

## Project Design Phase-II

### Technology Stack (Architecture & Stack)

|               |                              |
|---------------|------------------------------|
| Date          | 4 December 2025              |
| Team ID       |                              |
| Project Name  | FreelanceX -Freelance Finder |
| Maximum Marks | 4 Marks                      |

#### **Technical Architecture:**

The Freelance Finder platform follows a flexible and scalable 3-layer architecture, where the frontend handles user interaction, the backend manages all application processes, and the database securely stores system data. This setup ensures smooth performance, strong security, and easy integration with external services such as payment gateways.

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

| Component                  | Description  | Technology                       |
|----------------------------|--|----------------------------------|
| User Interface             | Interactive web screens for freelancers and clients to browse jobs, apply, and manage activities.    | HTML, CSS, JavaScript / React Js |
| Application Logic – Part 1 | Handles job creation, proposals, user dashboards, and real-time communication.                       | Node.js, Express.js              |
| Application Logic – Part 2 | Manages admin operations such as user monitoring, job oversight, and system notifications.           | React js, Node js                |
| Database                   | Stores all platform information including user records, job posts, proposals, messages, and reviews. | MongoDB                          |

**Table-2: Application Characteristics:**

| Characteristics        | Description  | Technology                   |
|------------------------|--|------------------------------|
| Open-Source Frameworks | Uses lightweight and flexible open-source tools to build | React.js, Node.js, BootStrap |

| Characteristics       | Description  | Technology    |
|-----------------------|--|---------------|
|                       | responsive UI and strong backend services.   |               |
| Scalable Architecture | Built using REST-based API communication and a 3-layer architecture to support growth and feature expansion. | MVC Framework |

**References:**

[React.js Documentation](#)

[Node.js Best Practice](#)

[JSON Web Server Reference](#)

<https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d>