Project Report

Revolutionizing Liver Care: Predicting Liver Cirrhosis using Advanced Machine Learning Techniques

1. INTRODUCTION

1.1 Project Overview

Liver cirrhosis is a chronic and progressive liver disease characterized by the irreversible scarring of liver tissue, which can lead to severe complications and liver failure if left untreated. This project aims to develop a predictive model for the early detection and prognosis of liver cirrhosis using machine learning techniques. The developed predictive model holds promise for early detection and prognosis of liver cirrhosis, enabling healthcare professionals to initiate timely interventions and personalized treatment strategies. By accurately identifying individuals at risk of cirrhosis, this research contributes to improved patient outcomes and the optimization of healthcare resources. Moreover, it highlights the potential of machine learning in the field of hepatology, showcasing its ability to leverage clinical data for predictive modeling and decision support.

1.2 Purpose

The purpose of this project is to develop an intelligent and non-invasive diagnostic tool that can accurately predict the presence of liver cirrhosis using clinical and behavioral data. By leveraging machine learning algorithms, the project aims to assist healthcare professionals in early detection, enabling timely treatment, reducing mortality, and improving patient outcomes.

2. IDEATION PHASE

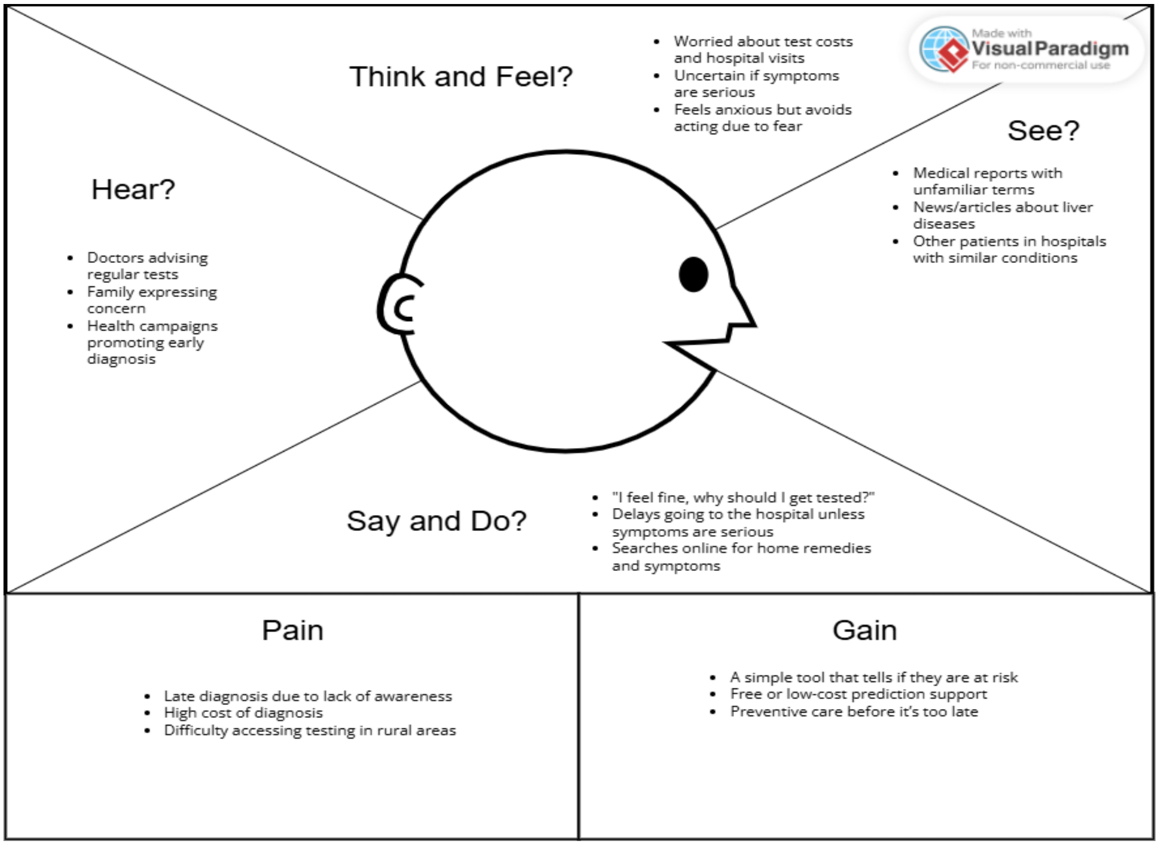
2.1 Problem Statement

Liver cirrhosis is a progressive and potentially fatal condition that often goes undiagnosed until advanced stages due to its subtle early symptoms. Traditional diagnostic methods can be time-consuming, costly, and dependent on expert interpretation. There is a pressing need for a fast, accurate, and accessible system that can assist healthcare professionals in early detection using available clinical data. This project aims to address that gap by leveraging machine learning techniques to build a predictive model that supports early diagnosis and improves patient outcomes.

Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS-1	a Patient	Understand if I'm at risk of cirrhosis	I don't have access to early testing	Diagnostic tools are costly and limited	Anxious and helpless
PS-2	a doctor in a rural area	detect cirrhosis early in my patients	patients come only in late stages	early symptoms are hard to detect without advanced tools	frustrated and concerned

2.2 Empathy Map Canvas

The Empathy Map Canvas helps identify and understand the needs, goals, and pain points of the primary users of the Liver Cirrhosis Prediction System—doctors and healthcare professionals. By mapping what users see, say, do, and hear, as well as their challenges and expected benefits, we ensure the solution is user-centered and practical in real-world clinical settings. This approach supports the development of a tool that not only functions accurately but is also intuitive and valuable to its end users.



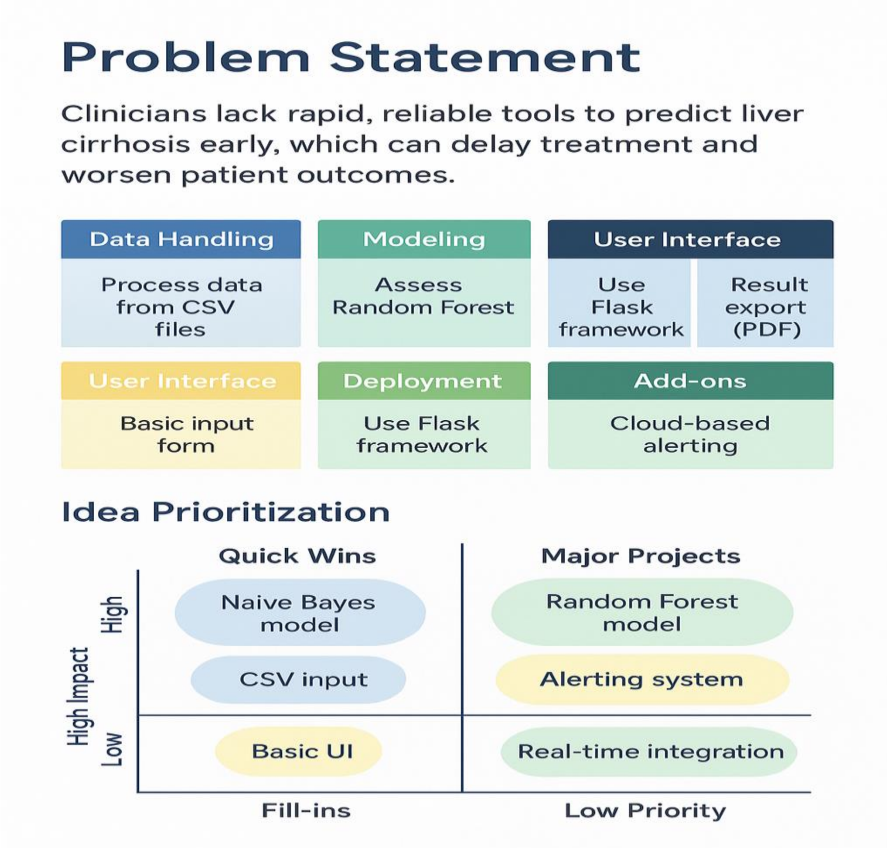
2.3 Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a 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 amount of creative solutions.

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

Step-2: Brainstorm, Idea Listing and Grouping

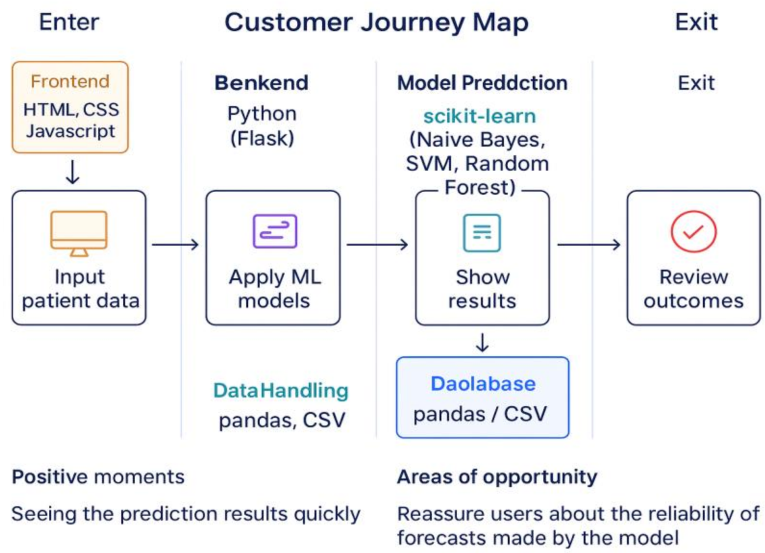
Step-3: Idea Prioritization



3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

The Customer Journey Map visually represents the step-by-step experience of a user interacting with the liver cirrhosis prediction system. It highlights key stages—such as scheduling, data entry, model prediction, and receiving results—while identifying positive experiences, pain points, and opportunities for improvement. This helps developers and stakeholders empathize with the user and refine the solution for better usability and satisfaction.



3.2 Solution Requirement

The system requires clinical data as input and uses machine learning algorithms (Naive Bayes, SVM, Random Forest) to predict liver cirrhosis. It is developed using Python, with tools like scikit-learn and Flask for model training and deployment. Key requirements include accurate data preprocessing, efficient model execution, and a simple web interface for user interaction.

Functional Requirements:

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	Data Input Interface	Allow user to input clinical values (e.g., Hemoglobin, Bilirubin, Platelet Count, etc.) Form validation and formatting
FR-2	Model Prediction	Use trained ML model to predict liver cirrhosis status Return prediction output as “Positive” or “Negative”
FR-3	Result Report Generation	Downloadable prediction report Include charts or risk level indicator
FR-4	Prediction Output	Display predicted liver cirrhosis status in the console/output cell

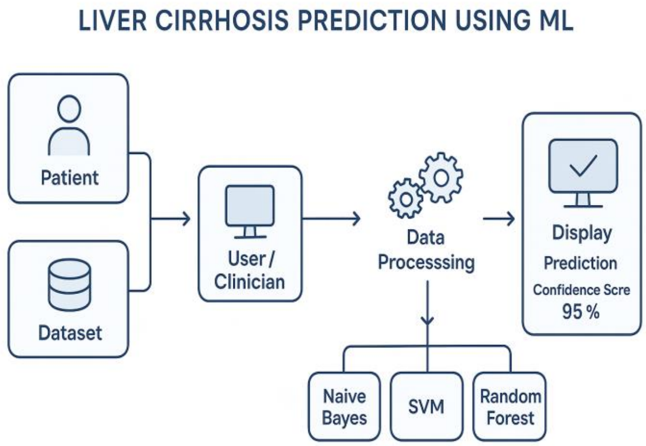
FR-5	Code Availability	Source code and sample dataset are hosted on GitHub for reproducibility

Non-functional Requirements:

FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	The interface should be simple, intuitive, and user-friendly for both doctors and non-technical users.
NFR-2	**Security**	All data inputs and predictions must be securely processed; user credentials and health data should be encrypted
NFR-3	**Reliability**	The model should consistently deliver correct results based on trained data without unexpected failures.
NFR-4	**Performance**	Predictions must be generated within 2–3 seconds of data input.
NFR-5	**Availability**	The system should be accessible 24/7 with minimal downtime.
NFR-6	**Scalability**	Should support a growing number of users and accommodate additional disease prediction modules in future.

3.3 Data Flow Diagram

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.The below figure represents the a streamlined workflow of how machine learning is applied in a clinical setting to predict liver cirrhosis. It begins with patient data and a medical dataset, which are accessed by a clinician or user. The data undergoes preprocessing before being passed through machine learning models such as Naive Bayes, SVM, and Random Forest. These models analyze the data and generate a prediction along with a confidence score. The final output is displayed to the user, providing an accurate and efficient decision-support tool for early detection of liver cirrhosis.



User Stories

The functional expectations of the system from the perspective of its users—clinicians and developers. Each user story defines a specific task or goal, ensuring the system meets real-world needs. The stories are categorized by functionality such as data upload, prediction, evaluation, and reporting, with associated priorities and sprint planning. This user-centered approach helps guide development and ensures all critical features are addressed efficiently

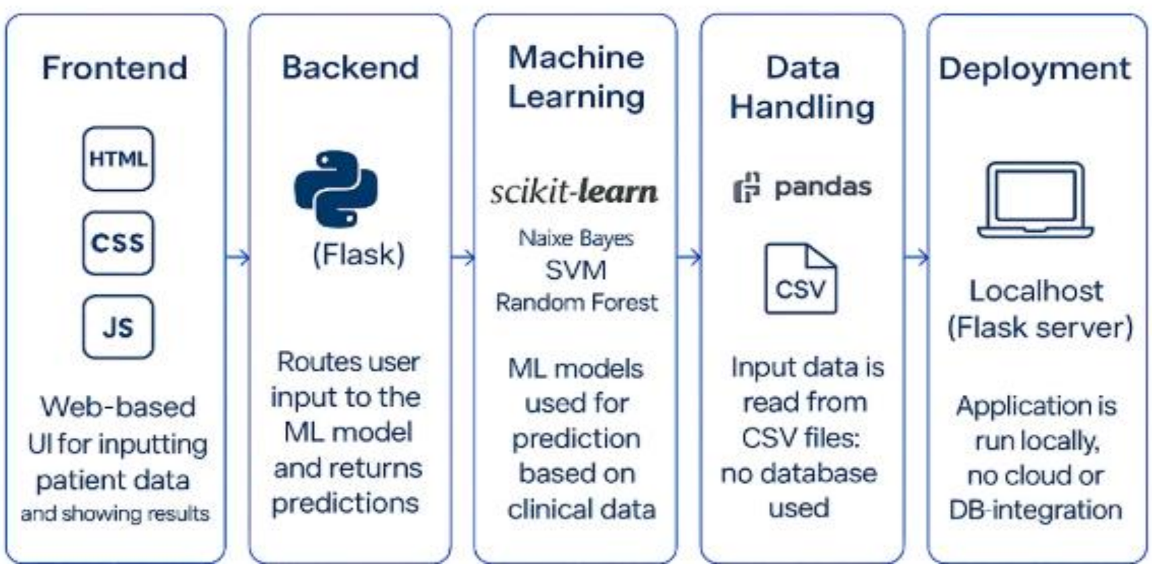
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Clinician/ User	Data Upload	USN-1	As a clinician, I want to upload patient clinical data to predict liver cirrhosis.	The system accepts and displays input data	High	Sprint-1
Clinician/ User	Prediction	USN-2	As a clinician, I want to receive a prediction result based on input data	Model returns prediction (Yes/No) on liver cirrhosis risk	High	Sprint-2
Clinician/ User	Result Display	USN-3	As a clinician, I want to view prediction results and confidence scores.	Accuracy/confidence is shown after prediction	Medium	Sprint-2

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Clinician/ User	Report Download	USN-4	As a clinician, I want to download the result as a report (PDF/DOCX)	Report gets downloaded successfully with prediction details	Medium	Sprint-3
Developer/Team	Model Evaluation	USN-5	As a developer, I want to evaluate ML models using performance metrics.	Metrics like accuracy, recall, precision are generated	High	Sprint-2
Developer/Team	GitHub Documentation	USN-6	As a developer, I want to document and upload the project to GitHub.	GitHub repo contains code, README, and model files	High	Sprint-3

3.4 Technology Stack

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the Components & Technologies & Application Characteristics



Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web interface for clinicians to input data and view predictions	HTML, CSS, JavaScript
2.	Application Logic-1	Data preprocessing and handling missing/categorical values	Python (pandas, NumPy)
3.	Application Logic-2	Model training, prediction, and evaluation	Python (scikit-learn)
4.	Application Logic-3	Flask-based integration to link frontend and backend	Python (Flask)
5.	File Storage	Store model files and datasets locally or in GitHub repo	Local Filesystem
6.	Machine Learning Model	Predicts whether a patient has liver cirrhosis	Naive Bayes, SVM, Random Forest,Logistic regression,
7.	Infrastructure (Server / Cloud)	Application runs locally using Flask; can be extended to cloud	Local (Flask server)

Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks and libraries used for development	Flask, scikit-learn, pandas
2.	Security Implementations	Application is local; data privacy ensured through offline processing	Local-only use; no cloud auth
3.	Scalable Architecture	Can be scaled using microservices or hosted on cloud in future	Flask + Modular Python files
4.	Performance	Optimized with preprocessed data and saved models for fast predictions	Flask,pandas

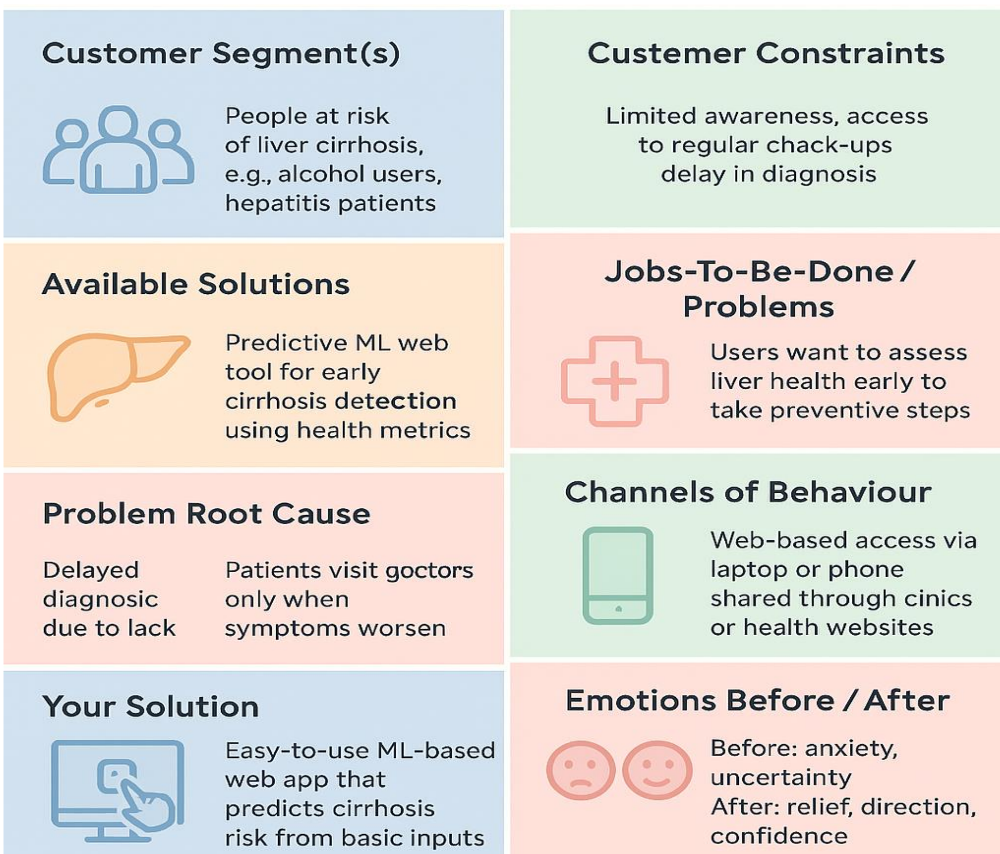
4. PROJECT DESIGN

4.1 Problem Solution Fit

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you 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

Purpose:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.
- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems.
- Understand the existing situation in order to improve it for your target group



4.2 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Early diagnosis of liver cirrhosis is difficult due to vague symptoms and dependency on invasive methods. Delays in detection reduce chances of effective treatment.
2.	Idea / Solution description	Our ML-based solution predicts the presence of liver cirrhosis using clinical parameters like hemoglobin, platelet count, SGOT/AST, bilirubin, and alcohol consumption. It enables early detection through a web-based tool accessible to both doctors and patients.
3.	Novelty / Uniqueness	Uses non-invasive clinical data and multiple machine learning algorithms (Naive Bayes, SVM, KNN, Logistic Regression, Random Forest) to deliver accurate predictions. Custom-tuned models ensure high precision and recall.
4.	Social Impact / Customer Satisfaction	Helps in saving lives by promoting early detection and timely intervention. Reduces cost and time compared to traditional diagnosis. Improves confidence and satisfaction for both patients and healthcare providers.
5.	Business Model (Revenue Model)	Freemium model for individuals, subscription model for hospitals/clinics. Potential integration with diagnostic labs and EMR systems.
6.	Scalability of the Solution	Can be expanded to include predictions for other liver conditions, deployed as a cloud-based API, integrated into hospital portals or mobile apps for wider access.

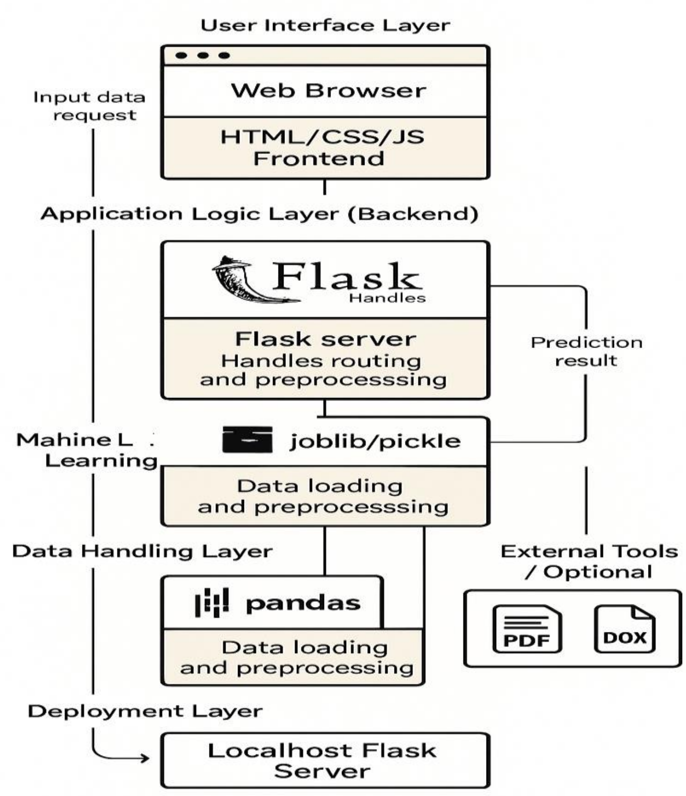
The system will accept patient data and use pre-trained models (Naive Bayes, KNN, SVM, etc.) to classify cirrhosis risk.

4.3 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 project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.

Solution Architecture Diagram:



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

The structured development of the Liver Cirrhosis Prediction System using agile methodology. The work was divided into sprints, with each sprint focusing on specific tasks like data preprocessing, model training, and evaluation. Story points were assigned to each task to estimate effort, and velocity was calculated to track team progress. This planning helped ensure timely, organized, and efficient project execution.

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Collection & Preprocessing	USN-1	As a developer, I want to clean and prepare the dataset for model training	6	High	
Sprint-1	Model Training	USN-2	As a developer, I want to train ML models (Naive Bayes, SVM, RF, etc.)	8	High	
Sprint-1	Exploratory Data Analysis (EDA)	USN-3	As a developer, I want to visualize distributions and correlations in the data	6	Medium	
Sprint-2	Model Evaluation	USN-4	As a developer, I want to evaluate models using accuracy, precision, recall	6	High	
Sprint-2	Result Display	USN-5	As a user, I want to view prediction results and confidence scores	4	Medium	
Sprint-2	Report Generation	USN-6	As a user, I want to download a formatted report (PDF/DOCX) of the results	5	Medium	
Sprint-3	GitHub Hosting & Documentation	USN-7	As a team, we want to upload code and create GitHub documentation	4	High	
Sprint-3	Final Testing & Demo Preparation	USN-8	As a team, we want to test the full pipeline and prepare for final demo	7	High	

Project Tracker, Velocity & Burndown Chart:

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	10 Days	19 May 2025	28 May 2025	20	28 May 2025
Sprint-2	15	10 Days	29 May 2025	7 June 2025	14	7 June 2025
Sprint-3	11	10 Days	9 June 2025	18 June 2025	11	18 June 2025

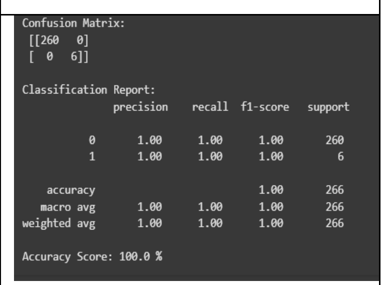
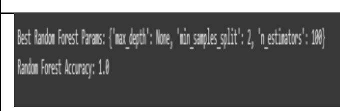
Velocity:

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

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

The models were evaluated using accuracy, Confusion Matrix, Classification Report. Random Forest achieved the best performance.

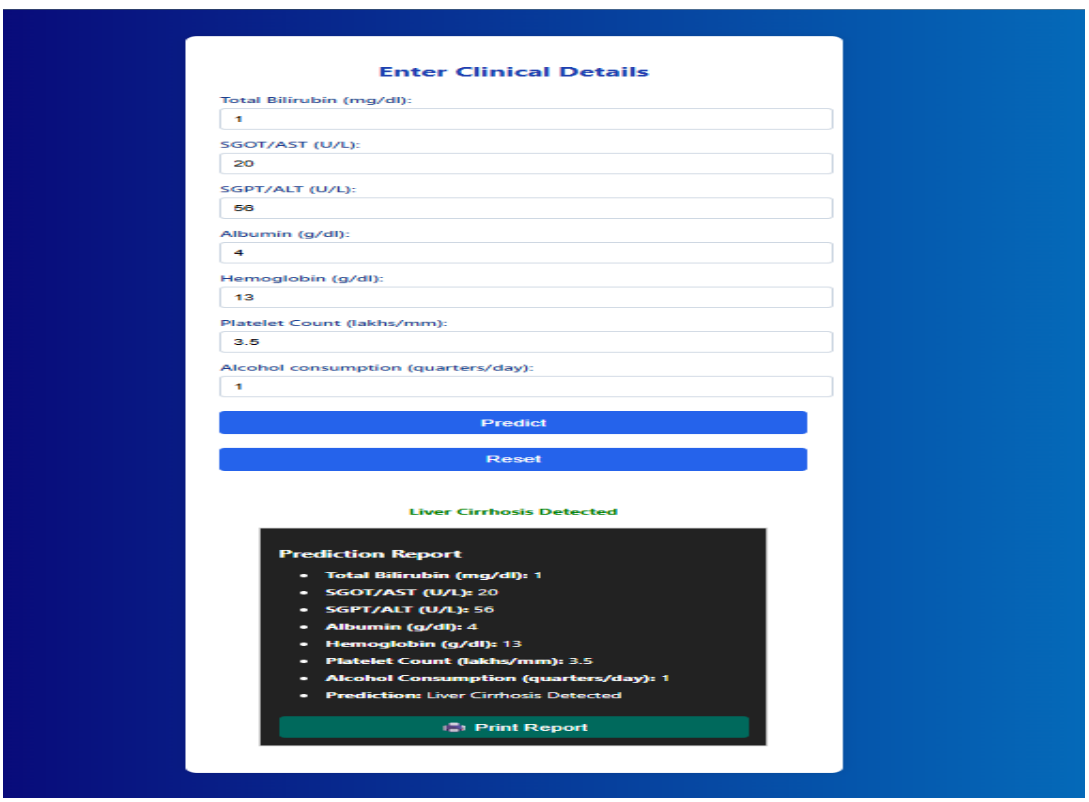
S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - , MSE - , RMSE - , R2 score - **Classification Model:** Confusion Matrix - , Accuray Score- & Classification Report -	
2.	Tune the Model	Hyperparameter Tuning - Validation Method -	

7. RESULTS

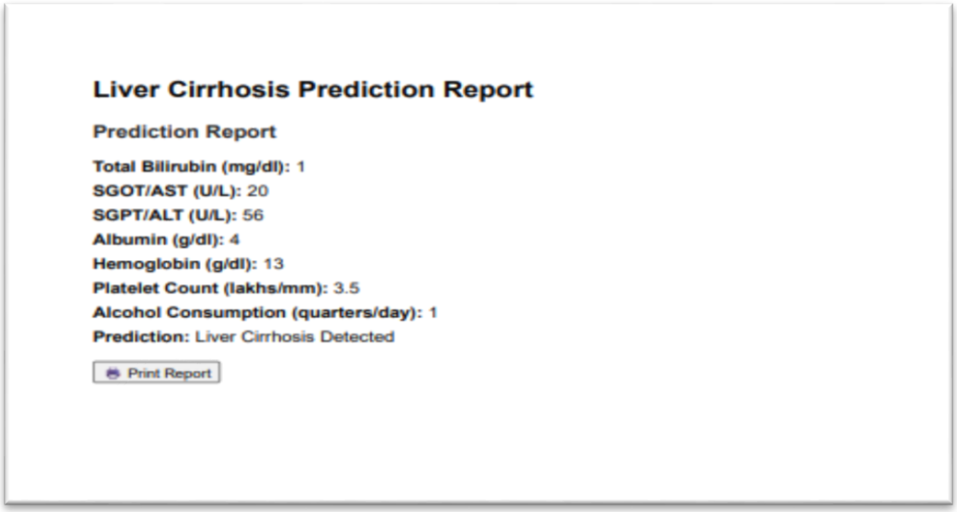
7.1 Output Screenshots

This section presents visual evidence of the working prototype and results obtained from the Liver Cirrhosis Prediction System. The screenshots illustrate key components and outcomes at different stages of the system, from data input to final prediction.

Web Interface Form:



Prediction Report:



Accuracy of Models:

```
print("Accuracy of Naive Bayes:", accuracy_score(y_test, NB_pred))
print("Accuracy of KNN:", accuracy_score(y_test, KNN_pred))
print("Accuracy of SVM:", accuracy_score(y_test, SVM_pred))
print("Accuracy of Logistic Regression:", accuracy_score(y_test, LR_pred))
print("Accuracy of Random Forest:", accuracy_score(y_test, RF_pred))
```

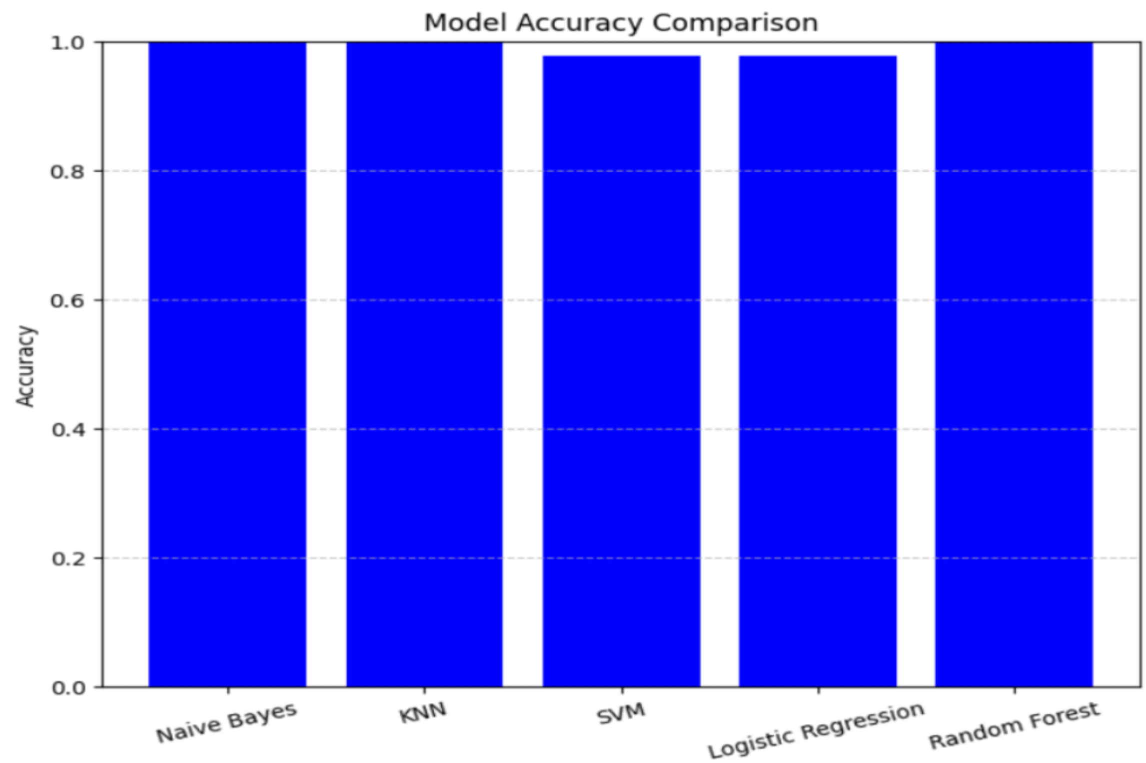
```
Accuracy of Naive Bayes: 1.0
Accuracy of KNN: 1.0
Accuracy of SVM: 0.9774436090225563
Accuracy of Logistic Regression: 0.9774436090225563
Accuracy of Random Forest: 1.0
```

Model Evaluation Metrics Output:

```
# Evaluate
conf_matrix = confusion_matrix(y_test, RF_pred)
class_report = classification_report(y_test, RF_pred)
accuracy = accuracy_score(y_test, RF_pred)

# Display results
print("Confusion Matrix:\n", conf_matrix)
print("\nClassification Report:\n", class_report)
print("Accuracy Score:", round(accuracy * 100, 2), "%")
```

```
Confusion Matrix:
 [[260   0]
 [  0   6]]

Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00       260
           1       1.00      1.00      1.00         6

    accuracy                           1.00       266
   macro avg       1.00      1.00      1.00       266
weighted avg       1.00      1.00      1.00       266

Accuracy Score: 100.0 %
```

Model Comparison:



8. ADVANTAGES & DISADVANTAGES

Advantages:

1. Early Detection: Predicts liver cirrhosis at an early stage using routine clinical data.
2. Non-Invasive: Avoids costly and invasive procedures like liver biopsy.
3. Fast and Efficient: Provides quick results, aiding timely medical decisions.
4. Cost-Effective: Reduces diagnostic costs by using readily available blood test values.
5. Scalable: Can be extended to other liver conditions or integrated into larger healthcare systems.
6. Multiple Algorithm Support: Compares performance across models (e.g., SVM, Random Forest, Naive Bayes) for better accuracy.

Disadvantages:

1. Not a Substitute for Doctor's Diagnosis: Should support, not replace, expert medical evaluation.
2. Requires Quality Data: Model accuracy depends on the dataset quality and feature availability.
3. Limited Interpretability (for some models): Algorithms like Random Forest or SVM may act as "black boxes" to non-technical users.
4. No Real-Time Monitoring: Currently does not support continuous or live patient monitoring.

9. CONCLUSION

This project successfully demonstrates the potential of machine learning algorithms in predicting liver cirrhosis using clinical data. By applying and comparing various models such as Naive Bayes, SVM, Random Forest, and others, we were able to identify the most accurate approach for early diagnosis support. The results highlight how data-driven solutions can assist healthcare professionals in making faster and more reliable decisions.

The system offers a user-friendly interface where clinicians or users can input relevant health parameters and obtain real-time predictions. While the current version focuses on classification and performance evaluation, the model lays a strong foundation for future expansion, including deeper clinical integration and advanced predictive analytics.

Ultimately, this project contributes toward building intelligent, accessible, and scalable tools for liver disease detection—helping bridge the gap between machine learning and real-world medical applications.

10. FUTURE SCOPE

1. Integration with Electronic Health Records (EHR): The prediction system can be integrated with hospital EHR systems to allow real-time data input and automated risk prediction for patients.
2. Incorporation of Deep Learning Models: Advanced deep learning techniques like CNNs or RNNs could be explored to enhance prediction accuracy, especially when dealing with imaging or sequential data.
3. Mobile App Deployment: Developing a mobile-based interface for doctors or patients can make the prediction model more accessible in rural or remote areas.
4. Continuous Model Improvement: The model can be retrained periodically with new clinical data to improve accuracy and adapt to changes in medical patterns.
5. Explainable AI Integration: Adding interpretability tools like SHAP or LIME could help clinicians understand why the model makes a specific prediction, increasing trust and adoption.
6. Multi-Disease Prediction System: The project can be expanded to predict other liver-related conditions or multiple diseases using the same patient dataset.
7. Language Localization & Accessibility: Making the tool available in regional languages with a voice-assist interface could improve usability for non-English speakers.

11. APPENDIX

Dataset Link: https://www.kaggle.com/datasets/bhavanipriya222/liver-cirrhosis-prediction

GitHub Link : https://github.com/Ramya-Kadali/predict-liver-cirrhosis

Project demo link:
https://drive.google.com/file/d/1WN7lATLbXgBPmHqNxIs5U4nLORau4BTU/view