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

Date	28 MAY 2025
Team ID	LTVIP2025TMID55905
Project Name BookNest: Where stories nestle	
Maximum Marks	4 Marks

### **Technical Architecture:**

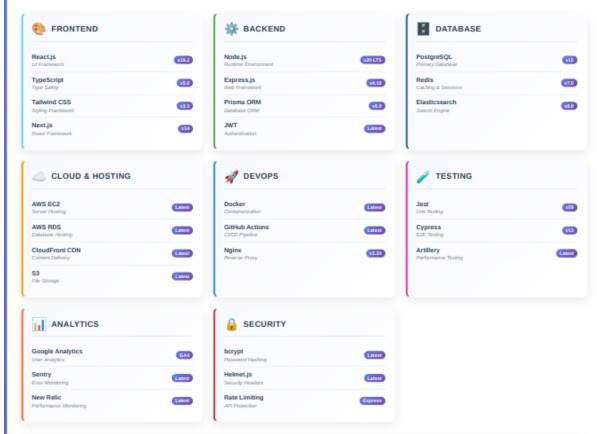
The Deliverable shall include the architectural diagram as below and the information as per the table 1 & table 2

**Example: Order processing during pandemics for offline mode** 

Reference: <a href="https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/">https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/</a>

## Book Nest Technology Stack

Comprehensive Technology Architecture & Implementation Plan



TECHNOLOGY	CATEGORY	PURPOSE	JUSTIFICATION	COST
React.js + TypeScript	Frontend	User Interface Development	Industry standard, large community, excellent for complex UIs with type safety	Free
Node.js + Express	Backend	Server-side Development	JavaScript ecosystem consistency, high performance, extensive package ecosystem	Free
PostgreSQL	Database	Data Storage & Management	ACID compliance, excellent performance, JSON support, strong community	Free
AWS Cloud Services	Infrastructure	Hosting & Scaling	Industry leader, comprehensive services, excellent scaling capabilities	Pay-as-use
Redis	Caching	Performance Optimization	In-memory speed, session management, reduces database load	Free
Elasticsearch	Search	Advanced Search Features	Full-text search, faceted search, excellent for book catalog searching	Licensing
Docker + CUCD	DevOps	Deployment & Automation	Consistent environments, automated deployments, scalability	Free

S.No	Component	Description	Technology
1.	User	How user interacts with application	HTML, CSS,ReactJS+Vite/Bootstrap, CSS etc.
	Interface	Web UI	
2.	Application	Logic for a process in the application	JavaScript.
	Logic-1		
3.	Database	Data Type, Configurations etc.	MongoDB, Mongoose.
4.	File Storage	File storage requirements	MongoDB Cluster storage.
5.	External	Purpose of External API used in the application	
	API-1		
6.		Purpose of External API used in the application	
	API-2		

## Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology	
Open-Source Frameworks		Frontend uses React (via Vite), Tailwind CSS, Bootstrap for UI components, Axios for HTTP requests. Backend is built using Node.js with Express.	React, Vite, Tailwind CSS, Bootstrap, Axios, Node.js, Express.js	
2.	Security Implementations	Passwords are encrypted using bcrypt. CORS is implemented for secure cross-origin communication. Input validations prevent injection attacks.	bcrypt, CORS, express-validator, Helmet (optional)	
3.	Scalable Architecture	Follows a modular architecture separating frontend, backend, and database (3-tier). Can be containerized using Docker for scaling.	Node.js Microservices (optional),	
4. Availability		Application can be deployed on cloud platforms (e.g., Heroku, Render, AWS) with horizontal scaling. Load balancers can be used if demand increases.	Cloud platforms (Render, AWS, etc.), Nginx (optional)	

5.	Performance	Efficient API calls with Axios, caching static content	Axios, MongoDB, CDN (e.g.,
		using CDN. MongoDB handles highvolume	Cloudflare), Compression
		reads/writes efficiently.	

### 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