Project Design Phase-II
Technology Stack (Architecture & Stack)

Date	31 January 3035
Team ID	LTVIP2026TMIDS88461
Project Name	DocSpot — Seamless Appointment Booking
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2
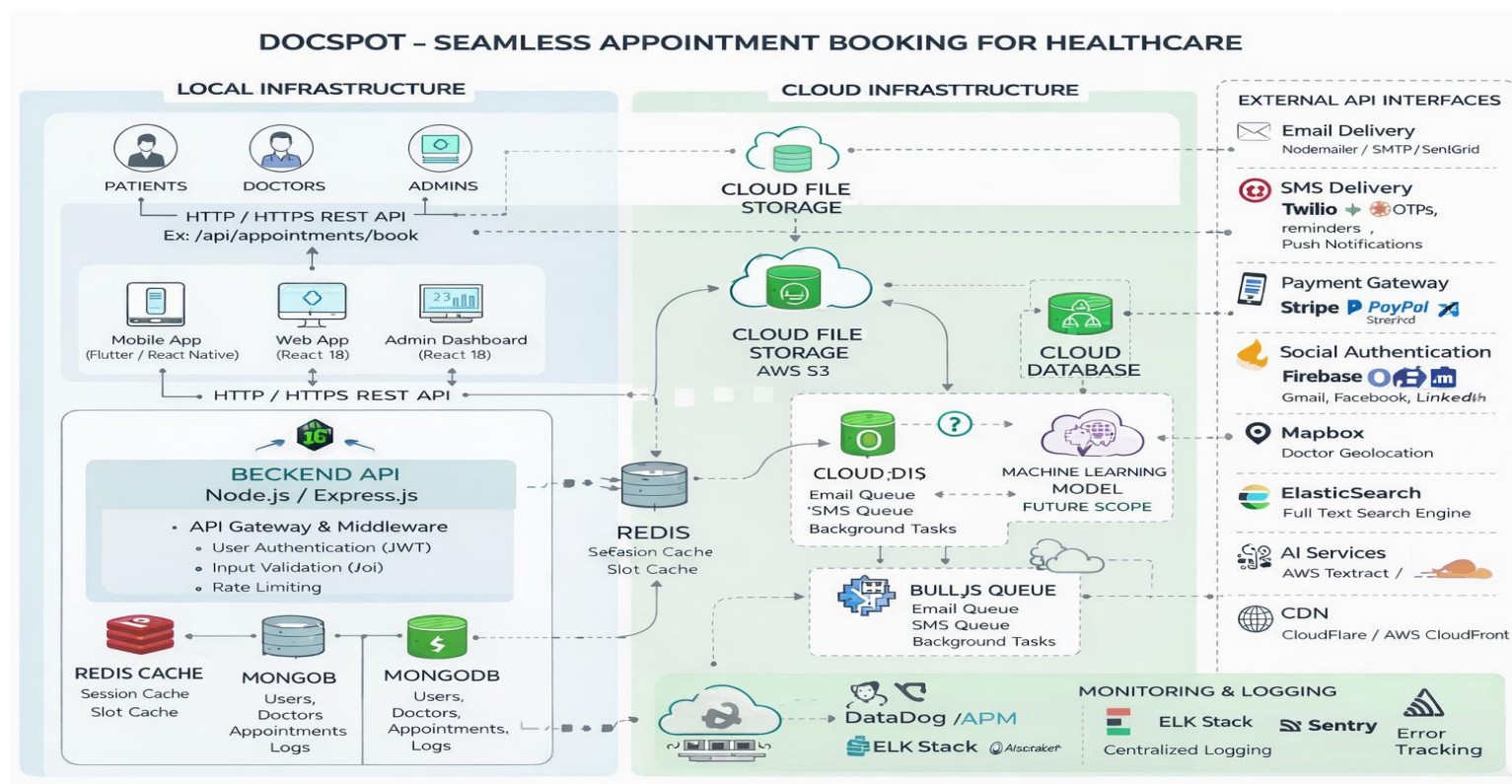
Table-1 : Components & Technologies:

S.No	Component	Description	Technology Used
1	User Interface – Web	Web interface for patients, doctors, and admin dashboard.	React 18, HTML5, CSS3, Bootstrap / Ant Design, Axios
2	User Interface – Mobile	Mobile app for appointment booking and management.	React Native / Flutter
3	API Gateway & Middleware	Handles routing, authentication, validation, and request filtering.	Express.js, CORS, Helmet.js, Morgan
4	User Authentication Service	Manages registration, login, password reset, JWT tokens.	Node.js, JWT, bcryptjs
5	Doctor Management Service	Handles doctor registration, approval workflow, availability management.	Node.js, Mongoose
6	Appointment Management Service	Core booking logic, slot validation, rescheduling, cancellation.	Node.js, Express.js
7	Notification Service	Sends email confirmations, reminders, and alerts.	Bull Queue, Redis, Nodemailer, SendGrid
8	SMS & Push Notification Service	Sends OTP and reminders via SMS and mobile push.	Twilio, Firebase Cloud Messaging
9	Document Management	Secure upload and storage of prescriptions and reports.	Multer, Local Storage / AWS S3
10	Database	Stores users, doctors, appointments, audit logs.	MongoDB (Atlas / Local), Mongoose
11	Cache Layer	Improves performance and stores temporary session data.	Redis
12	Message Queue	Handles asynchronous tasks like email & SMS processing.	Bull.js
13	File Storage	Stores uploaded medical documents securely.	Local FS / AWS S3
14	External APIs	Social login and third-party integrations.	Google OAuth, Facebook API, LinkedIn API
15	Infrastructure	Deployment environment for hosting backend and frontend.	AWS EC2 / Local Linux Server / Docker / Nginx

Table-2: Application Characteristics:

S.No	Characteristics	Description	Implementation
1	Open-Source Frameworks	Uses modern open-source technologies.	React, Express.js, MongoDB, Redis
2	Security	Protects user data and prevents unauthorized access.	HTTPS, JWT Authentication, bcrypt hashing, RBAC
3	Scalability	Supports increasing users and appointment load.	Auto-scaling (Cloud), Redis caching, Load balancer
4	Availability	Ensures system uptime and reliability.	Cloud deployment, Multi-instance setup
5	Performance	Fast API responses and optimized queries.	Indexing, Caching, CDN
6	Backup & Recovery	Prevents data loss with regular backups.	MongoDB automated backups
7	Monitoring & Logging	Tracks system health and errors.	Winston logs, CloudWatch, ELK Stack
8	Compliance	Ensures healthcare data privacy compliance.	GDPR-ready design, Encrypted storage
9	Deployment	Supports both local and cloud environments.	Docker, AWS, Nginx