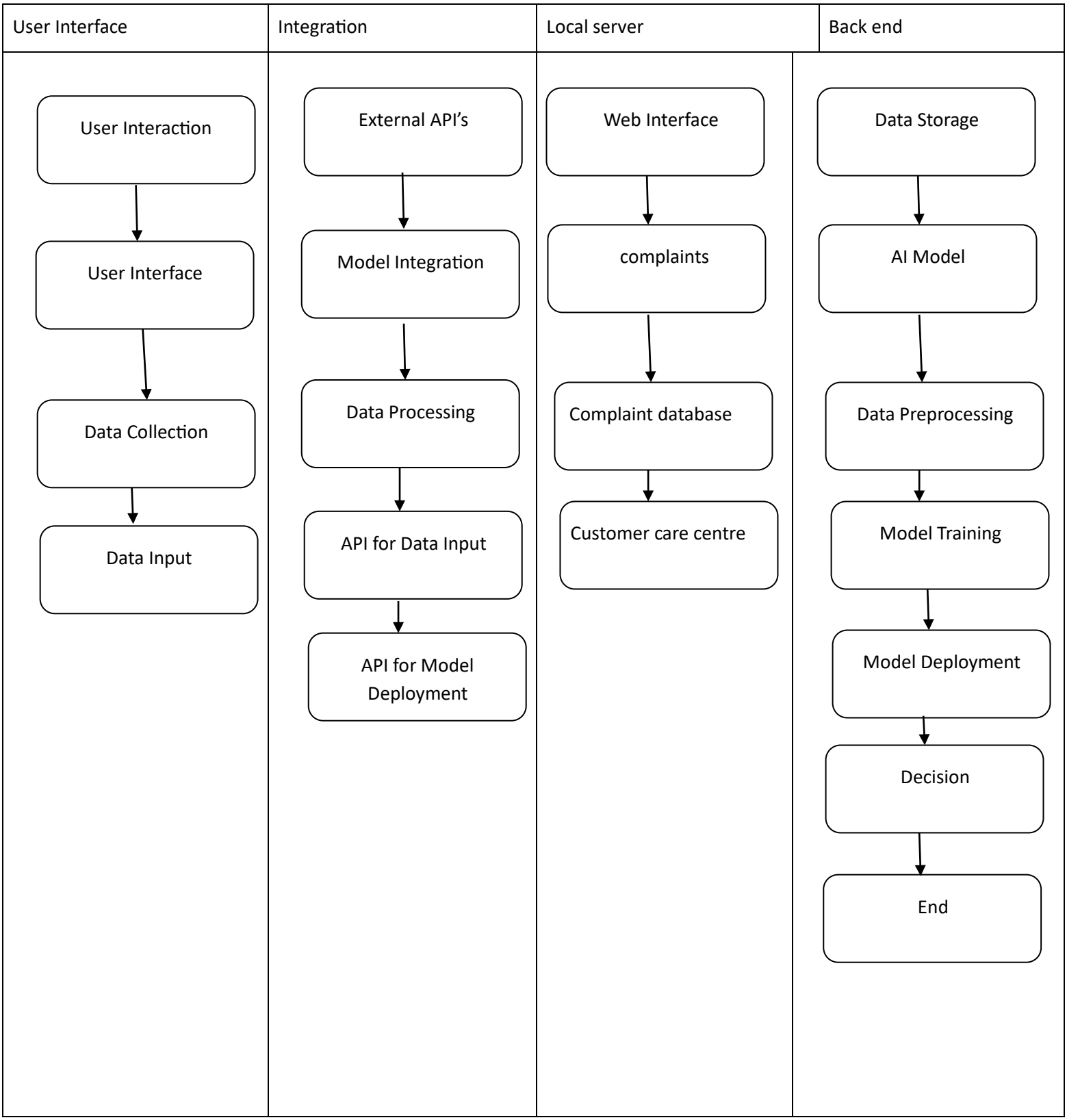
Project Design Phase-2

Technology Stack

Date	13-11-2023
Team ID	PNT2022TMID592372
Project Name	Diabetes Prediction Using Machine Learning
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable includes the architectural diagrams as below and the information as per the table1 (Components and Technolgies) & table 2 (Application Characteristics)


User Interface
Integration
Local server
Back end
User Interaction
User Interface
Data Collection
Data Input
External API's
Model Integration
Data Processing
API for Data Input
API for Model Deployment
Web Interface
complaints
Complaint database
Customer care centre
Data Storage
AI Model
Data Preprocessing
Model Training
Model Deployment
Decision
End

Component	Description	Technology
User Interaction	Here client connection happens, potentially through a web or versatile point of interaction.	Front-end Systems, Adobe XD, Figma, Sketch.
User Interface	The UI is where clients input information and associate with the computer based intelligence model.	HTML, CSS, JavaScript/Precise Js/Respond Js.
Data Collection	Information gathered from the UI is sent for additional handling.	SQL Information base, (e.g., MySQL), Python Libraries (e.g., Scrapy).
Data Input	This is the information placed by the client, which is shipped off the reconciliation layer for additional handling.	Python Libraries, glucose meters, consistent glucose screens and wearable gadgets.
External API's	These are points of interaction to outside information sources or administrations that can give extra information to the man-made intelligence model.	API keys, RESTful APIs, GraphQL
Model Integration	The mix layer consolidates the client's contribution with outside information hotspots for additional handling.	Python/Java scripts, or cloudbased incorporation administrations.
Data Processing	This step includes information preprocessing and change to make it appropriate for the computer based intelligence model.	AWS Glue, Google Dataflow
API for Data Input	This addresses the point of interaction through which client input is passed to the coordination layer.	Flask, Express.js, JSON, XML.

API for Model Deployment	A connection point that takes into account the arrangement and recovery of artificial intelligence model forecasts.	TensorFlow Serving, Flask
Data Storage	Data might be put away locally or in the cloud	Microsoft SQL Server, or Oracle, Amazon S3
Data Preprocessing	Data is cleaned, changed, and ready for model preparation.	Pandas, Numpy, seaborn, Matplotlib.
Model Training	The AI model is prepared on preprocessed information.	TensorFlow, PyTorch, ScikitLearn.
Model Deployment	The prepared artifitial intelligence model is sent to make expectations.	AWS Lambda, Azure Functions
Machine learning Models	This segment incorporates the prepared AI model that is utilized for expectations.	TensorFlow, PyTorch

Characteristics	Description	Technology
Data Privacy and Security	Carry out strong information protection and safety efforts to safeguard user data and keep up with consistence with significant guidelines	Encryption (e.g., SSL/TLS), access control
User Friendly Interface	The UI be instinctive and easy to use, particularly in the event that the application is planned for use by medical care experts and patients.	Web frameworks (e.g., React, Vue.js)
Accuracy	The application will give precise diabetes grouping results.	TensorFlow
Reliability	Users will actually want to reliably depend on the application's forecasts. It ought to have a serious level of consistency and strength.	Robust model validation techniques

Scalability	Justify the scalability of architecture	Cloud services (e.g., AWS, Google Cloud) for auto-scaling
Open-Source Frameworks	Flask is a lightweight Python web structure known for its straightforwardness and convenience an extraordinary decision for building little to medium-sized web applications.	Flask FrameWork Used
Availability	The application will be accessible on the nearby climate.	Local Host Used