

Project Report

- 1. INTRODUCTION**
 - 1.1 Project Overview
 - 1.2 Purpose
- 2. LITERATURE SURVEY**
 - 2.1 References
 - 2.2 Problem Statement Definition
- 3. IDEATION & PROPOSED SOLUTION**
 - 3.1 Empathy Map Canvas
 - 3.2 Ideation , Brainstorming & Prioritize
 - 3.3 Proposed Solution
 - 3.4 Problem Solution fit
- 4. REQUIREMENT ANALYSIS**
 - 4.1 Functional requirement
 - 4.2 Non-Functional requirements
- 5. PROJECT DESIGN**
 - 5.1 Data Flow Diagrams
 - 5.2 Solution & Technical Architecture
 - 5.3 User Stories
- 6. PROJECT PLANNING & SCHEDULING**
 - 6.1 Sprint Planning & Estimation
 - 6.2 Sprint Delivery Schedule
 - 6.3 Reports from JIRA
- 7. CODING & SOLUTIONING (Explain the features added in the project along with code)**
 - 7.1 Feature 1
 - 7.2 Feature 2
 - Database Schema (if Applicable)
- 8. TESTING**
 - 8.1 Test Cases
 - 8.2 User Acceptance Testing
- 9. RESULTS**
 - 9.1 Performance Metrics
- 10. ADVANTAGES & DISADVANTAGES**
- 11. CONCLUSION**
- 12. FUTURE SCOPE**
- 13. APPENDIX**
 - 13.1 Source Code
 - 13.2 GitHub & Project Demo Link

1. INTRODUCTION

1.1 Project Overview

Recently, clinicians have been actively engaged in improving medical diagnoses. The use of artificial intelligence and machine learning in combination with clinical findings has further improved disease detection. In the modern era, with the advantage of computers and technologies, one can collect data and visualize many hidden outcomes such as dealing with missing data in medical research. Statistical machine learning algorithms based on specific problems can assist one to make decisions. Machine learning (ML), data-driven algorithms can be utilized to validate existing methods and help researchers to make potential new decisions.

1.2 Purpose

People who want to know whether they have liver disease or not.

2. LITERATURE SURVEY

2.1 References

S. No	Paper	Year	Citation	Methodologies used
1	Statistical Machine Learning Approaches to Liver Disease Prediction	2021	Mostafa, F.; Hasan, E.; Williamson, M.; Khan, H. Statistical Machine Learning Approaches to Liver Disease Prediction. <i>Livers</i> 2021 , <i>1</i> , 294–312. https://doi.org/10.3390/livers1040023	The purpose of this study was to extract significant predictors for liver disease from the medical analysis of 615 humans using ML algorithms. Data visualizations were implemented to reveal significant findings such as missing values.

2	Performance Analysis of Liver Disease Prediction Using Machine Learning Algorithms	2018	International Research Journal of Engineering and Technology (IRJET) www.irjet.net	<p>P.Rajeswari,G.Sophia Reena et al.,[2010]has proposed the data classification is based on liver disorder. The training dataset is developed by collecting data from UCI repository consists of 345 instances with 7 different attributes. This paper deals with results in the field of data classification obtained with Naïve Bayes algorithms .FT tree algorithms, and KStar algorithms and on the whole performance made know FT Tree algorithm when tested on liver disease datasets, time taken to run the data for result is fast when compare to other algorithm with accuracy of 97.10%Based on the experimental results the classification accuracy is found to be better using FT Tree algorithm compare to other algorithms [2].</p>
---	--	------	---	--

3	Liver Disease Prediction System using Machine Learning Techniques	2021	<p>International Journal of Engineering Research & Technology (IJERT)</p> <p>Vol. 10 Issue 06, June-2021</p>	<p>Researchers are attempting to discover models for early diagnosis of illness utilizing biomedical information. Since the most recent couple of decades, they have utilized a parcel of models for early finding, each with their very own advantages and disadvantages. In this research, a CHIRP based model is proposed for the early forecast of liver disease.</p>
4	Liver Disease Prediction Using Machine Learning Algorithm	2021	<p>Authors:</p> <p>Sambit Mohanty, Pradosh Kumar Gantayat, Sachikanta Dash, Bhabani P. Mishra & Shiba Ch. Barik</p> <p>Conference paper</p> <p>First Online: 05 May 2021</p> <p>Part of the Advances in Intelligent Systems and Computing book series (AISC, volume 1407)</p>	<p>We can use these data to improve our healthcare services or proper identification of diseases. We have collected patient data from open source platform and applied various kinds of data analysis techniques and machine learning (ML) approaches applied to see the pattern of the</p>

				data sets. Then, a performance comparison between these models is made to get highly accurate model for predicting liver disease.
5	Liver disease prediction using machine learning and deep learning: A comparative study	2022	<ul style="list-style-type: none"> • March 2022 • <u>Intelligent Decision Technologies</u> 16(3):1-14 <p>DOI:10.3233/IDT-210065</p> <p>Authors:</p> <p>Bhawna Singla</p> <p>Soham Taneja</p> <p>Rishika Garg</p> <p>Preeti Nagrath</p>	This research aims to impart insight additional to the current state-of-the-art discoveries by focusing on a comparative analysis of some of the best ML/DL techniques which haven't been scrutinized altogether yet.

2.2 Problem Statement Definition

Chronic liver diseases include chronic hepatitis, fibrosis, and cirrhosis. Hepatitis can occur from viral infection (e.g., hepatitis c virus) or auto-immune origin. When liver disease is diagnosed at an earlier stage, in between infection and fibrosis but before cirrhosis, liver failure can be avoided. The use of artificial intelligence and machine learning in combination with clinical findings has further improved disease detection. Statistical machine learning algorithms based on specific problems can assist one to make decisions.

3.IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to help teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.2 Ideation , Brainstorming & Prioritize

Brainstorming



Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

🕒 10 minutes



Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.



Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.



Learn how to use the Ideation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) ➔



Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes

PROBLEM

Patients who suffer from liver disease need a way to prevent it from reaching the chronic stage by taking necessary treatment.



Key rules of brainstorming

To run a smooth and productive session



Stay in topic.



Encourage wild ideas.



Defer judgment.



Listen to others.



Go for volume.



If possible, be visual.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

TIP

You can select a sticky note and hit the pencil (switch to sketch) icon to start drawing!

Srija U

User Friendly	Easy Access and Easy Availability for the user	Expert Supervision to build and to review the app
All types of symptoms should be given	Location of the user should be accessible	How to highlight the symptoms and diseases to the user
Patients' personal health records should be stored	Emergency has all medical devices and medicines to detect medical symptoms	Report of the patients should be displayed

Sai Madhuree

Give patients health information as input	Compare the best previous report threshold score	To be the help from hospitals
Addition like smoking and drinking should be considered	The final report is given to the customers	It can be built in an app with healthcare institutions
At the end, customer has some extra work to be done to be healthy for output	Step of patient's health report can be taken as input	The final output report should be understandable

Saayi Shree

The doctor needs to be available at all times	Answers about everything	Update of the app frequently
Can be designed as a website or an app	Cost Free report	Doctor will also be able to view the patient's final report
Recommendation of hospitals and emergency	Easy to Operate by anyone	Report can be generated in any language

Sri Sarini U

It should predict all types of other disease	Age of the patient plays a vital role	The disease can be either curable or critical
Should have back-up to support the report generated	The algorithm used should be powerful enough to generate accurate results	Conclude Doctor that this is a treatable condition
App should be scalable for multiple users	Generated Report should be downloadable and readable	Site should be maintained and monitored for every user

3.3 Proposed Solution

Project Design Phase-I
Proposed Solution Template

Date	26 September 2022
Team ID	PNT2022TMD53422
Project Name	Statistical Machine Learning Approaches to Liver Disease Prediction
Maximum Marks	2 Marks

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	To detect disease, healthcare professionals need to collect samples from patients which can cost both time and money. Often, more than one kind of test or many samples are needed from the patient to accumulate all the necessary information for a better diagnosis. There is a need to find better ways to detect and diagnose liver disease with more accuracy.
2.	Idea / Solution description	Statistical machine learning algorithms based on specific problems can assist one to make decisions. Machine learning (ML), data-driven algorithms can be utilized to validate existing methods and help researchers to make potential new decisions.
3.	Novelty / Uniqueness	Various kinds of data sets, such as blood panels with liver function tests, histologically stained slide images, and the presence of specific molecular markers in blood or tissue samples, have been used to train classifier algorithms to predict liver disease with good accuracy.
4.	Social Impact / Customer Satisfaction	Application of the ML methods can help reduce the total burden of liver disease on public health worldwide by improving recognition of risk factors and diagnostic variables. More importantly, for chronic liver disease, detecting liver disease at earlier stages or in hidden cases by ML.

		could decrease liver-related mortality, transplants, and/or hospitalizations.
5.	Business Model (Revenue Model)	The global liver disease diagnostics market size was valued at USD 29.3 billion in 2019 and is estimated to grow at a compound annual growth rate (CAGR) of 6.5% from 2020 to 2027. Rising prevalence of acute and chronic liver diseases is one of the major factors expected to drive the market for liver disease diagnostics.
6.	Scalability of the Solution	The described ML methods can assist health sectors to achieve a better diagnosis providing effective results in identifying groups or levels within medical data to facilitate healthcare workers. The machine learning algorithms presented in this study can support medical experts but are not the alternative when making decisions from ML classifiers for diagnostic pathways.

3.4 Problem Solution fit

Project Title: Statistical Machine Learning Approaches to Liver Disease Prediction		Project Design Phase-I - Solution Fit Template		Team ID: PNT2022TMD 53422	
Define CS, fit into CC	1. CUSTOMER SEGMENT(S) <small>Who are your customers? (or existing players of 3-5 yrs old)</small> <p>Medical Sector Patients Common People</p>	6. CUSTOMER CONSTRAINTS <small>What constraints prevent your customers from using your solution or limit their interest in solutions? (i.e. spending priority, budget, access, cultural, convenience, available, etc.)</small> <p>Security threat Network Connection Reliability</p>	5. AVAILABLE SOLUTIONS <small>What solutions are available in the customer's environment? Have they tried them? If not, why not? What have they tried in the past? What products? What do these solutions have? (i.e. pros and cons) Is an alternative to digital or otherwise?</small> <p>Liver function test has to be performed manually Pros: Expert knowledge Cons: Time Consuming</p>	Explore AS, differentiate	
	2. JOBS-TO-BE-DONE / PROBLEMS <small>What jobs to be done (or problems) do you address for your customer? There could be more than one, explore different value</small> <p>The objective is to predict liver disease. We generate a health report based on the input provided by the user. The input may include symptoms, basic information regarding user's current health.</p>	9. PROBLEM ROOT CAUSE <small>What is the root cause that the problem exists? What is the root cause behind the need to do this job? (i.e. Customer's need to do it because of the change in requirements)</small> <p>Drug usage Genetic Consumption of alcohol Unhealthy lifestyle</p>	7. BEHAVIOUR <small>What does your customer do to address the problem and get the job done? Identify several that the right solution could involve, estimate usage and benefits. Identify the associated customer spend/fee into an alternative mode (i.e. alternatives)</small> <p>Analyze the symptoms Enter the input Get results</p>		
Focus on CS, fit into BE, understand BC	3. TRIGGERS <small>What triggers customers to act? (or seeing their neighbours installing, word passing, hearing about product/feature solution online news)</small> <p>When the user has symptoms of liver disease or he wants to know if he is diagnosed with liver disease</p>	10. YOUR SOLUTION <small>If you are building on an existing business, write down your current solution that offers the service, and think how much it has really. If you are building on a new business proposition, the way it looks different to the service, and how it will be different from the existing business. Address a problem and customer customer behaviour.</small> <p>Our current solution uses machine learning algorithms to analyze the user's symptoms and predict liver disease.</p>	8. CHANNELS OF BEHAVIOUR <small>8.1 ONLINE What kind of solution do customers use online? (Online sales channels, etc.) 8.2 OFFLINE What kind of solution do customers use offline? (Offline sales channels, etc.) and use these for customer segmentation</small> <p>Online: The application analyzes the symptoms of the user. Offline: Consult doctors and experts.</p>	Focus on BE, fit into BC, understand BC	
	4. EMOTIONS: BEFORE / AFTER <small>How do customers feel when they have experience on a job and afterwards? (i.e. feel, experience, confidence, in control, ease of use, customer satisfaction, etc.)</small> <p>Confused, curious, scared >>Clear, updated</p>				

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail Registration through Healthcare Portals
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Application	Filling of application Modification of application Verification of application
FR-4	User Verification	Verifying through Credentials through Database
FR-5	Database Update	1.Updating Data in Database 2.Altering data in Database 3.Deleting Data in Database
FR-6	Activities Tracking	Track the activities and keep the database updated

4.2 Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The prediction model can be used by users, doctors and health care institutions
NFR-2	Security	The prediction model is not vulnerable to any brute force attack or any security attacks.
NFR-3	Reliability	The prediction model can only be accessed by the users who possess the username and password.

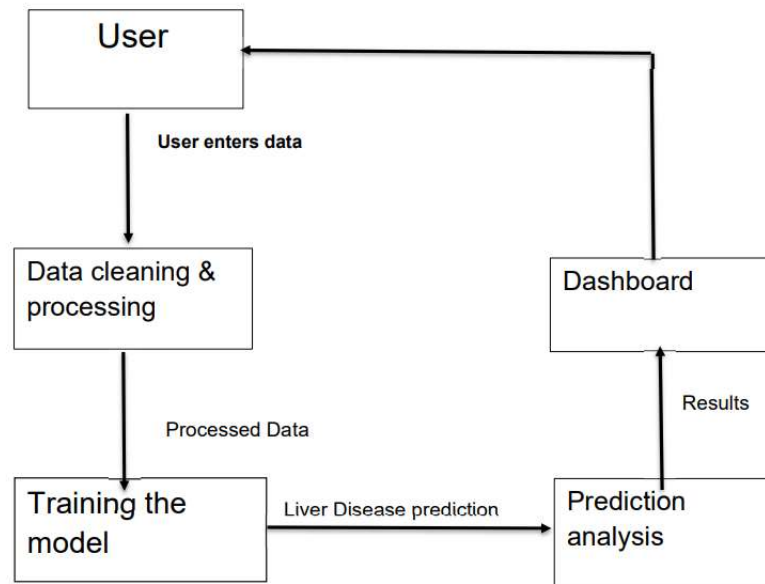
NFR-4	Performance	The prediction model can take in huge amounts of data , scalable for large-scale users and can do prediction as well as visualization.
-------	--------------------	--

NFR-5	Availability	The prediction model can be accessed at anytime and anywhere.
NFR-6	Scalability	The prediction model can be used by multiple users and multiple hospitals at the same time

5.PROJECT DESIGN

5.1 Data Flow Diagrams

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



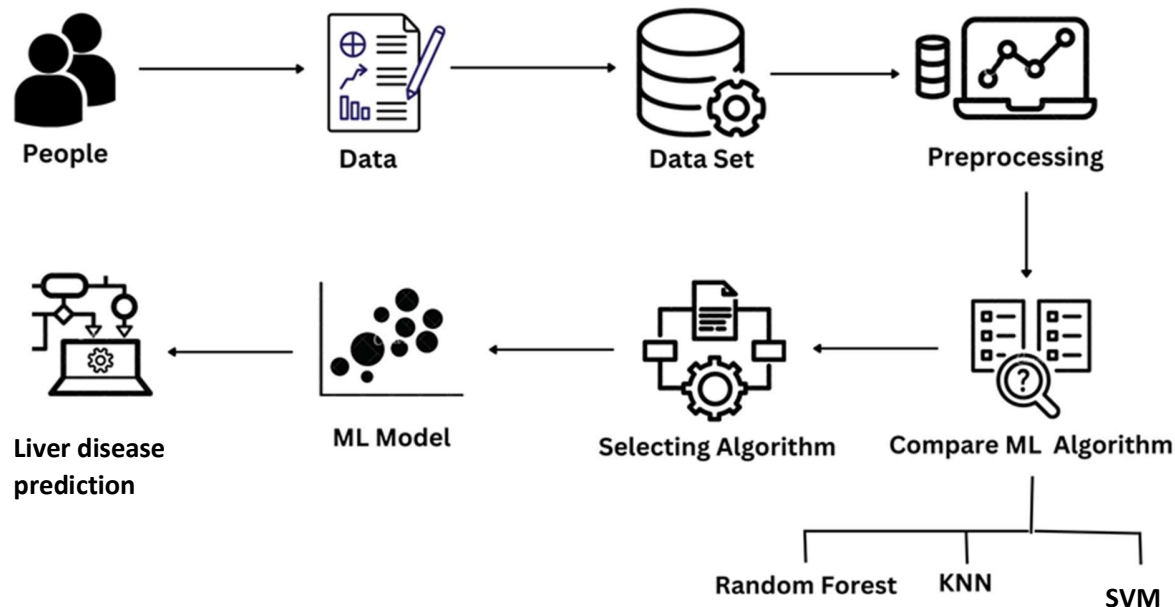
5.2 Solution & Technical Architecture

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, behaviour, and other aspects of the software to project stakeholders
- Define features, development phases, and solution requirements
- Provide specifications according to which the solution is defined, managed, and delivered.

SOLUTION ARCHITECTURE



5.3 User Stories

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 password.	5	High	Srija
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	5	High	Sri Sarini
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	10	Medium	Sai Madhuree
Sprint-2	Input Necessary Details	USN-4	As a user, I can give input test details to predict liver disease	15	High	Saayi Shree
Sprint-2	Data Pre-Processing	USN-5	Transform raw data into appropriate format for prediction	5	High	Srija
Sprint-3	Prediction of Liver Disease	USN-6	As a user I can get the results of liver disease prediction	15	High	Sri Sarini
Sprint-3		USN-7	As a user I can get accurate results of liver disease	5	Medium	Sai Madhuree
Sprint-4	Review	USN-8	As an admin I reinforce the result of prediction	20	High	Saayi Shree

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and 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 password.	5	High	Srija
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	5	High	Sri Sarini
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	10	Medium	Sai Madhuree
Sprint-2	Input Necessary Details	USN-4	As a user, I can give input test details to predict liver disease	15	High	Saayi Shree
Sprint-2	Data Pre-Processing	USN-5	Transform raw data into appropriate format for prediction	5	High	Srija
Sprint-3	Prediction of Liver Disease	USN-6	As a user I can get the results of liver disease prediction	15	High	Sri Sarini
Sprint-3		USN-7	As a user I can get accurate results of liver disease	5	Medium	Sai Madhuree
Sprint-4	Review	USN-8	As an admin I reinforce the result of prediction	20	High	Saayi Shree

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022		

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$

7. CODING & SOLUTIONING

7.1 Feature 1

127.0.0.1:5000/Index

Liver Disease Prediction

Age :	Gender :
<input type="text" value="62"/>	<input type="text" value="1"/>
Total Bilirubin :	Direct Bilirubin :
<input type="text" value="7.3"/>	<input type="text" value="4.1"/>
Alkaline Phosphatase:	Alamine Aminotransferase:
<input type="text" value="490"/>	<input type="text" value="60"/>
Aspartate Aminotransferase:	Total Protiens:
<input type="text" value="68"/>	<input type="text" value="7"/>
Albumin:	Albumin and Globulin Ratio:
<input type="text" value="3.3"/>	<input type="text" value="0.89"/>

[Predict](#)

Type here to search

7.2 Feature 2

```
File Edit Selection View Go Run Terminal Help
app.py - liver_disease_prediction - Visual Studio Code

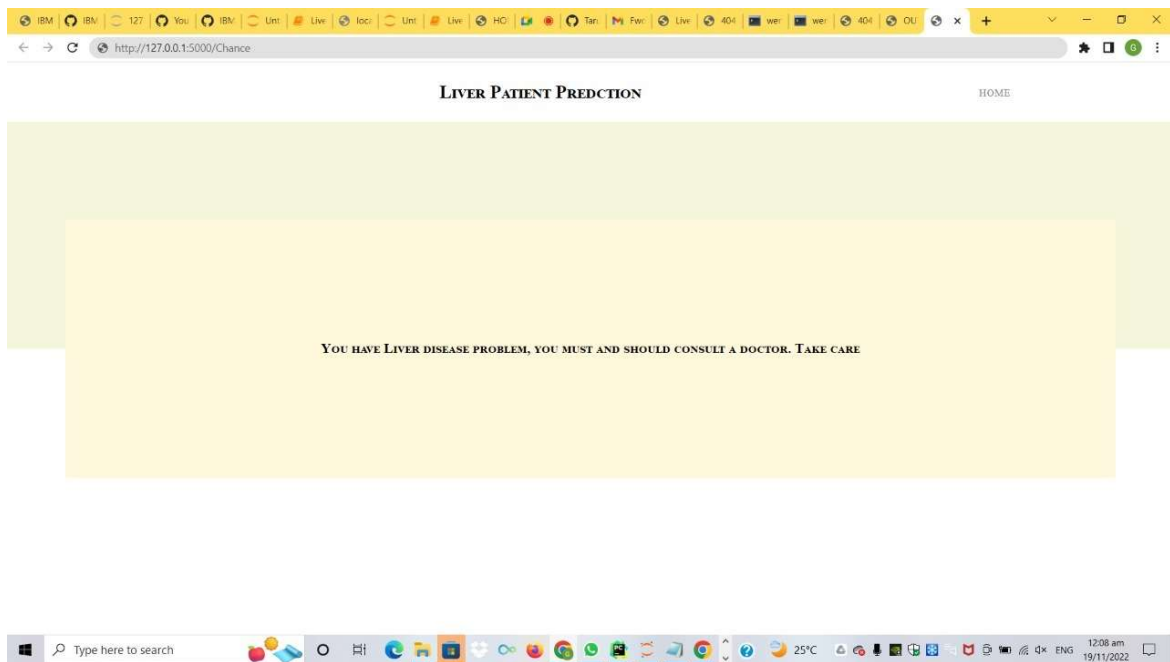
EXPLORER
  LIVER_DISEASE_PREDICTION
    .idea
    .gitignore
    liver_disease_prediction.iml
    misc.xml
    modules.xml
    workspace.xml
    .vscode
    launch.json
  sample_proj-master
    dataset
    indian_liver_patient.csv
  Flask_app
    __pycache__
    static
    templates
      Chance.html
      home.html
      Index.html
      noChance.html
    app.py
    liver_disease.ipynb
    liver_analysis.pkl
  .gitignore
  indian_liver_patient.csv
  liver_disease.ipynb
  liver_analysis.pkl

OUTLINE
  app.py
  liver_disease.ipynb
  liver_analysis.pkl

sample_proj-master > Flask_app > app.py > ...
1 import os, path
2 from flask import Flask, render_template, request
3 import pickle
4 STATIC_DIR=os.path.abspath('../static')
5
6 app = Flask(__name__, template_folder='templates', static_folder=STATIC_DIR)
7
8
9 # @app.route('/')
10 # def home():
11
12 # #return "<h1> HELLO </h1>"
13 # return render_template('home.html')
14
15 @app.route('/', methods=['GET'])
16 def home():
17     return render_template('home.html')
18
19
20 @app.route('/Index')
21 def Index():
22     return render_template("Index.html")
23
24
25 @app.route('/data_predict', methods=['POST', 'GET'])
26 def data_predict():
27     age = request.form['Age']
28     gender = request.form['Gender']
29     tb = request.form['Total_Bilirubin']
30     db = request.form['Direct_Bilirubin']
31     ap = request.form['Alkaline_Phosphotase']
32     aa1 = request.form['Alamine_Aminotransferase']
33     aa2 = request.form['Aspartate_Aminotransferase']
34     tp = request.form['Total_Protiens']
35     a = request.form['Albumin']
36     agr = request.form['Albumin_and_Globulin_Ratio']
37
38     data = [(float(age), float(gender), float(tb), float(db), float(ap), float(aa1), float(aa2), float(tp), float(a), float(agr)))]
```

The screenshot shows the Visual Studio Code editor with the file `app.py` open. The file contains a Flask application for liver disease prediction. The code includes a route `/data_predict` that accepts POST and GET requests. It extracts various health parameters from the request form, such as Age, Gender, Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase, Alanine Aminotransferase, Aspartate Aminotransferase, Total Proteins, Albumin, and Albumin and Globulin Ratio. These parameters are converted to floats and passed to a prediction model loaded from `liver_analysis.pkl`. The model's prediction is then used to render either `Chance.html` or `noChance.html` based on the result. The application is run in debug mode.

```
23
24
25 @app.route('/data_predict', methods=['POST', 'GET'])
26 def data_predict():
27     age = request.form['Age']
28     gender = request.form['Gender']
29     tb = request.form['Total Bilirubin']
30     db = request.form['Direct Bilirubin']
31     ap = request.form['Alkaline Phosphatase']
32     aal = request.form['Alanine Aminotransferase']
33     aa2 = request.form['Aspartate Aminotransferase']
34     tp = request.form['Total Proteins']
35     a = request.form['Albumin']
36     agr = request.form['Albumin_and_Globulin_Ratio']
37
38     data = [[float(age), float(gender), float(tb), float(db), float(ap), float(aal), float(aa2), float(tp), float(a),
39               float(agr))]]
40
41     model = pickle.load(open('liver_analysis.pkl', 'rb'))
42     prediction = model.predict(data)[0]
43
44     if prediction == 1:
45         return render_template("Chance.html")
46     else:
47         return render_template("noChance.html")
48
49 if __name__ == "__main__":
50     app.run(debug=True)
```



8.TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
tc01	Functional	Page	Verify user is able to click on Predict button		1.Enter URL and fill the form 2.Click on Predict button		Loan form should display	Working as expected	Pass				
tc02	Functional	Home Page	The web page is getting refreshed		1.Automatic page reload		Loan form must appear automatically after page reload	Working as expected	Fail	No steps needed	Y	BUG-1234	
tc03	Functional	Home page	Field address validation		1. Double-click on the E-mail address field		User should navigate to E-mail address field	Working as expected	Pass				
tc04	Functional	Output page	Loan Credibility predicted output		1. Click on predict button 2. View the predicted results		User should access the Loan credibility predicted result	Working as expected	Pass				

8.2 User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
------------	------------	------------	------------	------------	----------

By Design	9	3	1	2	15
Duplicate	0	0	4	0	4
External	1	2	0	0	3
Fixed	10	5	4	21	40
Not Reproduced	0	0	0	0	0
Skipped	0	0	0	0	0
Won't Fix	0	0	0	0	0
Totals	20	9	9	23	64

1. 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	5	0	0	5
Client Application	46	0	0	46
Security	4	0	0	4
Outsource Shipping	3	0	0	3
Exception Reporting	7	0	0	7
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

9.1 Performance Metrics

Model Performance Testing:

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

S.No.	Parameter	Screenshot / Values
1.	Data Responsiveness	The ML model takes about 0.3 seconds to process the dataset. The credibility result is predicted in approximately 0.9 seconds.
2.	Utilisation of Data Filters	Sufficient data filters have been used for ideal model building
3.	Effective User Story	No of Scene Added - 15
4.	Descriptive Reports	No of Visualisations / Graphs - 13

10. ADVANTAGES & DISADVANTAGES

Advantage:

Anyone can use the application.

Our application can be used by anyone with minimum knowledge. The Usability of the App is one of the advantage.

User Friendly UI

User Friendly User Interface enables the users to easily understand how the application works. Therefore, Encouraging users to use the application whenever they want.

Remote Access

At home diagnosis in case of emergencies as the results are much easier and faster to get. Simple diagnosis can be done at home without the need to go to the hospital.

Disadvantage:

Does not replace hospitals

The user cannot completely rely on the application and this application cannot replace hospitals or prevent the users to go to the hospitals .It is just an emergency or a support application for the users.

Prediction may not be accurate

The user cannot completely trust the diagnosis done through the application. The user may need to double check by visiting a doctor. But this may not happen everytime. This is advised just to be on a safer side.

11. CONCLUSION

The analysis starts from data cleaning and processing missing value, exploratory analysis and finally model building and evaluation of the model. The best accuracy on public test set is when we get higher accuracy score and other performance metrics which will be found out. This project can help to predict the liver disease of the user.

12. FUTURE SCOPE

The health or life of the user is relied on this application therefore the accuracy can be improved in order to completely trust the application. This application can also help healthcare workers and can serve as a support system.

13. APPENDIX

Source Code

app.py

```
import os.path

from flask import Flask, render_template, request
import pickle
STATIC_DIR=os.path.abspath('./static')
```

```
app = Flask(__name__, template_folder='templates',static_folder=STATIC_DIR)
```

```
# @app.route('/')  
# def home():
```

```
#     #return "<h1> HELLO </h1>"  
#     return render_template('home.html')
```

```
@app.route('/',methods=["GET"])  
def home():  
    return render_template('home.html')
```

```
@app.route('/Index')  
def Index():  
    return render_template("Index.html")
```

```
@app.route('/data_predict', methods=['POST','GET'])  
def data_predict():  
    age = request.form['Age']  
    gender = request.form['Gender']  
    tb = request.form['Total_Bilirubin']  
    db = request.form['Direct_Bilirubin']  
    ap = request.form['Alkaline_Phosphotase']  
    aa1 = request.form['Alamine_Aminotransferase']  
    aa2 = request.form['Aspartate_Aminotransferase']  
    tp = request.form['Total_Protiens']  
    a = request.form['Albumin']  
    agr = request.form['Albumin_and_Globulin_Ratio']  
  
    data = [[float(age), float(gender), float(tb), float(db), float(ap), float(aa1),  
             float(aa2), float(tp), float(a),  
             float(agr)]]  
  
    model = pickle.load(open('liver_analysis.pkl', 'rb'))  
    prediction = model.predict(data)[0]  
  
    if prediction == 1:  
        return render_template("Chance.html")  
    else:  
        return render_template("noChance.html")
```



```
        placeholder="Albumin and Globulin Ratio"
name="Albumin_and_Globulin_Ratio" style="width: 190px ; margin-left:
100px;">
        <br>
        <br>
        <br>
        <!-- <button type="submit" style="width: 100px ; margin-left:
200px;">Predict</button> -->
        <a href="/data_predict">Predict</a>

</form>
```

13.2 GitHub& Project:

<https://github.com/IBM-EPBL/IBM-Project-18196-1659680664/upload/main/Project%20Development%20Phase>