**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 26 June 2025 |
| Team ID | LTVIP2025TMID52122 |
| Project Name | Comprehensive Analysis and Dietary Strategies with Tableau |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

**Table-1: Components & Technologies:**

| S.No. | Component | Description |
| --- | --- | --- |
| 1 | User Interface | Web-based dashboard and mobile app interface for users to log meals, view visualizations, and receive insights. |
| 2 | Application Logic-1 | User authentication and registration logic, including login via email, Gmail, or LinkedIn. |
| 3 | Application Logic-2 | Logic for processing dietary logs and converting food items into quantified nutrients. |
| 4 | Application Logic-3 | Logic for triggering visual insights and alerts based on nutrient analysis and health goal alignment. |
| 5 | Database | Stores structured user data, dietary logs, food databases, preferences—using MySQL or PostgreSQL. |
| 6 | Cloud Database | Cloud-hosted database (e.g., Amazon RDS or Google Cloud SQL) for scalability and high availability. |
| 7 | File Storage | Stores uploaded meal photos, personalized reports, and exported dashboards—via AWS S3 or similar. |
| 8 | External API-1 | Nutrition Analysis API for retrieving macro/micronutrient values of user-input food items. |
| 9 | External API-2 | Integration with wearable APIs (e.g., Fitbit, Apple Health) to collect biometric and activity data. |
| 10 | Machine Learning Model | Recommends dietary adjustments and predicts nutrient deficiencies based on consumption patterns. |

**Table-2: Application Characteristics:**

| S.No. | Characteristics | Description | Technology |
| --- | --- | --- | --- |
| 1 | Open-Source Frameworks | Frameworks used for frontend/backend development, preprocessing, and integration with Tableau | React.js (UI), Node.js (Backend), Python (Data Processing), Flask |
| 2 | Security Implementations | Ensures data confidentiality, user authentication, and secure transmission of dietary logs and health profiles | HTTPS, OAuth 2.0, JWT Tokens, AES Encryption, OWASP Compliance |
| 3 | Scalable Architecture | 3-Tier architecture with modular design, enabling component-wise scaling and integration with external APIs and Tableau | Microservices architecture with Docker & Kubernetes deployment |
| 4 | Availability | High availability achieved through cloud load balancers, redundant server clusters, and real-time failover mechanisms | AWS Elastic Load Balancer, Auto Scaling Groups, Multi-zone Deployment |
| 5 | Performance | Optimized for fast dashboard rendering and user responsiveness using caching, query optimization, and edge content delivery | Redis Cache, Tableau Extracts (TDE/Hyper), Cloudflare CDN |