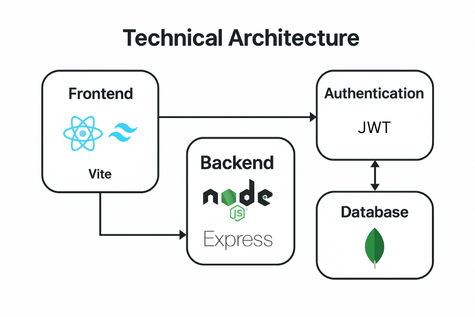
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 31 January 3035 |
| Team ID |  |
| Project Name |  |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

****

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

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  | User Interface | How user interacts with the app (UI/UX) | HTML, CSS, React JS |
|  | Application Logic-1 | Handles routing and API requests | Node.js |
|  | Application Logic-2 | Handles business logic like expense categorization | JavaScript (Node.js) |
|  | Application Logic-3 | Handles authentication, authorization | JWT, bcrypt |
|  | Database | Data types, schemas, configurations. | MongoDB (NoSQL) |
|  | Cloud Database | Cloud storage of DB | MongoDB Atlas |
|  | File Storage | If profile images/receipts are stored | Cloudinary / Firebase / Local Filesystem |
|  | External API-1 | Expense analysis/predictive APIs | Chart.js or ML APIs |
|  | Infrastructure (Server / Cloud) | Deployment environment | Vercel, Netlify (Frontend) / Render, Railway (Backend) |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | List the open-source frameworks used | React.js, Node.js, Express.js, MongoDB, Tailwind CSS |
|  | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | JWT, bcrypt, HTTPS, Helmet.js, CORS, OWASP Standards, Environment-based config files |
|  | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | RESTful APIs, Scalable DB (MongoDB Atlas) |
|  | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Deployment on Render / Railway |
|  | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Browser Caching, Indexed MongoDB Queries, Lazy Loading React Components |