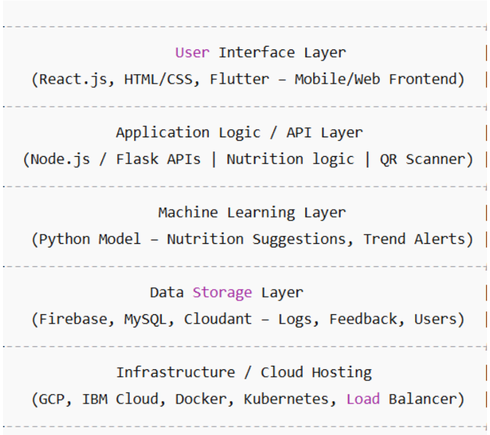
Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	24 January 2025
Team ID	LTVIP2025TMID49554
Project Name	Comprehensive Analysis & Dietary Strategies
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 Layer                    |
   (React.js, HTML/CSS, Flutter – Mobile/Web Frontend)    |

              Application Logic / API Layer               |
  (Node.js / Flask APIs | Nutrition logic | QR Scanner)    |

                  Machine Learning Layer                  |
   (Python Model – Nutrition Suggestions, Trend Alerts)   |

                    Data Storage Layer                    |
   (Firebase, MySQL, Cloudant – Logs, Feedback, Users)    |

              Infrastructure / Cloud Hosting              |
   (GCP, IBM Cloud, Docker, Kubernetes, Load Balancer)    |
```

Table-1 : Components & Technologies:

S.No	**Component**	**Description**	**Technology**
1	User Interface	How user interacts with application via mobile and web platforms	HTML, CSS, Java Script, Flutter(for mobile)

2	Application Logic-1	Logic for food logging, daily meal entries, and QR code scan processing	Java / Python
3	Application Logic-2	Logic for real-time dietary suggestions and nutritional gap detection	IBM Watson NLP or custom Python logic
4	Application Logic-3	Logic for generating diet challenges, reminders, and visual insights	Node.js / Flask API backend
5	Database	Stores user info, logs, meal data, mess menus, and preferences	MySQL or Mongo DB.
6	Cloud Database	NULL	NULL
7	File Storage	Storage of feedback reports, diet charts, and admin-generated reports	Firebase Storage / AWS S3
8	External API-1	Nutrient data from external verified sources	USDA FoodData Central API/ Edamam Nutrition API
9	External API-2	Health and lifestyle	ArogyaSetu API or Google Fit / Fitbit APIs

		integration	
10	Machine Learning Model	Predicting diet recommendations based on patterns, lifestyle, and past logs	Nutrition recommendation model (Python, Scikit-learn, TensorFlow Lite)
11	Infrastructure (Server / Cloud)	App deployment and hosting	IBM Cloud / AWS / Google Cloud Platform Docker + Kubernetes for scaling

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Utilizes open-source frontend and backend frameworks for building the mobile/web app.	React.js, Node.js, Flask (Python), Bootstrap, Firebase SDK
2	Security Implementations	Implements secure user authentication and data protection mechanisms. Includes encryption, HTTPS, and firewalls.	SHA-256, JWT, OAuth 2.0, HTTPS, Firebase Authentication, OWASP security best practices
3	Scalable Architecture	Designed with a microservices-based architecture to support multiple modules (logs, feedback, recommendations).	Kubernetes, Docker, REST APIs, Microservices using Flask/Node.js
4	Availability	Ensures high availability through cloud infrastructure, load balancing, and real-time backup.	AWS/GCP Load Balancer, Distrbiuted DB(Cloud Fire Store), Auto -scaling groups

5	Performance	Optimized for fast response using CDN, caching layers, and efficient backend queries for large student bases.	Cloudflare CDN, Redis Cache, Indexed NoSQL queries, Lazy loading (React)