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

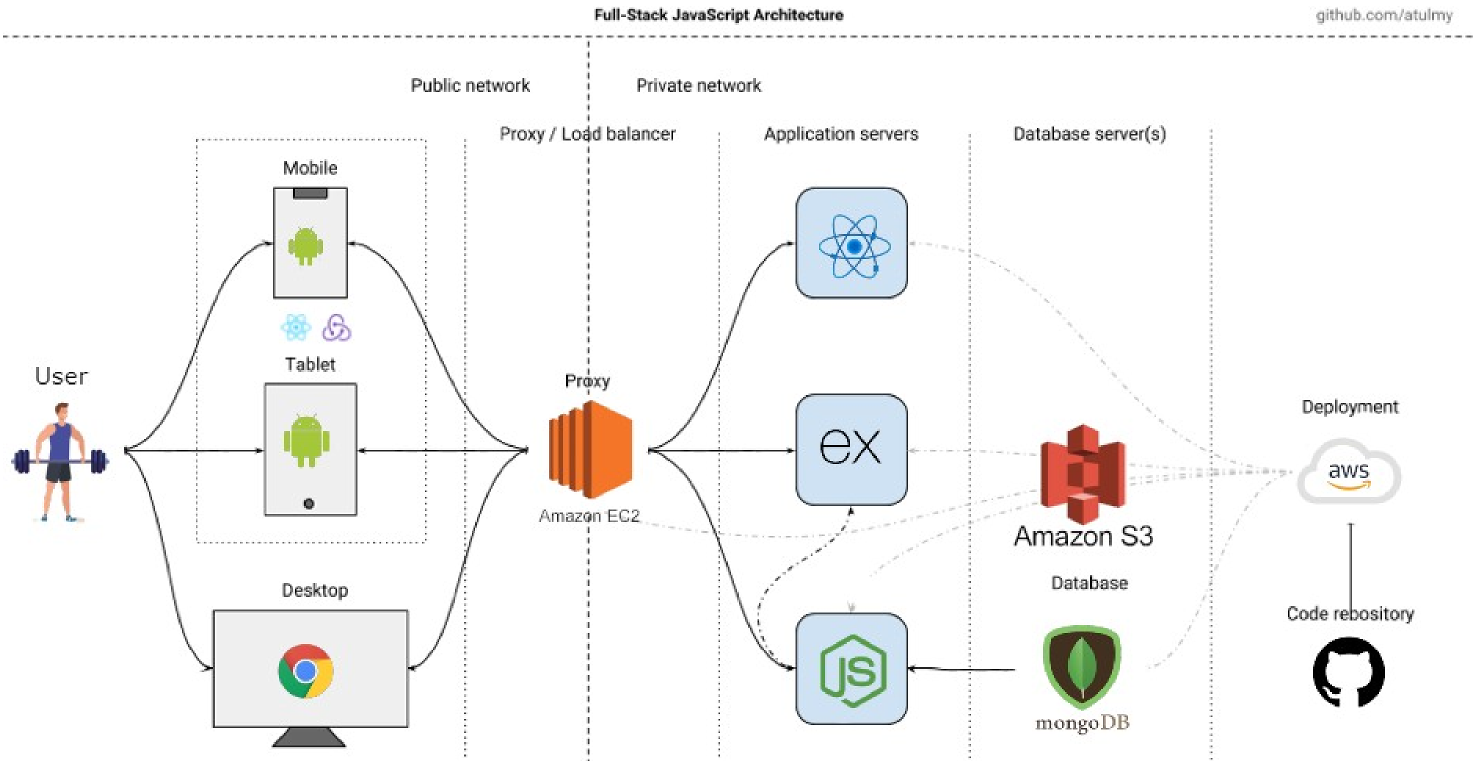
Date	23 October 2022
Team ID	PNT2022TMID09928
Project Name	AI-powered Nutrition Analyzer for Fitness Enthusiasts
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table 1 and table 2

Example: AI-powered Nutrition Analyzer for Fitness Enthusiasts

Reference: https://www.healthifyme.com/pro/



S.no	Component	Description	Technology
1	User Interface	How user interacts with application	HTML, CSS, Javascript, JQuery
2	UI / UX Framework	Frameworks used to make User interface comfortable	Tailwind CSS, React JS, Cloudflare

3	Application Logic - 1	CRUD Operations	Axios, Express, Router
4	Application Logic - 2	Backend connectivity Logic	Mongoose, Node JS
5	Database	Data Type, Configurations	MongoDB
6	Cloud Database	Database Service on Cloud	AWS S3 Buckets
7	File Storage	File storage requirements	Local Storage, JST Session
8	External API	User Authentication	Google OAuth 2.0
9	ML Model	Training dataset for model	Object Recognition Model
10	Infrastructure Server	Local Server Configuration	Kubernetes, Docker, Redux
11	Infrastructure Cloud	Cloud Server Configuration	AWS EC2 Instances

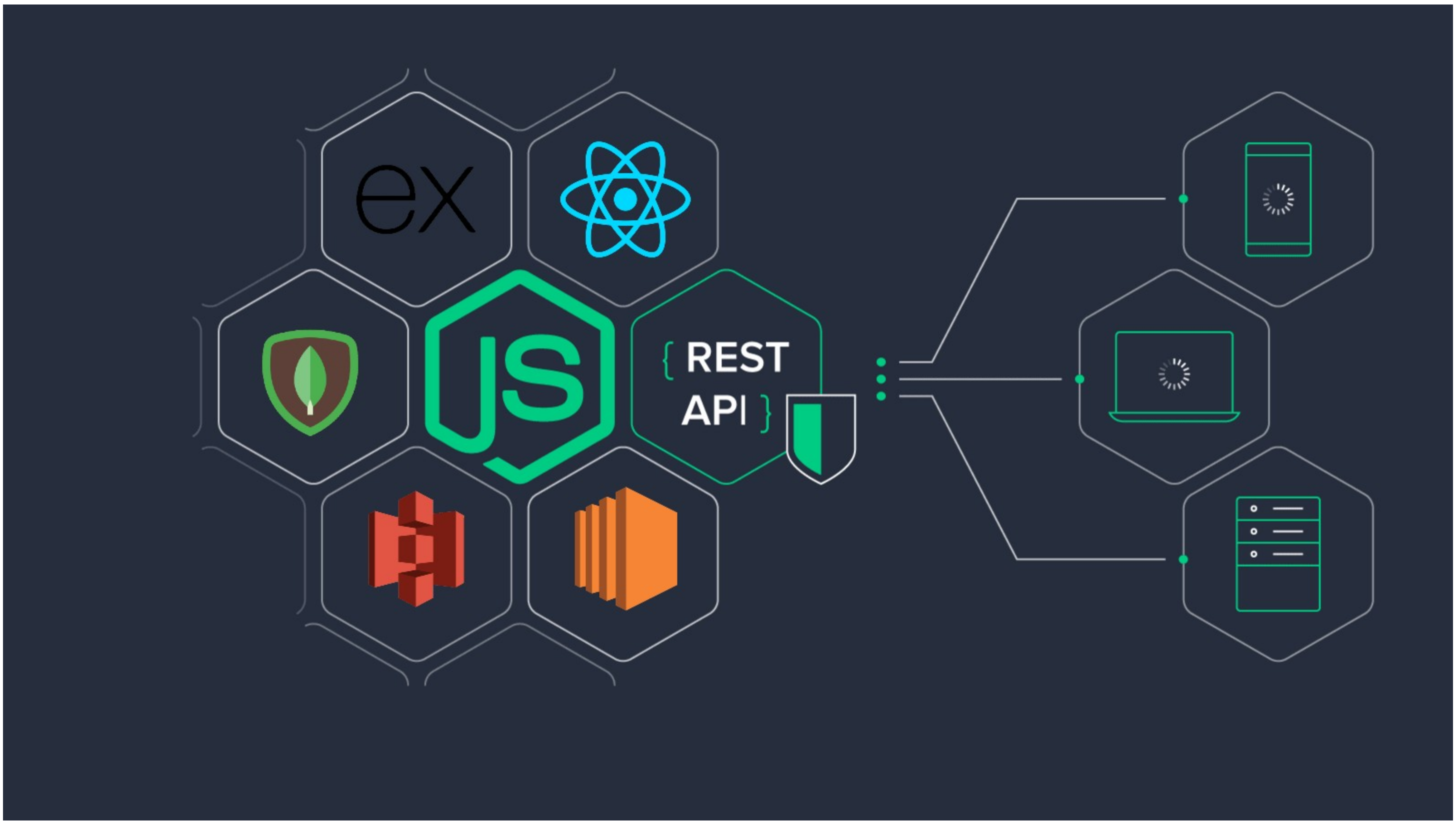


Table-2: Application Characteristics

S.no	Characteristics	Description	Technology
1	Open Source Frameworks	List of open-source frameworks used	React JS, Express, Mongoose, Axios
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	AWS Complaince, IAM Roles, Security Groups, Decrypt and ObjectId , SSH
3	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Windows t2.micro or t3.micro instance by AWS EC2 Instance
4	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Nginx Reverse Proxy, PuttyGen with Ubuntu, Load balancers and elastic IP
5	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Single Console for monitoring, Use of Manage Engine Corporate Data center, Agentless monitering

Referernces :

Tech Stack : MERN Stack Architecture - By Hackemoon

AWS S3 Architecture : S3 Buckets Architecture Blogs - By AWS

AWS EC2 Architecture : EC2 Instance Architecture Blogs - By AWS

MongoDB and NoSql Architecture : MongoDB Architecture Guide - By MDB

Working Example Architecture : https://www.healthifyme.com/pro/