Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	24 JUNE 2025
Team ID	LTVIP2025TMID33932
Project Name	Revolutionizing Liver Care : Predicting Liver Cirrhosis using Advanced Machine Learning Techniques
Maximum Marks	4 Marks

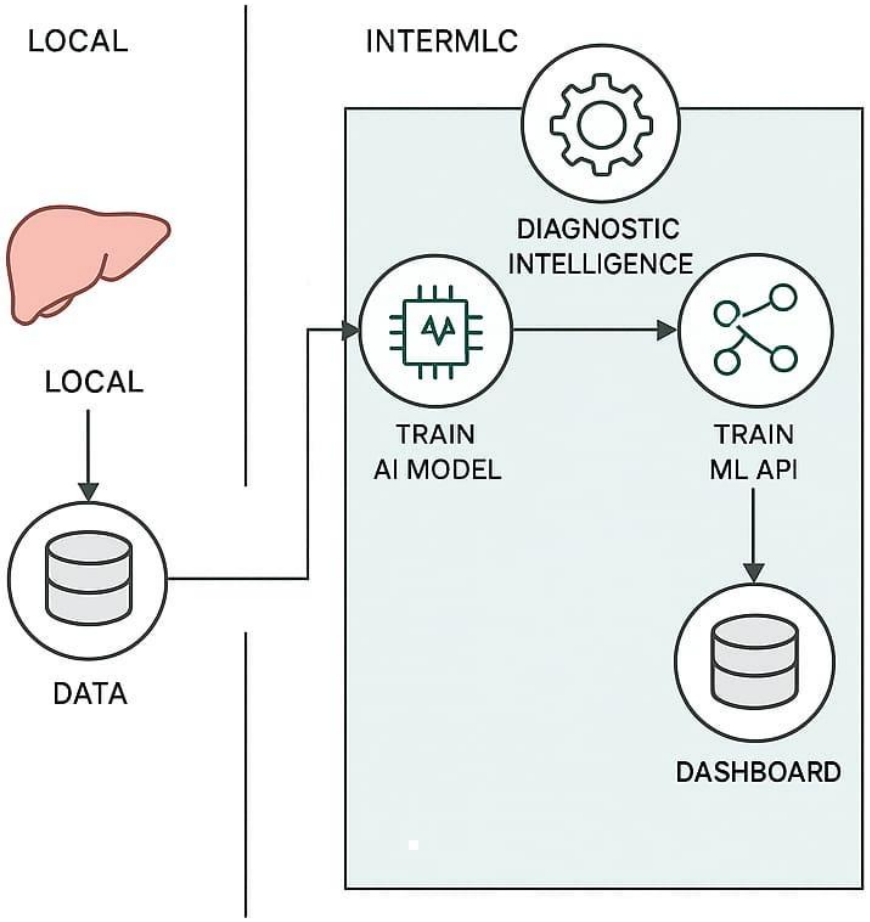
Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Order processing during pandemics for offline mode

Reference: https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/

Example Technical Architecture



GUIDELINES:

Include all the processes (As an application logic/ Technology Block)
Provide infrastructural demarcation (Local / Cloud)
Indicate external interfaces (third party API's etc.)
Indicate Data Storage components / services If applicable

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web-based dashboard for doctors and analysts	. HTML, CSS, JavaScript, React.js
2.	Application Logic-1	Data ingestion and preprocessing logic	Python (Pandas, NumPy, Requests)
3.	Application Logic-2	ML model training and cirrhosis risk prediction	Python (scikit-learn, XGBoost, TensorFlow/Keras)
4.	Application Logic-3	API service to expose model predictions	Flask / FastAPI
5.	Database	Store structured liver patient data	PostgreSQL / MongoDB
6.	Cloud Database	Scalable cloud database for patient analytics	AWS RDS / Google Cloud Firestore
7.	File Storage	Store raw CSV files, reports, and trained model artifacts	AWS S3 / Google Cloud Storage
8.	External API-1	Access medical terminology or drug info APIs (if needed)	NIH API / FDA Drug API (optional based on expansion)
9.	External API-2	(Optional) Integration with EHR/EMR for patient records	HL7 / FHIR APIs
10.	Machine Learning Model	Predict liver cirrhosis using clinical data	XGBoost / Random Forest / Logistic Regression
11.	Infrastructure	Cloud deployment, container orchestration & scaling	Docker, Kubernetes, AWS / GCP / Azure

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Use of open-source ML and web development tools	scikit-learn, Flask, React, MongoDB
2.	Security Implementations	Secure APIs, encrypted data flow, and controlled system access	HTTPS, JWT, OAuth2, IAM (AWS/GCP)
3.	Scalable Architecture	Microservice-based architecture with containerization and auto-scale	Docker, Kubernetes
4.	Availability	High availability via multi-zone cloud deployment and load balancing	AWS ELB, GCP Load Balancer
5.	Performance	Fast model inference with caching and optimized delivery	Redis, Flask async, Cloud CDN

References:

https://c4model.com/

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://www.ibm.com/cloud/architecture

https://aws.amazon.com/architecture

https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d