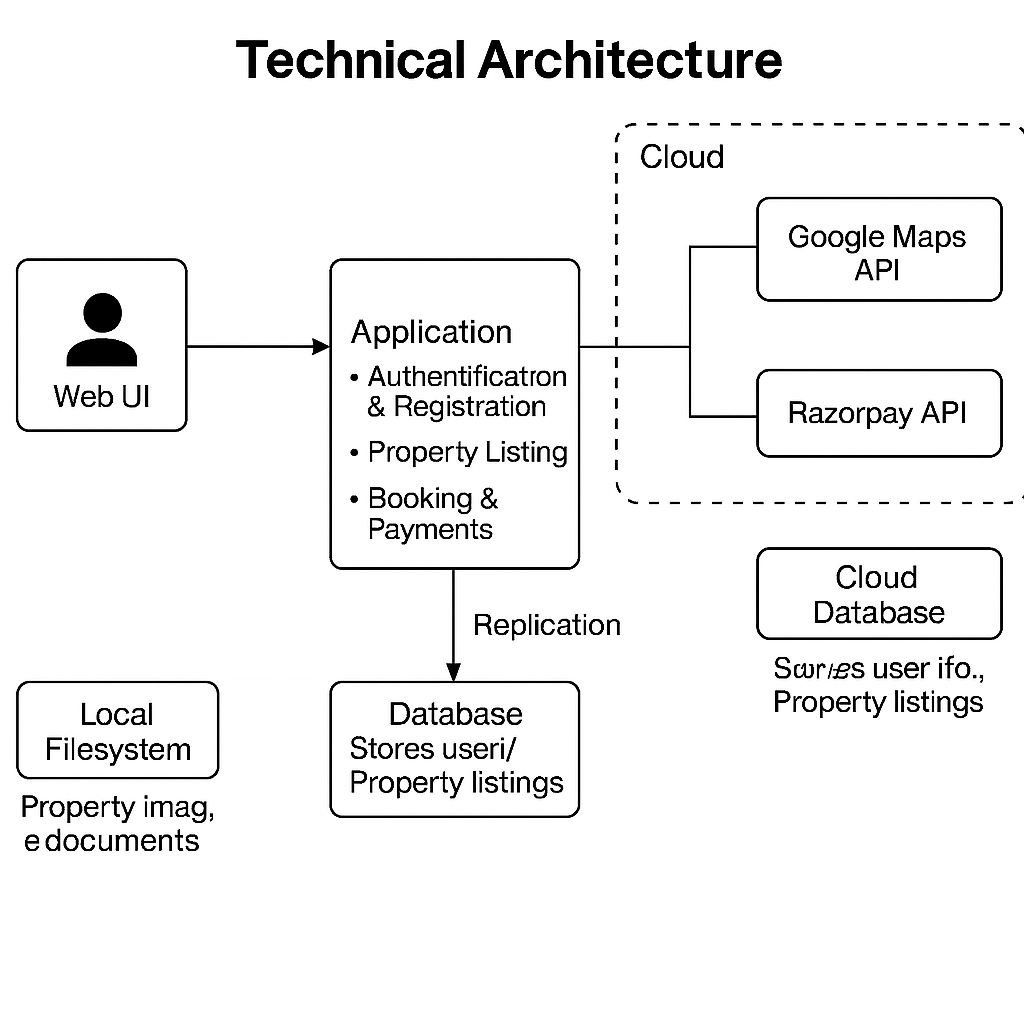
**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 14 April 2025 |
| Team ID | SWTID1743354369 |
| Project Name | House Rent App Using MERN – House Hunt |
| Maximum Marks | 4 Marks |

* **Technical Architecture:**



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

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1 | User Interface | Web UI and responsive Mobile Web | HTML, CSS, JavaScript, React.js |
| 2 | Application Logic-1 | User authentication & registration | Node.js / Express.js |
| 3 | Application Logic-2 | Property search, filtering, and listing | Node.js / Express.js |
| 4 | Application Logic-3 | Booking and payment module | Razorpay API + Node.js |
| 5 | Database | Stores user info, property listings, bookings | MongoDB (NoSQL) |
| 6 | Cloud Database | Cloud-hosted version for scaling | MongoDB Atlas |
| 7 | File Storage | Property images and documents | AWS S3 / Local Filesystem |
| 8 | External API-1 | Geolocation for property address mapping | Google Maps API |
| 9 | External API-2 | Payment gateway integration | Razorpay API |
| 10 | Machine Learning Model | Recommendation of listings based on user interest (optional feature) | Collaborative Filtering Model (Python) |
| 11 | Infrastructure | Cloud deployment of entire application | AWS EC2 / Docker / Kubernetes |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1 | Open-Source Frameworks | React.js, Node.js, Express.js | JavaScript (ME\_N\_ Stack) |
| 2 | Security Implementations | SHA-256 hashing for passwords, JWT tokens, HTTPS, Role-based access control | Bcrypt, JWT, HTTPS, Helmet.js |
| 3 | Scalable Architecture | Supports horizontal scaling, containerized microservices | Docker, Kubernetes, 3-tier architecture |
| 4 | Availability | Load balanced architecture, auto-scaling group | AWS ELB, Auto Scaling |
| 5 | Performance | CDN for static assets, caching for repeated search queries | Cloudflare CDN, Redis for caching |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/)

[**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture)

[**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)