Requirement Gathering and Analysis Phase
Technology Stack (Architecture & Stack)

Date	6/7/2024
Team ID	SWTID1719923176
Project Name	Project - FreelanceFinder: Discovering Opportunities, Unlocking Potential
Maximum Marks	

Technical Architecture:

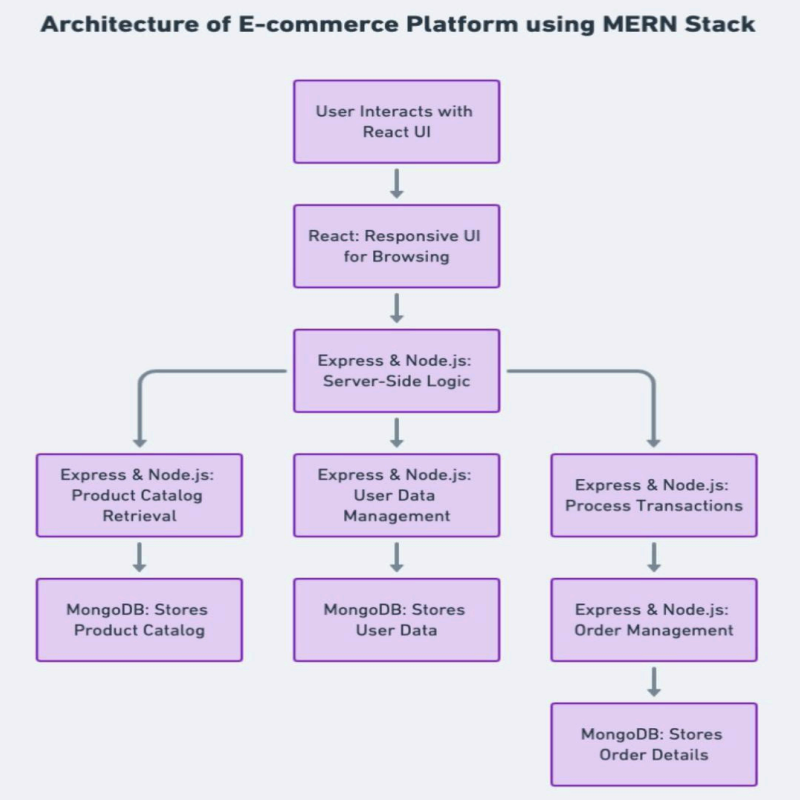

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1	Frontend	Web UI allowing freelancers and clients to interact	Html, CSS, React.js, Axios, React router
2	Backend logic 1(Frontend Interaction)	Handles server-side operations related to user profiles, job listings, and interactions with the frontend.	Node.js (server-side runtime). Express.js (web application framework). Middleware for authentication (e.g., Passport.js). Mongoose (for MongoDB interaction).
3	Backend Logic-2 (Backend Processing)	Manages authorization processes.	Node.js (server-side runtime). Express.js (web application framework).
4	Authentication	JWT (JSON Web Tokens) for user authentication	Express.js middleware
5	Database	MongoDB storing user profiles, job listings, etc.	MongoDB (NoSQL)
6	Cloud Storage	storing file uploads	AWS
7	Payment Gateway	To make payments	Stripe
8	Email Service	to send mails	SendGrid
9	Cloud platform	Deploy frontend (React) and backend (Node.js + Express)	AWS

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
	Open-Source Frameworks	It simplifies building APIs and handling routes, middleware, and requests.	Express.js (Node.js)

S.No	Characteristics	Description	Technology
1	Security Implementations	SHA-256: A cryptographic hash function used for secure data integrity. Encryptions: Techniques like AES (Advanced Encryption Standard) or RSA (Rivest–Shamir–Adleman) for protecting sensitive data. IAM (Identity Access Management): Controls access to resources based on user roles and permissions. Firewalls: Network security devices that filter incoming and outgoing traffic.	SHA-256, Encryptions, IAM Controls, OWASP etc.
2	Scalable Architecture	3-tier: A common architecture with presentation, application, and database tiers. Microservices: A modular approach where each component is a separate service, allowing scalability and independent deployment.	3 tier, Microservices
3	Availability	Load Balancers: Distribute incoming traffic across multiple servers to ensure high availability. .	Load balancers
4	Performance	Caching: Storing frequently accessed data in memory to reduce response time. Redis: In-memory data store for caching.	Caching,redis