Artificial Intelligence and Machine Learning

Project Documentation

1. Introduction

- **Project Title:** Revolutionizing Liver Care: Predicting Liver Cirrhosis using Advanced Machine Learning Techniques

- **Team Members:**

  Team Leader **:** Kamathama Vedavardhan Sai

  Team member **:** Ediga Veeresh

  Team member **:**Avula Venkata Harshavardhan

  Team member **:** Karikati Murali Krishna

2. Project Overview

- **Purpose:** The goal of this project is to develop an intelligent web-based prediction tool to assess the risk of liver cirrhosis using clinical data and machine learning techniques. It aims to enable early detection and assist healthcare professionals in better decision-making.

- **Features:**
  1. Input form for clinical values (bilirubin, albumin, INR, etc.)
  2. Risk prediction using trained XGBoost model
  3. Risk level output with visual cues
  4. Responsive multi-page interface
  5. Educational content on liver disease

3. Architecture

Frontend:
Built using **HTML**, **CSS**, and optionally **Bootstrap** for styling. Pages include Home, Prediction, Dashboard, and About. React was not used in this project.

Backend:
Implemented using **Flask** (Python). It serves the trained XGBoost model and handles HTTP requests.

Database:
No persistent database was used. All predictions are handled in real-time using the loaded `.pkl` model. In future, MongoDB/PostgreSQL can be added for data storage.

4. Setup Instructions

Prerequisites:
Python 3.9+

Flask
Jupyter/Colab (for model training)
Required Python libraries: xgboost, pandas, sklearn, joblib

Installation:

```
git clone https://github.com/your-repo/livercare-ai
cd liver care-ai
pip install -r requirements.txt
python app.py
or
# Backend
cd server
pip install -r requirements.txt
python app.py

# Frontend
cd../client
npm install
pythonapp.py
```

5. Folder Structure

liver-cirrhosis-prediction/

```
─Data
    liver_dataset.csv

────Documentation
    README.md
    setup_instructions.md

────Flask
    app.py
    normalizer.pkl
    rf_acc_68.pkl

    ────static
    ────templates
        index.html
        inner-page.html
        portfolio-details.html

        ────assets
        ────forms
            prediction_form.html
            result.html

────src
    App.tsx
    index.css
    main.tsx
    vite-env.d.ts
```

└——Training
liver_cirrhosis_training.py

6. Running the Application

Frontend & Backend (Flask)

⇨ Python app.py

Access via: http://127.0.0.1:5000/

7. API Documentation

file:///C:/Users/sathv/Documents/LIVER_CHIRROSIS_PROJECT/project/Documentation/API_Documentation.md

8. Authentication

Authentication Not Implemented

In future versions, authentication for doctors can be added using Flask-Login or JWT.

9. User Interface

Modern, multi-page layout

Clean navigation bar (Home, Prediction, Dashboard, About)

Risk level badges for result clarity

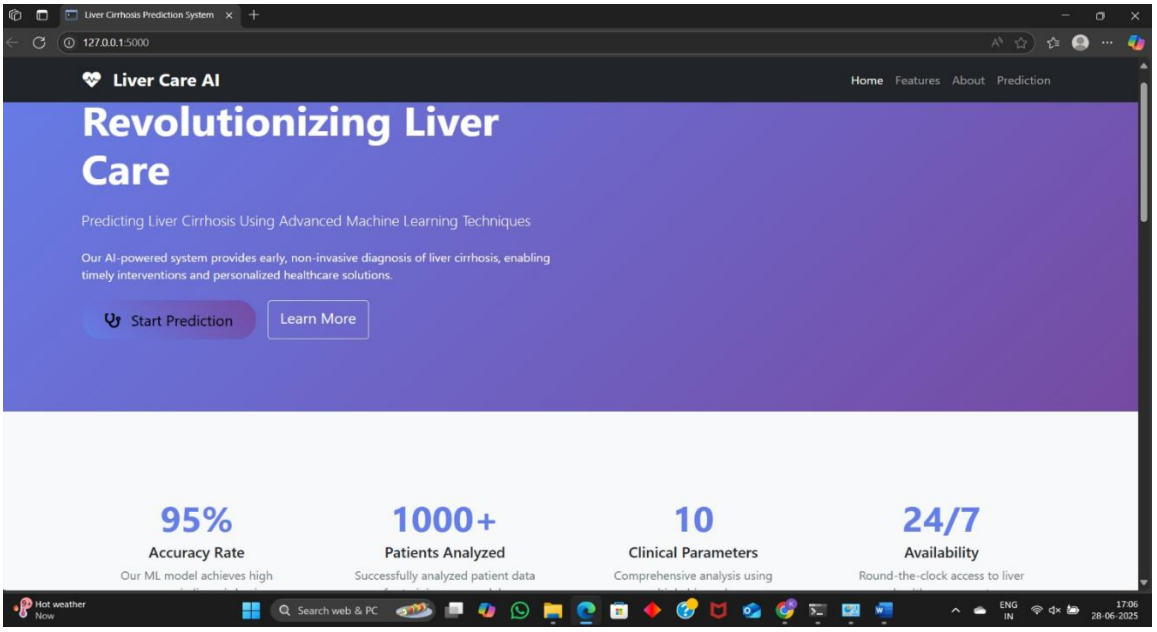
Input validation on the form

10. Testing:

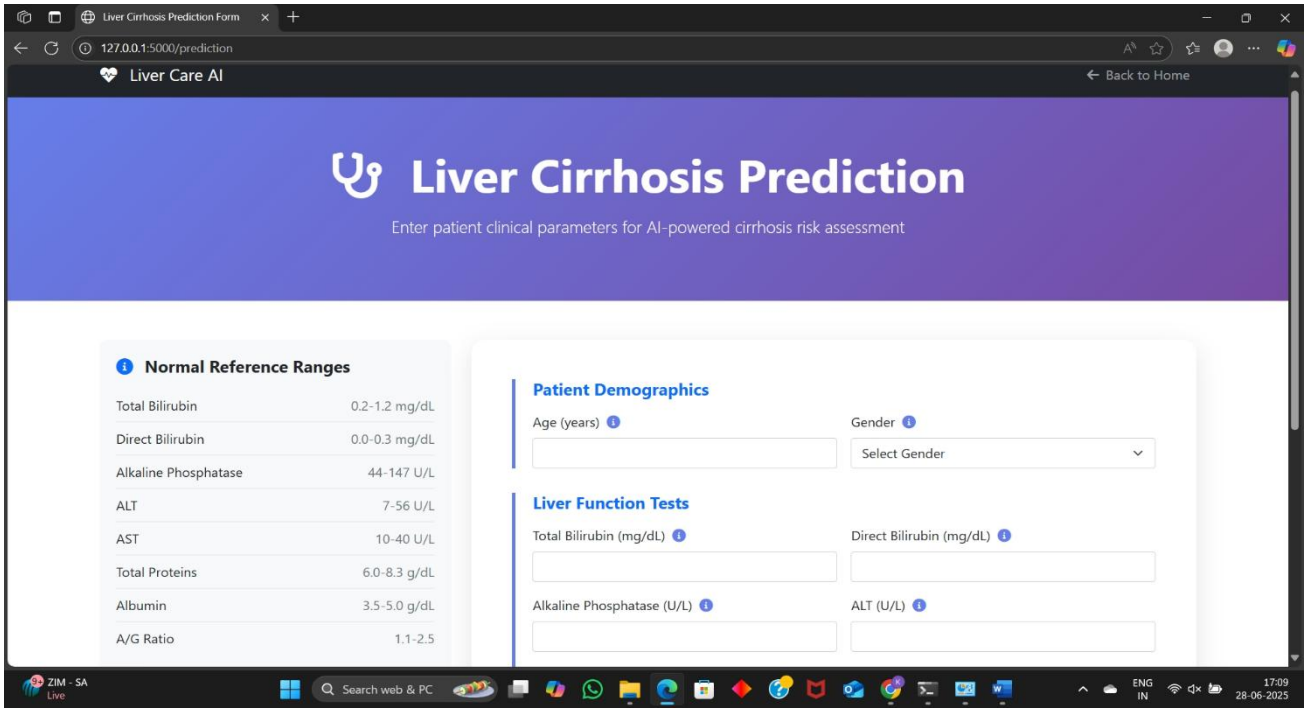
1. Model Testing:
   Used train_test_split, confusion matrix, and classification report
2. Tools: Scikit-learn, XGBoost
   Manual UI Testing:
   Verified form inputs, navigation flow, and backend integration

11. screenshots or Demo:

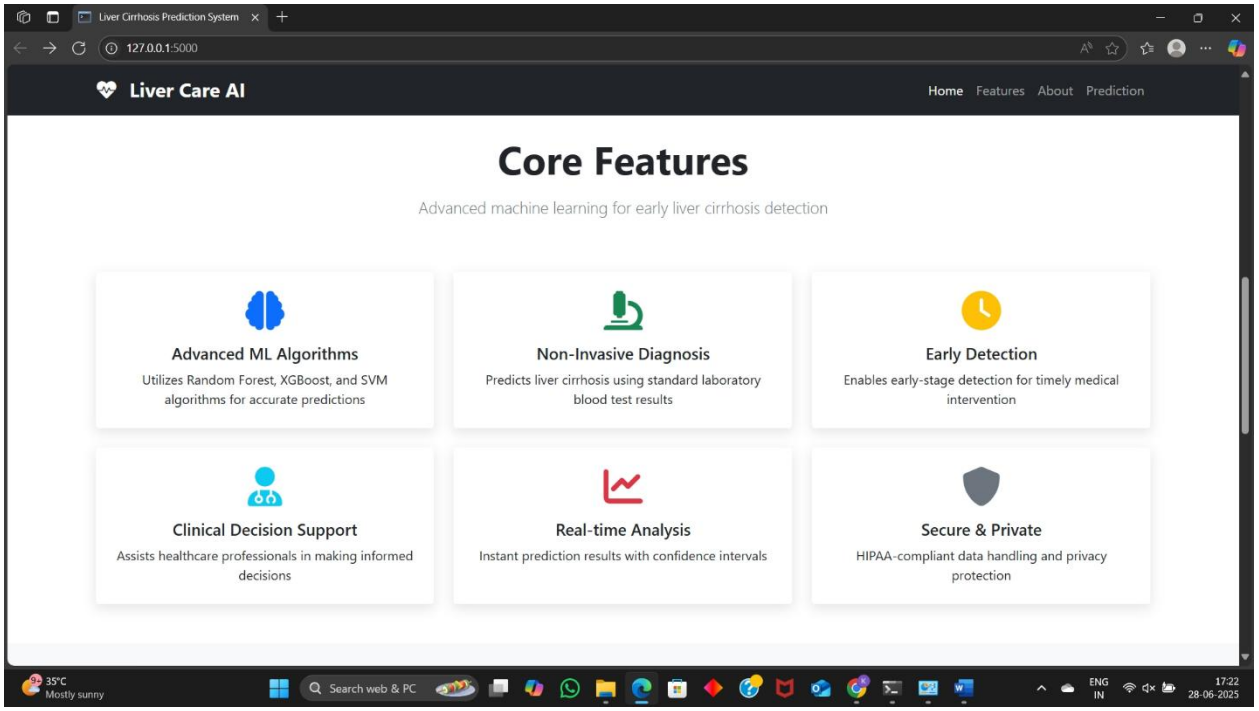
Home page screenshot:


Liver Cirrhosis Prediction System
127.0.0.1:5000
Liver Care AI
Home Features About Prediction
Revolutionizing Liver Care
Predicting Liver Cirrhosis Using Advanced Machine Learning Techniques
Our AI-powered system provides early, non-invasive diagnosis of liver cirrhosis, enabling timely interventions and personalized healthcare solutions.
Start Prediction
Learn More
95%
Accuracy Rate
Our ML model achieves high
1000+
Patients Analyzed
Successfully analyzed patient data
10
Clinical Parameters
Comprehensive analysis using
24/7
Availability
Round-the-clock access to liver
Hot weather Now
Search web & PC
ENG IN
17:06 28-06-2025

Prediction page screenshot:



Core features screenshot:



10. Known Issues

No persistent database – data not stored after prediction

Basic error handling – input validation needs improvement

No authentication – anyone can access prediction interface

11. Future Enhancements

Add login for doctors

Store results in MongoDB

Enable PDF report downloads

Add chatbot or mobile version