODUCTION

* 1. Project Overview
  2. Purpose

#### LITERATURE SURVEY

* 1. Existing problem
  2. References
  3. Problem Statement Definition

#### IDEATION & PROPOSED SOLUTION

* 1. Empathy Map Canvas
  2. Ideation & Brainstorming
  3. Proposed Solution
  4. Problem Solution fit

#### REQUIREMENT ANALYSIS

* 1. Functional requirement
  2. Non-Functional requirements

#### PROJECT DESIGN

* 1. Data Flow Diagrams
  2. Solution & Technical Architecture
  3. User Stories

#### PROJECT PLANNING & SCHEDULING

* 1. Sprint Planning & Estimation
  2. Sprint Delivery Schedule
  3. Reports from JIRA

#### CODING & SOLUTIONING

1. **TESTING**
   1. Test Cases
   2. User Acceptance Testing

#### RESULTS

* 1. Performance Metrics **10.ADVANTAGES & DISADVANTAGES 11.CONCLUSION**

#### FUTURE SCOPE

1. **APPENDIX**

**CHAPTER 1**

# INTRODUCTION

### 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 of its components is characterized by these medical conditions.

In order 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.

### 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 treatedappropriately. 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 inthe 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

### EXISTING PROBLEM

### The diagnosis of heart disease in most cases depends on a complex

### combination of clinical and pathological data. Because of this complexity,

### there exists a significant amount of interest among clinical professionals

### and researchers regarding the efficient and accurate prediction of heart

### disease. In this paper, develop a heart disease predict system that can

### assist medical professionals in predicting heart disease status based on

### the clinical data of patients. These approaches include three steps.

### Firstly, select 13 important clinical features, i.e., age, sex, chest pain

### type, trestbps, cholesterol, fasting blood sugar, resting ecg, max heart

### rate, exercise induced angina, old peak, slope, number of vessels

### colored, and thal. Secondly, develop an artificial neural network

### algorithm for classifying heart disease based on these clinical features.

### The accuracy of prediction is near 80%. Finally, develop a user-friendly

### heart disease predict system (HDPS). The HDPS system will be

### consisted of multiple features, including input clinical data section,

### ROC curve display section, and prediction performance display section

### (execute time, accuracy, sensitivity, specificity, and predict result).

### Our approaches are effective in predicting the heart disease of a patient.

### The HDPS system developed in this study is a novel approach that

### can be used in the classification of heart disease.

 References:

1. C. Sowmiya and P. Sumitra, "Analytical study of heart disease diagnosis using classification techniques," 2017
2. IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS),

2017, pp. 1-5, doi: 10.1109/ITCOSP.2017.8303115.

1. J. Thomas and R. T. Princy, "Human heart disease prediction system using data mining techniques," 2016

International Conference on Circuit, Power and Computing Technologies (ICCPCT), 2016, pp. 1-5,

doi: 10.1109/ICCPCT.2016.7530265.

1. M. S. Raja, M. Anurag, C. P. Reddy and N. R. Sirisala, "Machine Learning Based Heart Disease Prediction System,

" 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-5,

doi: 10.1109/ICCCI50826.2021.9402653.

1. N. M. Lutimath, N. Sharma and B. K. Byregowda, "Prediction of Heart Disease using Random Forest," 2021

Emerging Trends in Industry 4.0 (ETI 4.0), 2021, pp. 1-4, doi: 10.1109/ETI4.051663.2021.9619208.

1. G. N. Ahmad, H. Fatima, S. Ullah, A. Salah Saidi and Imdadullah, "Efficient Medical Diagnosis of Human Heart

Diseases Using Machine Learning Techniques With and Without GridSearchCV," in IEEE Access, vol. 10, pp.

80151-80173, 2022, doi: 10.1109/ACCESS.2022.3165792.

1. J. P. Li, A. U. Haq, S. U. Din, J. Khan, A. Khan and A. Saboor, "Heart Disease Identification Method Using

Machine Learning Classification in E-Healthcare," in IEEE Access, vol. 8, pp. 107562-107582, 2020,

doi: 10.1109/ACCESS.2020.3001149.

7. A. H. Chen, S. Y. Huang, P. S. Hong, C. H. Cheng and E. J. Lin, "HDPS: Heart disease prediction system," 2011 Computing in Cardiology, 2011, pp. 557-560.

8. F. Demir, A. Şengür and M. Çavaş, "HEART SOUNDS CLASSIFICATION WITH DEEP FEATURES AND

SUPPORT VECTOR MACHINES," 2018 International Conference on Artificial Intelligence and Data Processing

(IDAP), 2018, pp. 1-5, doi: 10.1109/IDAP.2018.8620733.

9. T. Feng, H. Tang, M. Wang, C. Zhang, H. Wang and F. Cong, "Continuous Estimation of Left Ventricular

Hemodynamic Parameters Based on Heart Sound and PPG Signals Using Deep Neural Network," 2020 International Conference on Sensing, Measurement & Data Analytics in the era of Artificial Intelligence (ICSMD), 2020,

pp. 313-318, doi: 10.1109/ICSMD50554.2020.9261681.

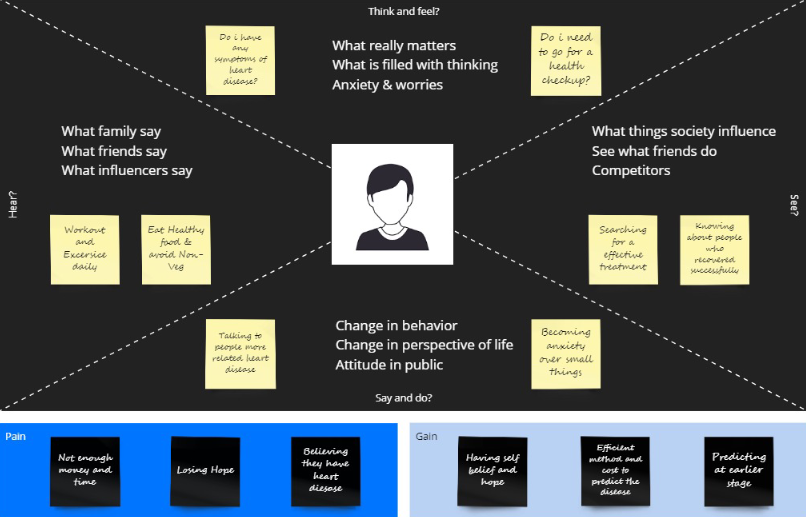
10. M. Chourasia, A. Thakur, S. Gupta and A. Singh, "ECG Heartbeat Classification Using CNN," 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON),

2020, pp. 1-6, doi: 10.1109/UPCON50219.2020.93764

#### CHAPTER 3

#### IDEATION & PROPOSED SOLUTION

**EMPATHY MAP CANVAS**

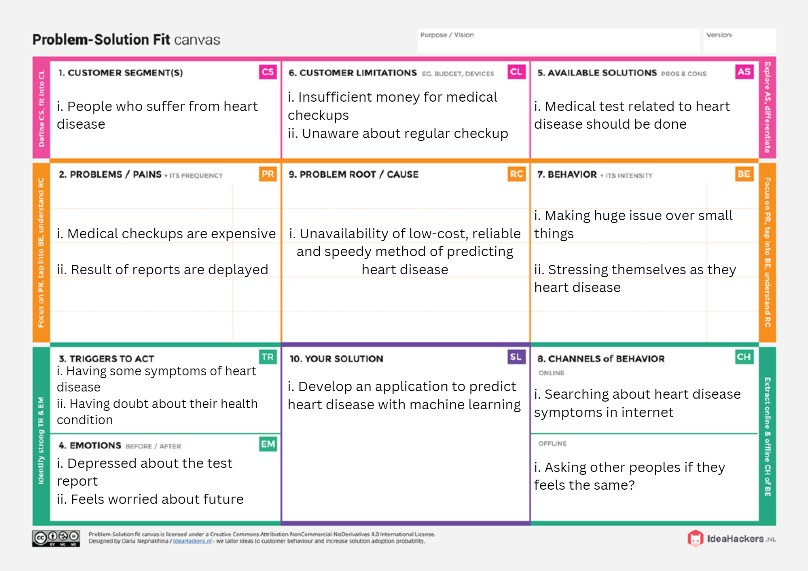
****

### PROPOSED SOLUTION

|  |  |  |
| --- | --- | --- |
| **S.No** | **Parameter** | **Description** |
| **1.** | Problem Statement (Problem to be solved) | To analyse which patients are most likely to suffer from heart disease based on givenparameters.It can provide visualization dashboards and uses this information to easily visualize and predict the patient details |
| **2.** | Idea / Solution description | Parameters in data set helps hospitals toidentify the patient heart condition andtheir health condition.A dashboard using cognitive analysis can be created to present the data and utilize it for future use |
| **3.** | Novelty / Uniqueness | Many tests are taken by doctors to detect presence of heart disease. The parameters used are often understood only by medical professional.Time can be saved.To providea significant contribution in computingstrength scores with significant predictors inHeart disease prediction |
| **4.** | Social Impact / Customer Satisfaction | Reduces the patient’s risk level Reduces the medical cost Save human lives.Handy Interactive dashboard It will make the hospital to work efficiently It help the hospitals to knowthe healthrecords of the heart patientλ |
| **5.** | Business Model (Revenue Model) | Awareness can be created amongthepatients through ads Updates will be updated accordingtothe necessity for the patients No complexity Data security This project can be converted to ansoftware kit, webpage or even an application which users can interact with. |
| **6.** | Scalability of the Solution | Maintains best user experiences Disease Easy prediction of the patient details with heart Adding new characteristics Scalable dataset Machine learning λ |

**PROBLEM SOLUTION FIT**

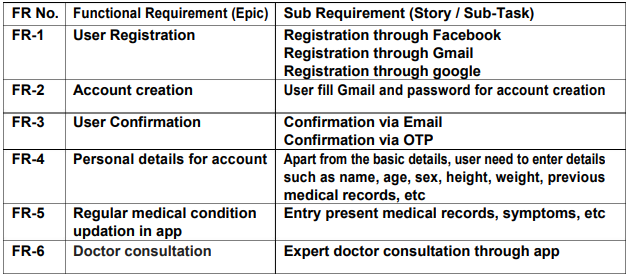
The Problem-Solution Fit simply means that we have found a problem with our customer and that the solution we have realized for it actually solves the customer’s problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why. The 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 behaviour



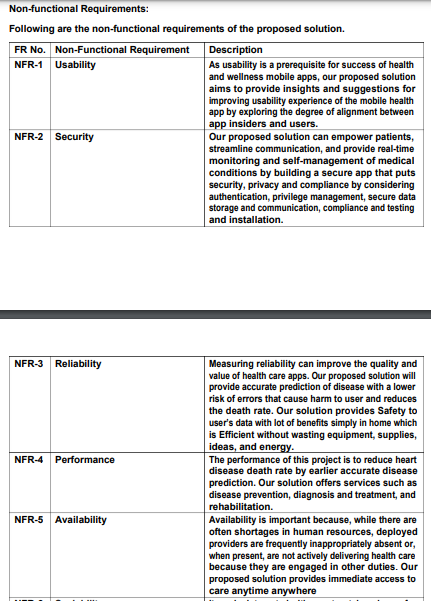
**CHAPTER 4**

## REQUIREMENT ANALYSIS

**FUNCTIONAL REQUIREMENTS**



**NON-FUNCTIONAL REQUIREMENTS**

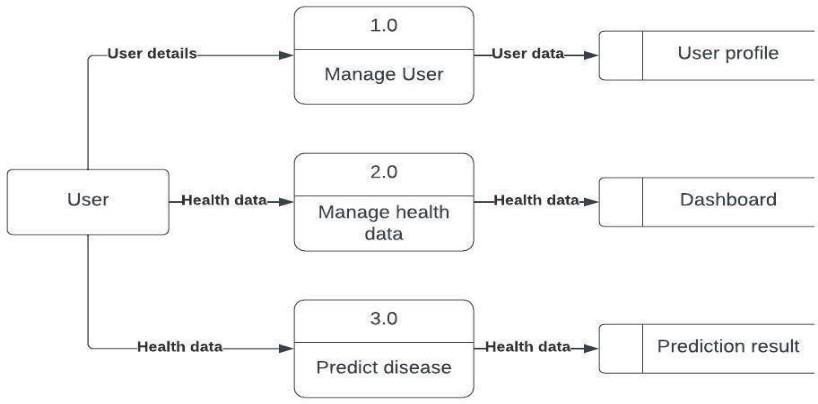
****

## CHAPTER 5 PROJECT DESIGN

### Data Flow Diagrams ,Solution & Technical Architecture

**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a graphical representation of the flow of datain a business information system. It describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation. It shows how data enters and leaves the system, what changes the information, and where data is stored.

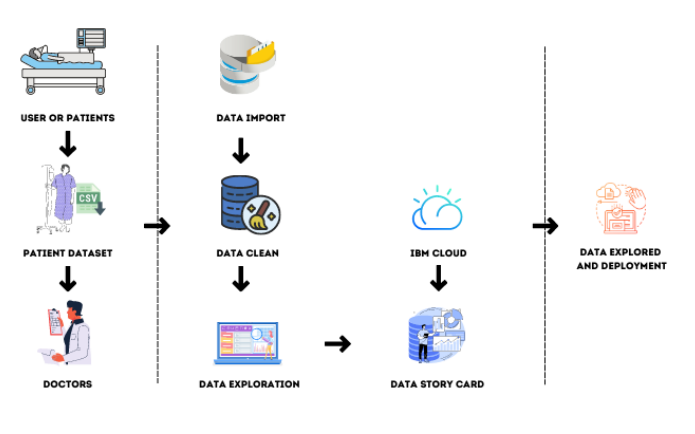


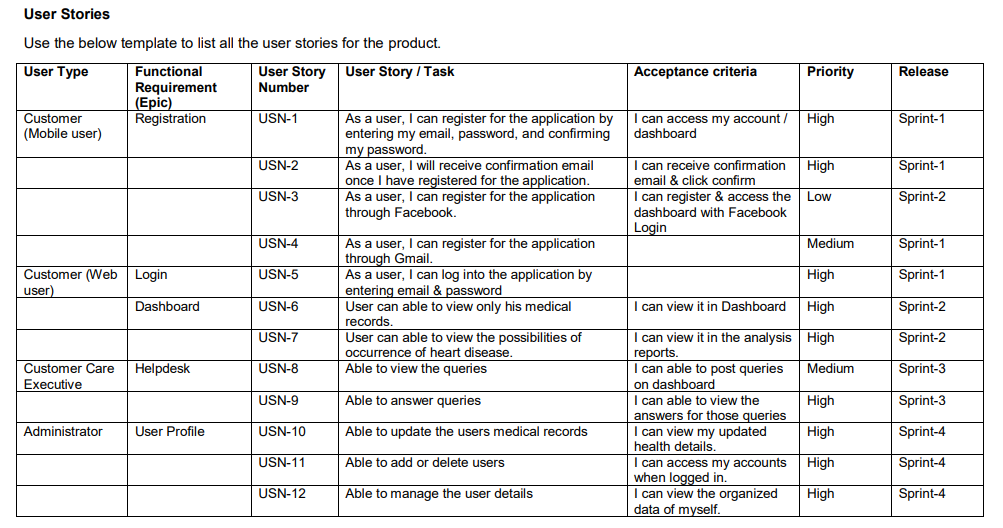
#### Solution and Architecture diagram:

**Solution Architecture:**

Solution architecture is a complex process with many sub-processes that bridges the gap between business problems and technology solutions. Its goals are to:

* Find the best tech solution to solve existing business problems.
* Describe the structure, characteristics, behavior, and other aspects of the software to projectstakeholders.
* Define features, development phases, and solution requirements.
* Provide specifications according to which the solution is defined, managed, and delivered.

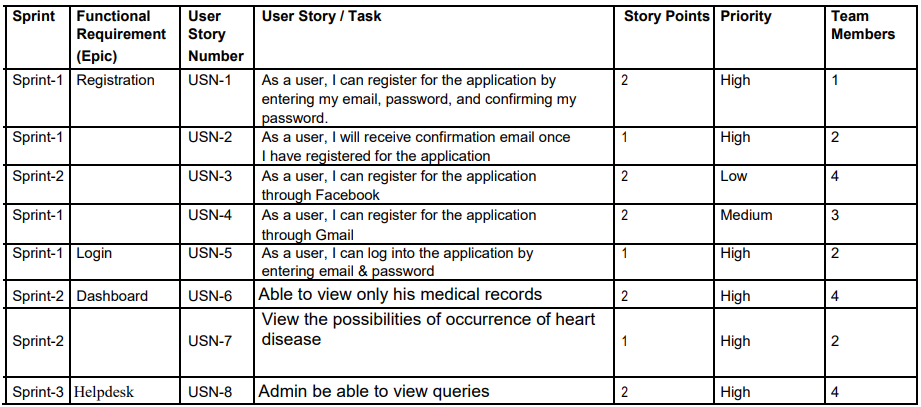




#### ProjectPlanning Phase

**Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)**

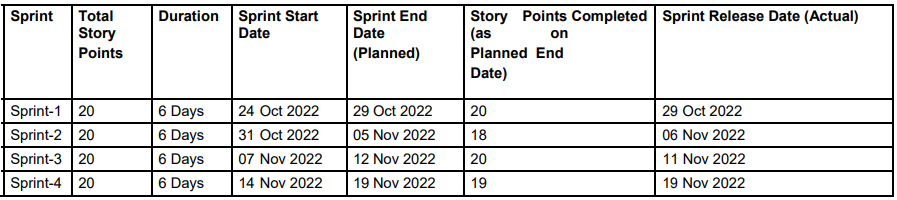
|  |  |
| --- | --- |
| Team ID | PNT2022TMID04459 |
| Project Name | Project - Visualizing and Predicting Heart  Diseases  with an Interactive Dashboard |

****

**Burndown Chart:**

**A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down chartscan be applied to any project containing measurable progressover time.**

#### 



#### CHAPTER 7 CODING & SOLUTIONING

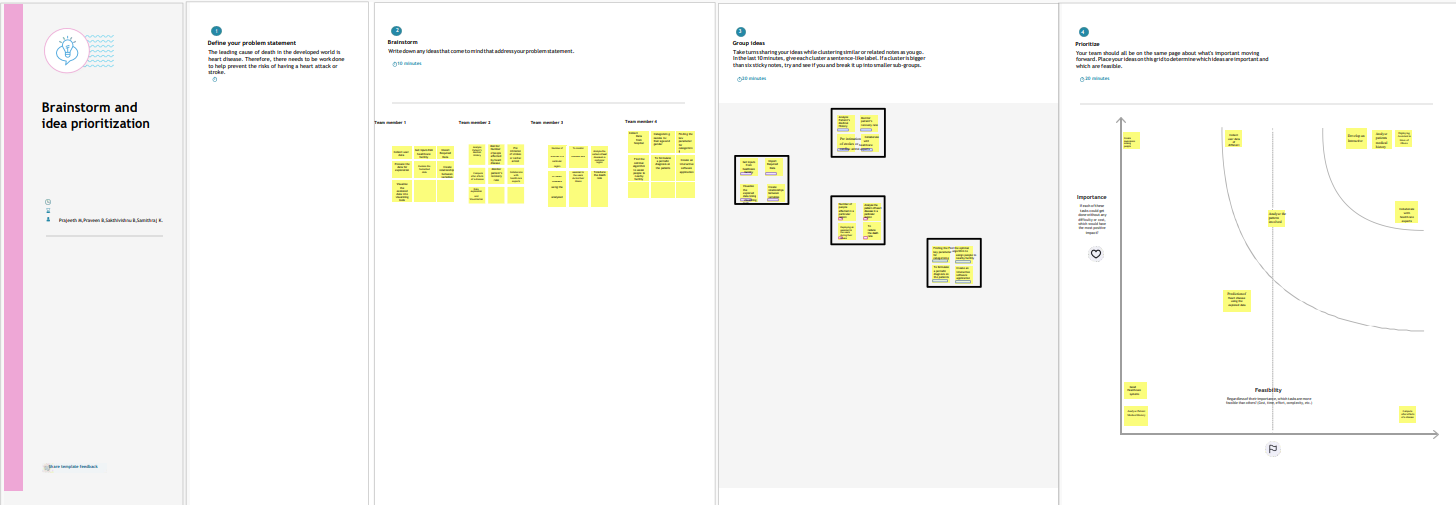
**Feature 1: Log In Login.jsx:**

|  |
| --- |
|  |

Login.jsx

import React, { useState } from "react";  
import { Link, useNavigate } from "react-router-dom";  
import { GoogleButton } from "react-google-button";  
import { UserAuth } from "../context/AuthContext";  
import Button from "react-bootstrap/Button";  
import Form from "react-bootstrap/Form";  
import axios from "axios";  
import Joi from "joi";  
import toast from "react-hot-toast";  
import jwtDecode from "jwt-decode";  
import signInImg from "../assets/img1.jpg";  
  
const SignIn = () => {  
  const { userState, googleSignIn } = UserAuth();  
  
  const [user, setUser] = userState;  
  
  const [emailError, setEmailError] = useState();  
  const [passwordError, setPasswordError] = useState();  
  
  const signInEmailRef = React.createRef();  
  const signInPasswordRef = React.createRef();  
  
  const navigate = useNavigate();  
  
  const handleSignIn = async (e) => {  
    e.preventDefault();  
    const email = signInEmailRef.current.value;  
    const password = signInPasswordRef.current.value;  
    console.log({ email, password });  
  
    // VALIDATION  
    const emailSchema = Joi.object({  
      email: Joi.string()  
        .required()  
        .email({ tlds: { allow: false } }),  
    });  
    const passwordSchema = Joi.object({  
      password: Joi.string().min(8).max(20).required(),  
    });  
  
    const emailErr = emailSchema.validate({ email }).error;  
    const passwordErr = passwordSchema.validate({ password }).error;  
  
    if (emailErr && emailErr.message) {  
      setEmailError("email" + emailErr.message.slice(7));  
    } else {  
      setEmailError(null);  
    }  
    if (passwordErr && passwordErr.message) {  
      setPasswordError("password" + passwordErr.message.slice(10));  
    } else {  
      setPasswordError(null);  
    }  
  
    if (!emailErr && !passwordErr) {  
      await axios  
        .post("<http://localhost:8000/auth/sign-in>", { email, password })  
        .then((res) => {  
          sessionStorage.setItem("token", res.data);  
          console.log(res.data);  
          setUser(jwtDecode(res.data));  
          toast.success(`Welcome ${jwtDecode(res.data).username}`);  
          navigate("/");  
        })  
        .catch((err) => {  
          console.log(err.response.data);  
          if (err.response.data === "Invalid Password") {  
            setPasswordError(err.response.data);  
          } else {  
            setEmailError(err.response.data);  
          }  
        });  
    }  
  };  
  
  const handleGoogleSignIn = async () => {  
    try {  
      await googleSignIn();  
    } catch (error) {  
      console.log(error);  
    }  
  };  
  return (  
    <div  
      style={{  
        width: "100%",  
        display: "flex",  
        justifyContent: "center",  
        alignItems: "center",  
        marginTop: "56px",  
        // border: "5px solid black",  
      }}  
    >  
      <div  
        style={{  
          height: "800px",  
          // border: "2px solid red",  
          display: "flex",  
          justifyContent: "center",  
          alignItem: "center",  
        }}  
        className="col-lg-6 col-sm-12"  
      >  
        <div style={{ width: "300px", marginTop: "150px" }}>  
          <h3>  
            <i>Sign In</i>  
          </h3>  
          <Form>  
            <Form.Group className="mb-3" controlId="formBasicEmail">  
              <Form.Label>Email address</Form.Label>  
              <Form.Control  
                type="email"  
                placeholder="Enter email"  
                ref={signInEmailRef}  
                defaultValue={"[test@gmail.com](mailto:test@gmail.com)"}  
              />  
              {emailError && <i className="text-danger">\* {emailError}</i>}  
            </Form.Group>  
  
            <Form.Group className="mb-3" controlId="formBasicPassword">  
              <Form.Label>Password</Form.Label>  
              <Form.Control  
                type="password"  
                placeholder="Password"  
                ref={signInPasswordRef}  
                defaultValue={"12345678"}  
              />  
              {passwordError && (  
                <i className="text-danger">\* {passwordError}</i>  
              )}  
            </Form.Group>  
            <Link  
              to="/forgot-password"  
              style={{  
                textDecoration: "none",  
                float: "right",  
                marginTop: "-8px",  
                marginBottom: "10px",  
              }}  
            >  
              Forgot password?  
            </Link>  
            <Button  
              variant="dark"  
              type="submit"  
              style={{ width: "300px" }}  
              onClick={handleSignIn}  
            >  
              Submit  
            </Button>  
            <br />  
  
            <h6 className="text-center">or</h6>  
          </Form>  
          <GoogleButton  
            onClick={handleGoogleSignIn}  
            className="bg-dark"  
            style={{ width: "300px" }}  
          />  
          <br />  
          <p className="text-center">  
            <Link  
              to="/sign-up"  
              style={{  
                textDecoration: "none",  
                marginTop: "-10px",  
                marginBottom: "10px",  
              }}  
            >  
              Doesn't have an account? Sign Up  
            </Link>  
          </p>  
        </div>  
      </div>  
      <div  
        style={{  
          height: "800px",  
          // border: "2px solid red",  
          display: "flex",  
          justifyContent: "center",  
          alignItem: "center",  
          backgroundColor: "#011945",  
        }}  
        className="col-lg-6 col-sm-12"  
      >  
        <div style={{ marginTop: "130px" }}>  
          <img  
            src={signInImg}  
            alt=""  
            width={600}  
            style={{ borderRadius: "20px" }}  
          />  
        </div>  
      </div>  
  
      {/\* <div className="col-lg-6 col-sm-12">  
        <div style={{ width: "300px" }}>  
          <h3>Sign In</h3>  
          <Form>  
            <Form.Group className="mb-3" controlId="formBasicEmail">  
              <Form.Label>Email address</Form.Label>  
              <Form.Control  
                type="email"  
                placeholder="Enter email"  
                ref={signInEmailRef}  
                defaultValue={"[test@gmail.com](mailto:test@gmail.com)"}  
              />  
              {emailError && (  
                <Form.Text className="text-danger">\* {emailError}</Form.Text>  
              )}  
            </Form.Group>  
  
            <Form.Group className="mb-3" controlId="formBasicPassword">  
              <Form.Label>Password</Form.Label>  
              <Form.Control  
                type="password"  
                placeholder="Password"  
                ref={signInPasswordRef}  
                defaultValue={"12345678"}  
              />  
              {passwordError && (  
                <Form.Text className="text-danger">\* {passwordError}</Form.Text>  
              )}  
            </Form.Group>  
            <Link  
              to="/forgot-password"  
              style={{  
                textDecoration: "none",  
                float: "right",  
                marginTop: "-8px",  
                marginBottom: "10px",  
              }}  
            >  
              Forgot password?  
            </Link>  
            <Button  
              variant="dark"  
              type="submit"  
              style={{ width: "300px" }}  
              onClick={handleSignIn}  
            >  
              Submit  
            </Button>  
            <br />  
  
            <h6 className="text-center">or</h6>  
          </Form>  
          <GoogleButton  
            onClick={handleGoogleSignIn}  
            className="bg-dark"  
            style={{ width: "300px" }}  
          />  
          <br />  
          <p className="text-center">  
            <Link  
              to="/sign-up"  
              style={{  
                textDecoration: "none",  
                marginTop: "-10px",  
                marginBottom: "10px",  
              }}  
            >  
              Doesn't have an account? Sign Up  
            </Link>  
          </p>  
        </div>  
      </div>  
      <div style={{border: "2px solid black"}}>  
        <img src={signInImg} alt="" width={600} />  
      </div> \*/}  
    </div>  
  );  
};

|  |  |  |  |
| --- | --- | --- | --- |
| **Ideation Phase Brainstorm&Idea PrioritizationTemplate** | | |  |
|  | Team ID | PNT2022TMID04459 |  |
| Project Name | Visualizing and predicting heart diseases  withan interactive dashborad |  |
| **Brainstorm & Idea Prioritization Template:**  Brainstorming provides a free and open environment that encourages everyone withina team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amountof creative solutions.  Use this template in your own brainstorming sessions so your team can unleash theirimagination and start shaping concepts even if you're not sitting in the same room.  **Step-1: Team Gathering, Collaboration and Select the Problem Statement** | | |  |



<p>The model that we are going to use to predict the disease is Logistic Regression.

The Training and Testing accuracy was recorded 87 and 83 respectively.</p>

</section>

</div>

</body>

</html>

#### Feature 4:

|  |
| --- |
|  |
|  | <html>  <head> |
|  | <!-- Bootstrap CSS --> |
|  | <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" integrity="sha384-JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z" crossorigin="anonymous"> |
|  | <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script> |
|  | <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script> |
|  | <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script> |
|  | <title>Heart Disease Test</title> |
|  | </head> |
|  | <body> |
|  | <!-- Java Script --> |
|  | <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj" crossorigin="anonymous"></script> |
|  | <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH81PoYlFhbGU+6BZp6G7niu735Sk7lN" crossorigin="anonymous"></script> |
|  | <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPlYxofvL8/KUEfYiJOMMV+rV" crossorigin="anonymous"></script> |
|  |  |
|  |  |
|  | <!-- Navbar--> |
|  | <nav class="navbar navbar-dark" style="background-color: rgb(13, 102, 87);"> |
|  | <span class="navbar-brand mb-0 h1">Heart Disease Test</span> |
|  | </nav> |
|  | <div class="container"> |
|  | <br> |
|  | <!--Form--> |
|  | <form action = "{{url\_for('predict')}}" method ="POST" > |
|  | <fieldset> |
|  | <legend style="color: rgb(41, 15, 134);"><b>Heart Disease Test Form</b></legend><br> |
|  | <div class="card card-body" style="background-color:  rgb(194 245 236 / 56%);"> |
|  | <div class="form-group row"> |
|  | <div class="col-sm-3"> |
|  | <label for="age">Age</label> |
|  | <input type="number" class="form-control" id="age" name="age" required> |
|  | </div> |
|  | <div class="col-sm-3"> |
|  | <label for="sex">Sex</label> |
|  | <select class="form-control" id="sex" name="sex" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "0">Female</option> |
|  | <option value = "1">Male</option> |
|  | </select> |
|  | </div> |
|  | </div> |
|  | <br> |
|  | <div class="form-group row"> |
|  | <div class="col-sm"> |
|  | <label for="cp">Chest Pain Type</label> |
|  | <select class="form-control" id="cp"  name = "cp" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "1">Typical Angina</option> |
|  | <option value = "2">Atypical Angina</option> |
|  | <option value = "3">Non-anginal Pain</option> |
|  | <option value = "4">Asymptomatic</option> |
|  | </select> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="trestbps">Resting Blood Pressure in mm Hg</label> |
|  | <input type="number" class="form-control" id="trestbps" name="trestbps" required> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="chol">Serum Cholestoral in mg/dl</label> |
|  | <input type="number" class="form-control" id="chol" name="chol" required> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="fbs">Fasting Blood Sugar > 120 mg/dl</label> |
|  | <select class="form-control" id="fbs"  name="fbs" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "0">False</option> |
|  | <option value = "1">True</option> |
|  | </select> |
|  | </div> |
|  | </div> |
|  |  |
|  | <br> |
|  | <div class="form-group row"> |
|  | <div class="col-sm"> |
|  | <label for="restecg">Resting ECG Results  </label> |
|  | <select class="form-control" id="restecg" name="restecg" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "0">Normal </option> |
|  | <option value = "1">  Having ST-T wave abnormality </option> |
|  | <option value = "2">Probable or definite left ventricular hypertrophy</option> |
|  | </select> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="thalach">Maximum Heart Rate</label> |
|  | <input type="number" class="form-control" id="thalach" name="thalach" required> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="exang">Exercise Induced Angina  </label> |
|  | <select class="form-control" id="exang" name="exang" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "0">No</option> |
|  | <option value = "1">Yes</option> |
|  | </select> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="oldpeak">ST Depression Induced  </label> |
|  | <input type="number" step="any" class="form-control" id="oldpeak" name="oldpeak" required> |
|  | </div> |
|  | </div> |
|  | <br> |
|  | <div class="form-group row"> |
|  | <div class="col-sm"> |
|  | <label for="slope">Slope of the Peak Exercise  ST Segment </label> |
|  | <select class="form-control" id="slope" name="slope" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "1">Upsloping</option> |
|  | <option value = "2">Flat</option> |
|  | <option value = "3">Downsloping</option> |
|  | </select> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="ca">Number of Vessels Colored by Flourosopy</label> |
|  | <select class="form-control" id="ca" name = "ca"  required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "0">0</option> |
|  | <option value = "1">1</option> |
|  | <option value = "2">2</option> |
|  | <option value = "3">3</option> |
|  | </select> |
|  | </div> |
|  | <div class="col-sm"> |
|  | <label for="thal">Thalassemia</label> |
|  | <select class="form-control" id="thal" name = "thal" required> |
|  | <option disabled selected value> -- Select an Option -- </option> |
|  | <option value = "3">Normal</option> |
|  | <option value = "6">Fixed defect</option> |
|  | <option value = "7">Reversable defect</option> |
|  | </select> |
|  | </div> |
|  | </div> |
|  | <br> |
|  | <div class="form-group"> |
|  | <input class="btn btn-primary" type="submit" value="Result"> |
|  | </div> |
|  |  |
|  | <!--Prediction Result--> |
|  | <div id ="result"> |
|  | <strong style="color:red">{{result}}</strong> |
|  | </div> |
|  | </div> |
|  | </fieldset> |
|  | </form> |
|  |  |
|  | </div> |
|  |  |
|  | </body> |
|  | </html> |

#### Integration:

PredictionForm.jsx

|  |
| --- |
|  |

import React from "react";  
import { UserAuth } from "../context/AuthContext";  
import UserDetailsForm from "./UserDetailsForm";  
  
const UserDetails = () => {  
  const { userState, signUpDetailsState } = UserAuth();  
  const [user, setUser] = userState;  
  const [signUpDetails, setSignUpDetails] = signUpDetailsState;  
  return (  
    <div style={{ backgroundColor: "#001e3f"}}>  
      <div style={{ marginTop: "56px" }}>  
      {!user && (  
        <h3 className="text-center" style={{marginTop: "50px", color: "white"}}>  
          <i>Please fill the below details to continue signing up</i>  
        </h3>  
      )}  
      <UserDetailsForm />  
      </div>  
    </div>  
  );  
};  
  
export default UserDetails;

#### Styleshe et styles,cs s

.visual{

text-align: center; background-color: #000;

}

.topic{

text-decoration: none;font-size: 1.5em; color: crimson;

font-weight: bolder; margin- bottom: 32px;

}

.topic:hover{ color: crimson;

text-decoration: none;

}

.navbar-light .navbar-nav .nav-item

.nav-link{color:crimson; font-weight: bold;

}

.navbar-light .navbar-nav .nav-item .nav-

link:hover{color:black;

}

.about{

text-align: center;

}

.list{

text-align: justify;margin- left: 38%;

}

.wel{

color: crimson;

}

.login{

text-align: center; margin-top: 10%; background-color: black;margin-left: 35%;

padding: 5%; width: fit- content;

}

.ll{

color: #fff;

font-weight: bold;

}

.btn{

margin-top: 16px; background-color: crimson;color: #fff; font-weight: bold;

}

.title{

font-size: 1.5em;color: #fff;

}

.ag{

margin-top: 0;

}

### CHAPTER 8

**TESTING**

#### Project Development PhaseModel Performance Test

|  |  |
| --- | --- |
| Team ID | PNT2022TMID04459 |
| Project Name | Visualizing and Predicting Heart Diseases with an Interactive Dash Board |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Screenshot / Values** |
| 1. | Dashboard design | No of Visulizations / Graphs - 10 |
| 2. | Data Responsiveness | Good |
| 3. | Amount Data to Rendered (DB2 Metrics) | - |
| 4. | Utilization of Data Filters | Yes for filtering out visualisations concerning people with existing heart disease |
| 5. | Effective User Story | No of Scene Added - 8 |
| 6. | Descriptive Reports | No of Visulizations / Graphs - 7 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **1.**  **2.** | **Acceptance Testing**  **UAT Execution & Report Submission**  **Purpose of Document**  The purpose of this document is to briefly explain the test coverage and op issues of the [ProductName] project at the time of the release to Us Acceptance Testing (UAT).  **Defect Analysis**  This report shows the number of resolved or closed bugs at each severity level, and how they were resolved | | | | | n er |
|  | **Resolution** | **Severity 1** | **Severity 2** | **Severity 3** | **Severity 4** | **Subtotal** |
|  | By Design | 10 | 4 | 2 | 3 | 20 |
| Duplicate | 1 | 0 | 3 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 11 | 2 | 4 | 20 | 37 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 0 | 1 | 1 | 2 |
| Won't Fix | 0 | 5 | 2 | 1 | 8 |
| Totals | 24 | 14 | 13 | 26 | 77 |

|  |  |
| --- | --- |
| Team ID | PNT2022TMID04459 |
| Project Name | Visualizing and Predicting Heart Diseaseswith an Interactive Dash Board |

#### Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Section** | **Total Cases** | **Not Tested** | **Fail** | **Pass** |
| Print Engine | 7 | 0 | 0 | 7 |
| Client Application | 51 | 0 | 0 | 51 |
| Security | 2 | 0 | 0 | 2 |
| Outsource Shipping | 3 | 0 | 0 | 3 |
| Exception Reporting | 9 | 0 | 0 | 9 |
| Final Report Output | 4 | 0 | 0 | 4 |
| Version Control | 2 | 0 | 0 | 2 |

#### RESULTS 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%

#### ADVANTAGES & DISADVANTAGES:

**ADVANTAGES:**

* Smooth User Interface
* Accuracy is achieved quickly

**DISADVANTAGES:**

Random forest can be used for both classification and regression tasks, butit is not more suitable for Regression tasks

* 1. **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 oftechniques and procedures to obtain a prognosis.

* 1. **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 caseof an emergency.