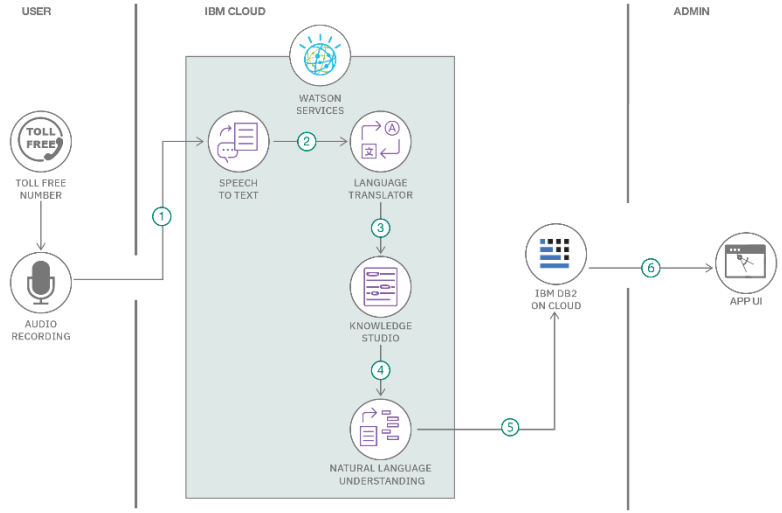
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	14 March 2025
Team ID	PNT2025TMID00705
Project Name	TrafficTelligence: Advanced Traffic Volume Estimation with Machine Learning
Maximum Marks	4 Marks

Technical Architecture:

The architecture of **TrafficTelligence** will be based on a **cloud-hosted, microservices-based** model with a **machine learning backend**. The system will include multiple components such as a web-based user interface, a backend processing system, a scalable cloud database, and an AI-driven traffic prediction model.



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
- Indicate interface to machine learning models (if applicable)

S.No	Component	Description	Technology
1.	User Interface	Web UI for traffic data visualization, reports, and analytics	React.js, Tailwind CSS, Material UI
2.	Mobile Interface	Mobile app for real-time traffic updates	React Native, Flutter
3.	Backend Application Logic	Business logic for traffic volume estimation	Python (FastAPI, Flask, Django)
4.	Machine Learning Model	AI model for traffic volume prediction	TensorFlow, Scikit-learn, XGBoost
5.	Database	Storage of historical traffic data	PostgreSQL, MongoDB (NoSQL)
6.	Cloud Database	Cloud-based data storage	Google BigQuery, Firebase Realtime Database
7.	File Storage	Storing uploaded traffic datasets	AWS S3, Google Cloud Storage
8.	External API-1	Real-time traffic data	Google Maps API, TomTom Traffic API
9.	External API-2	Weather data for prediction improvement	OpenWeather API, IBM Weather API
10.	Deployment Infrastructure	Cloud platform for hosting	AWS EC2, Kubernetes, Docker, Google Cloud Run
11.	Authentication & Security	User authentication & security implementation	Firebase Authentication, OAuth 2.0, JWT

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Frameworks used for development (backend, frontend, database)	React.js, Flask, TensorFlow, PostgreSQL

S.No	Characteristics	Description	Technology
2.	Security Implementations	Data encryption, role-based access control, firewall security	AES Encryption, SHA-256, IAM Controls, OAuth 2.0
3.	Scalable Architecture	Microservices-based, cloud-native, serverless components	Kubernetes, Docker, AWS Lambda
4.	Availability	High availability via distributed architecture, load balancers, failover support	AWS Load Balancer, Kubernetes Cluster
5.	Performance	Caching, API rate limiting, CDN for faster data delivery	Redis Cache, Cloudflare 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