Project Design Phase-I

Solution Architecture

Date	19 September 2022
Team ID	**NM2023TMID01707**
Project Name	FOOD TRACKING SYSTEM
Maximum Marks	10 Marks

Solution Architecture:

Designing the architecture for a food tracking system involves considering various components, modules, and their interactions to ensure a scalable, reliable, and efficient solution. Here's a high-level outline of a solution architecture for a food tracking system:

User Interface (UI):

Description: The front-end component that users interact with to input food data, view tracked information, and receive feedback.
Technologies: HTML5, CSS, JavaScript, and a front-end framework like React or Angular.
Application Layer:

Description: Handles the business logic, user authentication, and communication between the UI and the backend services.
Technologies: Node.js, Python (Django/Flask), or another suitable backend framework.
Authentication and Authorization:

Description: Manages user authentication and authorization to ensure secure access to the system.
Technologies: OAuth 2.0, JSON Web Tokens (JWT), or other authentication protocols.
Database:

Description: Stores user profiles, food data, and other relevant information.
Technologies: MongoDB, PostgreSQL, or another database system based on data requirements.
Food Database:

Description: Contains information about various foods, including nutritional data.
Technologies: Use a dedicated database or API integration with existing food databases like USDA FoodData Central.
Machine Learning Models:

Description: Performs tasks such as food recognition, calorie estimation, and nutritional analysis.
Technologies: TensorFlow, PyTorch, or another machine learning framework.
APIs:

Description: Provides a set of APIs to allow communication between different components of the system.
Technologies: RESTful APIs or GraphQL for communication between the application layer, database, and machine learning models.
Image Recognition Service:

Description: If the system involves image recognition for identifying foods, integrate a dedicated service or library.
Technologies: TensorFlow Serving, OpenCV, or specialized image recognition APIs.
Notification Service:

Description: Sends notifications to users based on their preferences or goals.
Technologies: Firebase Cloud Messaging (FCM), Pusher, or a similar notification service.
Logging and Monitoring:

Description: Tracks system activities, logs errors, and monitors performance.
Technologies: ELK Stack (Elasticsearch, Logstash, Kibana), Prometheus, or other logging and monitoring tools.

Example - Solution Architecture Diagram: