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

Date	19 November 2023
Team ID	Team-593004
Project Name	Restaurant Recommendation System
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/

Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
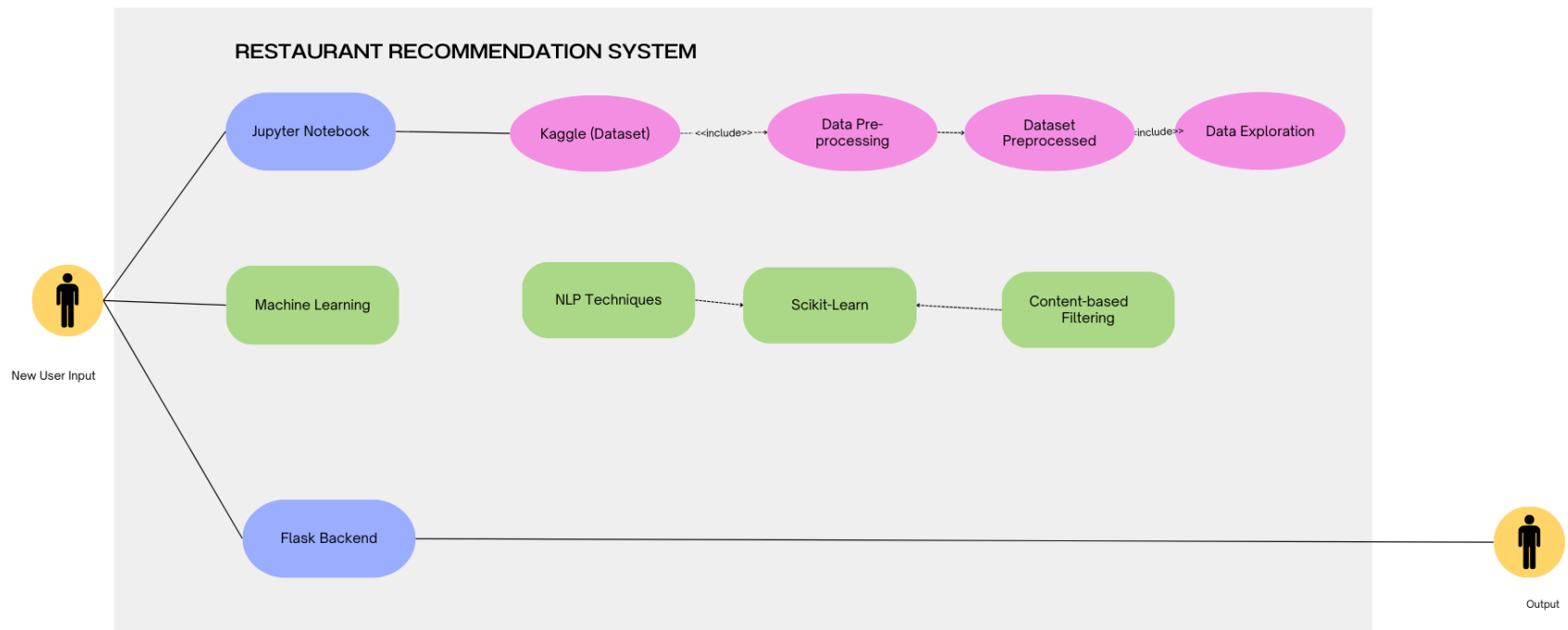5. Indicate interface to machine learning models (if applicable)

RESTAURANT RECOMMENDATION SYSTEM

New User Input

Jupyter Notebook

Kaggle (Dataset) — <<include>> → Data Pre-processing — Dataset Preprocessed — <<include>> — Data Exploration

Machine Learning

NLP Techniques → Scikit-Learn ← Content-based Filtering

Flask Backend

Output

Table-1 : Components & Technologies:

S. No	Component	Description	Technology
1	User Interface	Web UI for user interaction	HTML, CSS, JavaScript, Flask
2	Application Logic-1	Data Pre-processing and Cleaning Logic	Python (Jupyter Notebook for development)
3	Application Logic-2	Content-Based Recommender System Logic	Python (Scikit-Learn, NLTK)
4	Application Logic-3	Web Application Building Logic	Python (Flask)
5	Database	Local storage for Zomato Bangalore dataset	CSV File, Pandas
6	Cloud Database	Cloud-based database for scalability (Optional)	Not Used
7	Machine Learning Model	Content-Based Recommender System Model	Scikit-Learn, Cosine Similarity

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	Utilization of open-source frameworks for development	Flask (Python Web Framework), Scikit-Learn
2	Security Implementations	Implementation of security measures for data protection	SSL/TLS Encryption, Secure Coding Practices
3	Scalable Architecture	Adoption of a scalable architecture for future expansion	Microservices Architecture (Optional)
4	Availability	Ensuring high availability through appropriate strategies	Load Balancers (Optional), Redundancy Strategies
5	Performance	Design considerations for optimizing application performance	Caching Mechanisms, CDN Integration (Optional)

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