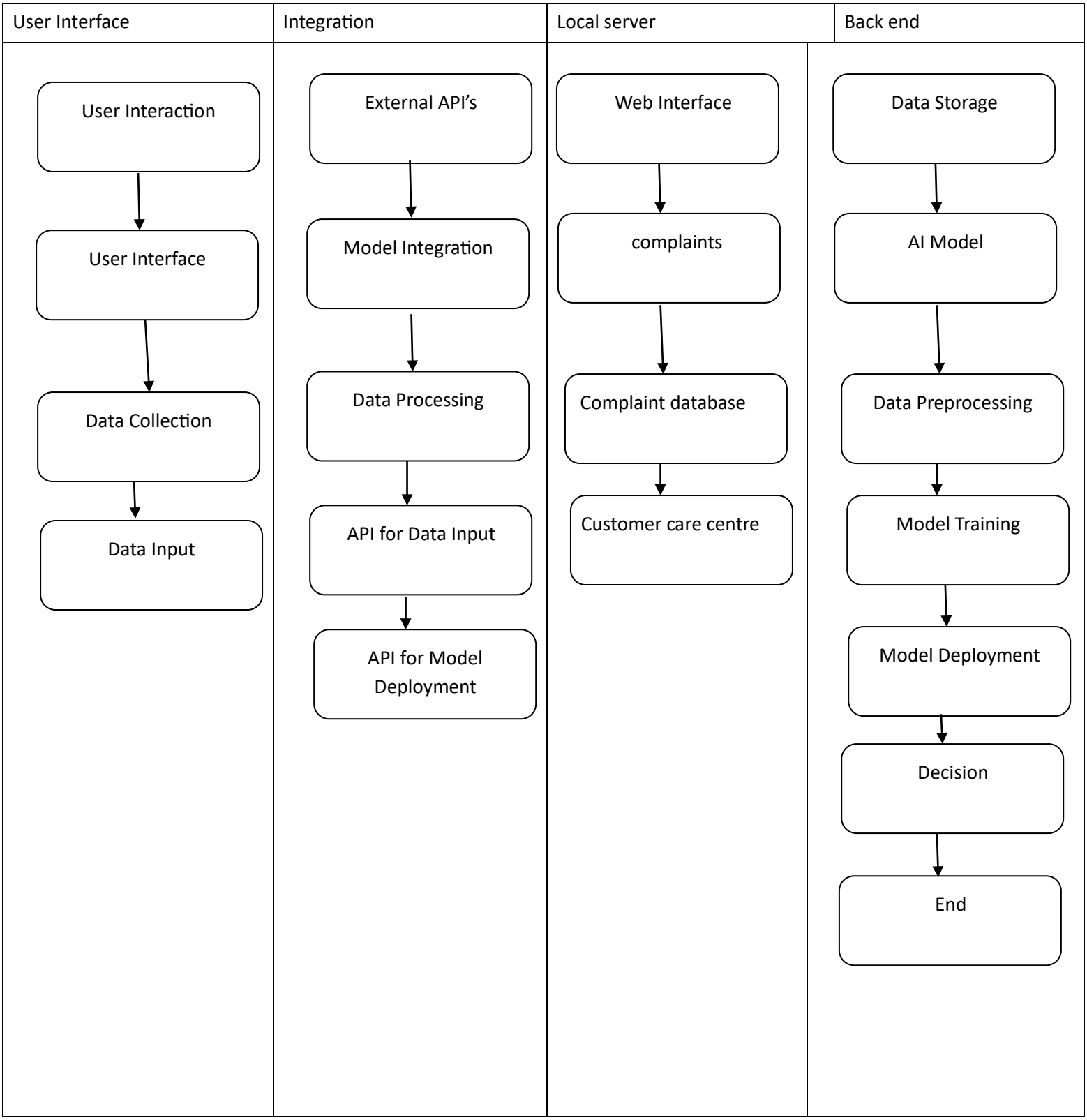
Project Design Phase-2

Technology Stack

Date	26-10-2023
Team ID	PNT2022TMID593014
Project Name	Diabetes Prediction Using Machine Learning
Maximum Marks	4 Marks

Technical Architecture:

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


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	This is the starting point where user interaction occurs, possibly through a web or mobile interface	Front-end Frameworks, Adobe XD, Figma, Sketch.
User Interface	The user interface is where users input data and interact with the AI model.	HTML, CSS, JavaScript / Angular Js /React Js
Data Collection	Data collected from the user interface is sent for further processing.	SQL Database, (e.g., MySQL), Python Libraries (e.g., Scrapy).
Data Input	This is the data entered by the user, which is sent to the integration layer for further processing.	Python Libraries, glucose meters, continuous glucose monitors (CGMs), and wearable devices.
External API's	These are interfaces to external data sources or services that can provide additional data for the AI model.	API keys, RESTful APIs, GraphQL
Model Integration	The integration layer combines the user's input with external data sources for further processing.	Python/Java scripts, or cloudbased integration services.
Data Processing	This step involves data preprocessing and transformation to make it suitable for the AI model.	AWS Glue, Google Dataflow
API for Data Input	This represents the interface through which user input is passed to the integration layer.	Flask, Express.js, JSON, XML.
API for Model Deployment	An interface that allows for the deployment and retrieval of AI model predictions.	TensorFlow Serving, Flask
Data Storage	Data may be stored locally or in the cloud	Microsoft SQL Server, or Oracle, Amazon S3
Data Preprocessing	Data is cleaned, transformed, and prepared for model training.	Pandas, Numpy, seaborn, Matplotlib.
Model Training	The machine learning model is trained on preprocessed data.	TensorFlow, PyTorch, ScikitLearn.

Model Deployment	The trained AI model is deployed to make predictions.	AWS Lambda, Azure Functions
Machine learning Models	This section includes the trained AI model that's used for predictions.	TensorFlow, PyTorch

Characteristics	Description	Technology
Data Privacy and Security	Implement robust data privacy and security measures to protect user data and maintain compliance with relevant regulations	Encryption (e.g., SSL/TLS), access control
User Friendly Interface	The user interface be intuitive and user-friendly, especially if the application is intended for use by healthcare professionals and patients.	Web frameworks (e.g., React, Angular, Vue.js)
Accuracy	The application will provide accurate diabetes classification results.	TensorFlow
Reliability	Users will be able to rely on the application's predictions consistently. It should have a high degree of consistency and stability.	Robust model validation techniques
Scalability	Justify the scalability of architecture	Cloud services (e.g., AWS, Azure, Google Cloud) for auto-scaling
Open-Source Frameworks	Flask is a lightweight Python web framework known for its simplicity and ease of use a great choice for building small to medium-sized web applications.	Flask FrameWork Used
Availability	The application will be available on the local environment.	Local Host Used