Project Design Phase-II
Technology Stack (Architecture & Stack)
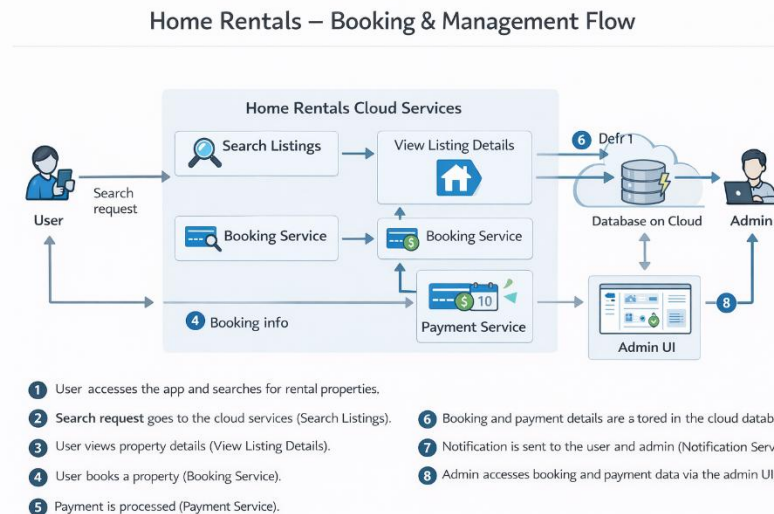
Date	22 February 2026
Team ID	LTVIP2026TMIDS42582
Project Name	Home-Rentals
Maximum Marks	4 Marks

Technical Architecture:

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

Example: Home-Rentals Application

Reference: https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/

Home Rentals — Booking & Management Flow

Guidelines:

1. **Include All Processes (Application Logic / Technology Blocks):**
   - Clearly represent each major process or technology block (e.g., user authentication, property search, booking, payment, notifications).
   - Use distinct shapes or icons for each process to enhance clarity.
2. **Provide Infrastructural Demarcation (Local / Cloud):**
   - Visually separate components running locally (e.g., client/mobile app, admin dashboard) from those hosted in the cloud (e.g., backend server, database, cloud services).
   - Use shaded areas or labeled sections to indicate "Local" and "Cloud" boundaries.

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application (Web UI, Mobile App)	HTML, CSS, JavaScript, React.js
2.	Application Logic-1	Logic for user authentication and registration	Node.js, Express.js
3.	Application Logic-2	Logic for property search and booking	Node.js, Express.js
4.	Application Logic-3	Logic for payment processing	Stripe API, Node.js
5.	Database	Data storage for users, listings, bookings	MongoDB (NoSQL)
6.	Cloud Database	Database Service on Cloud	MongoDB Atlas
7.	File Storage	Storage for images and documents	Local Filesystem, Cloud Storage (AWS S3)
8.	External API-1	Payment gateway integration	Stripe API
9.	External API-2	Email/SMS notification service	SendGrid API, Twilio API
10.	Machine Learning Model	Recommendation engine for listings (if applicable)	TensorFlow, Scikit-learn
11.	Infrastructure (Server / Cloud)	Application deployment on local/cloud systems	Local Server, Vercel (Client), AWS EC2 (Server), Docker, Kubernetes

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	List the open-source frameworks used	React.js, Node.js, Express.js, MongoDB
2.	Security Implementations	Security/access controls, encryption, firewalls, etc.	JWT, bcrypt, HTTPS, OWASP, IAM Controls

S.No	Characteristics	Description	Technology
3.	Scalable Architecture	Justify scalability (multi-tier, microservices, etc.)	REST API, Docker, Kubernetes
4.	Availability	Load balancers, distributed servers, cloud deployment	Nginx, AWS ELB, Vercel, MongoDB Atlas
5.	Performance	Caching, CDN, request handling, optimization	Redis, Cloudflare CDN, MongoDB Indexes

References:

https://c4model.com/

https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/

https://www.ibm.com/cloud/architecture

https://aws.amazon.com/architecture

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