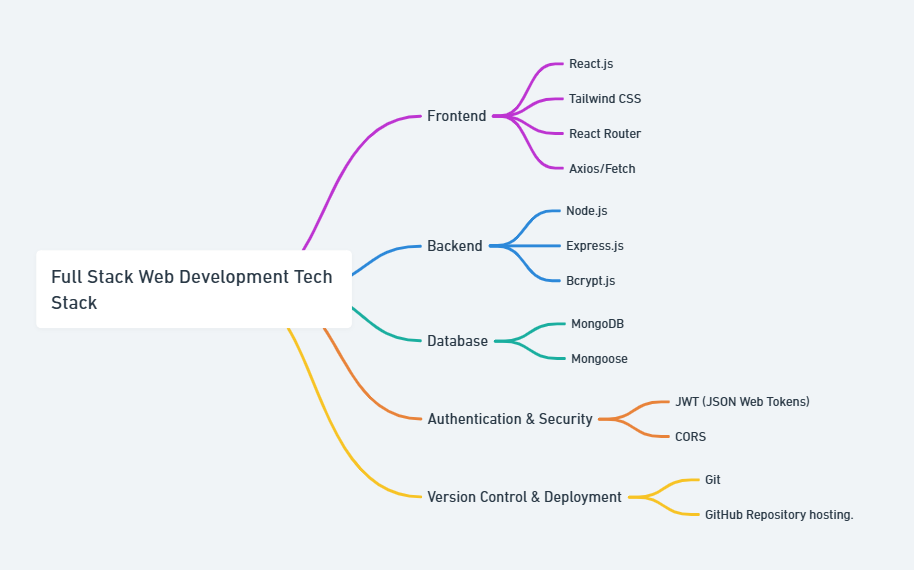
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 5 March 2025 |
| Team ID | SWTID1743519439 |
| Project Name | BookNest |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

****

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

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | How user interacts with application e.g.  Web UI, Mobile App, Chatbot etc. | |  | | --- | |  |  |  | | --- | | HTML, CSS, JavaScript, React.js,  Tailwind CSS, React Router | |
|  | Application Logic-1 | Logic for a process in the application | Node.js, Express.js |
|  | Application Logic-2 | Logic for a process in the application | JWT, CORS for Authentication/Security |
|  | Application Logic-3 | Logic for a process in the application | Axios/Fetch for API requests, Bcrypt.js for password hashing |
|  | Database | Data Type, Configurations etc. | MongoDB, Mongoose |
|  | File Storage | File storage requirements | Local filesystem (for book images, profile pictures) |
|  | Infrastructure (Server / Cloud) | Application Deployment on Local System | |  | | --- | |  |  |  | | --- | | GitHub (for repo), Vercel | |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | List of open-source frameworks used to build the application, reducing cost and increasing flexibility | React.js, Node.js, Express.js, MongoDB |
|  | Security Implementations | List of all the security/access control measures like authentication, password encryption, and protection from common threats | JWT, Bcrypt.js, HTTPS, CORS, OWASP security principles |
|  | Scalable Architecture | Application is designed with scalability in mind using a modular (3-tier) architecture allowing horizontal scaling | MERN stack + REST APIs |
|  | Performance | Optimized performance via caching, efficient database queries, use of CDNs for static assets, and minimalistic frontend | React performance hooks, CDN for images/assets, MongoDB Indexing, Lazy loading, Axios optimization |