**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 27 June 2025 |
| Team ID | **LTVIP2025TMID35735** |
| Project Name | **Revolutionizing Liver Care : Predicting Liver Cirrhosis using Advanced Machine Learning Techniques** |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

**1. Data Layer (Input & Storage)**

Data Sources:

Hospital EMRs (Electronic Medical Records)

Public datasets (e.g., Indian Liver Patient Dataset – ILPD)

User-uploaded CSVs or manual entry via web form

Storage:

Structured data stored in SQL Database (e.g., MySQL/PostgreSQL)

Raw files (CSV/Excel) in Cloud Storage or local file system

**2. Data Processing & Feature Engineering**

Steps:

Null value imputation

Outlier detection & removal

Label encoding / one-hot encoding

SMOTE (for class imbalance)

Feature scaling (StandardScaler/MinMaxScaler)

Tools:

Pandas, NumPy, Scikit-learn

**3. Machine Learning Layer**

ML Algorithms:

Random Forest, XGBoost, Logistic Regression, SVM

Model Selection:

Train/Test split or K-Fold Cross Validation

Evaluate using Accuracy, F1-Score, ROC-AUC

Model Tuning:

Hyperparameter tuning via GridSearchCV

Model Saving:

Serialized with Joblib or Pickle

**4. Backend API Layer**

Framework:

Flask or FastAPI

Endpoints:

/predict – Accepts input and returns prediction

/train – (Optional) Re-trains model with new data

/feedback – Collects feedback from users for continuous improvement

**5. Frontend / UI Layer**

Framework:

Streamlit or Flask-based HTML templates

Features:

Data upload and visualization

Real-time liver health prediction form

Risk interpretation (Low / Moderate / High)

Historical prediction dashboard

**6. Monitoring & Feedback**

Model Monitoring:

Track model accuracy drift over time

Versioning of models

User Feedback Loop:

Allow users/doctors to validate model output

Save responses for retraining

**7. Deployment & Hosting**

Deployment Options:

Cloud: AWS (EC2/S3), Azure, GCP

Containerization: Docker

CI/CD: GitHub Actions / Jenkins for pipeline

Domain:

Host on a health-tech subdomain (e.g., livercare.ai)

**Solution Architecture Diagram:**

**┌────────────────────────────┐**

**│ Data Sources │**

**│ (EMR, ILPD, User Inputs) │**

**└────────────┬───────────────┘**

**↓**

**┌────────────────────────────┐**

**│ Data Preprocessing & │**

**│ Feature Engineering │**

**│ (Cleaning, SMOTE, Scaling) │**

**└────────────┬───────────────┘**

**↓**

**┌────────────────────────────┐**

**│ Machine Learning Models │**

**│ (Training, Evaluation) │**

**└────────────┬───────────────┘**

**↓**

**┌────────────────────────────┐**

**│ Model Serialization │**

**│ (Joblib / Pickle) │**

**└────────────┬───────────────┘**

**↓**

**┌────────────┐ ↓ ┌────────────────────────┐**

**│ Frontend │ ←──────→ │ Flask API Server │**

**│ (Streamlit │ │ (/predict, /train, etc.) │**

**└────────────┘ └──────────────────────────┘**

**↓**

**┌────────────────────────────┐**

**│ Output Predictions │**

**│ (Risk Score, Recommendations) │**

**└────────────────────────────┘**