

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

Date	31 January 3035
Team ID	SWTID1743696165
Project Name	Bookie Nest: Book management website
Maximum Marks	4 Marks

Technical Architecture:

Example: The application is an online book store platform named **Bookie Nest** that allows users to browse a wide range of books, add them to their cart, and place orders. Users can also manage their profiles and view order history. Administrators can manage book listings, categories, inventory, and orders through a dedicated admin dashboard.



Guidelines:

- Include all processes as technology blocks (Frontend & Backend).
- Show infrastructure separation – Local & Cloud.
- Indicate external interfaces – Payment API, Books API, etc.
- Mention data storage – MongoDB, Cloud Storage.
- Show machine learning model interface (if used).
- Define architecture layers – UI, Logic, Database.
- Highlight security – JWT, bcryptjs.
- Provide booking process flow – Search → Select → Pay → Confirm.
- Specify communication – API calls, Notifications.
- Show deployment setup – Hosting servers & Cloud DB.

Table-1: Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Add Books	This component allows administrators to add new Books to the system. It includes a form for entering Books details such as name, description, genres, release date, runtime, and certification. It also supports editing existing Books details.	React (for building the component) Axios (for API calls to add or edit Books) React Router (for navigation) SWR (for data fetching and caching) React Toastify (for notifications) SCSS (for styling)
3.	Add Books to Cart	This component enables administrators to add new Books to the cart. It includes a form for selecting a Books.	<ul style="list-style-type: none">○ React (for building the component)○ Axios (for API calls to add books)○ React Router (for navigation)○ React Toastify (for notifications)○ SCSS (for styling)
4.	Admin	The admin dashboard component provides an interface for managing Books, and other administrative tasks. It includes options to delete or add books and navigate to other admin functionalities.	<ul style="list-style-type: none">• React (for building the component)• Axios (for API calls)• React Router (for navigation)• React Toastify (for notifications)• SCSS (for styling)
5.	Admin Books	This component displays a list of Books available in the system. It allows administrators to edit or delete Books and provides a visual interface for managing Books data.	<ul style="list-style-type: none">• React (for building the component)• Axios (for API calls)• React Toastify (for notifications)• SCSS (for styling)
6.	Cart	This component displays the user's list of Books Which they want to purchase.	React, Axios, SCSS

7.	Checkout	The checkout component handles the payment process for purchasing Books. It displays payment confirmation and allows users to claim their purchase.	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • React Router (for navigation) • SCSS (for styling)
8.	Favourite Books	This component displays a list of the user's favourite Books. Users can view and manage their saved Books.	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • React Toastify (for notifications) • SCSS (for styling)
9.	Home	The home page component serves as the landing page for the application, showcasing featured Books.	<ul style="list-style-type: none"> • React (for building the component) • SCSS (for styling)
10.	Login	This component provides a login interface for users and administrators to access their respective dashboards.	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • React Router (for navigation) • SCSS (for styling)
11.	Books Details	This component displays detailed information about a specific Books, including its description, genres, release date, runtime, and user reviews. It also allows users to like or dislike the Books.	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • React Router (for navigation) • React Toastify (for notifications) • SCSS (for styling)
12.	Profile	The profile component allows users to view and edit their personal details	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • SCSS (for styling)
13.	Register	This component provides a registration interface for new users to create an account.	<ul style="list-style-type: none"> • React (for building the component) • Axios (for API calls) • SCSS (for styling)
14.	Search	This component allows users to search for Books and genre.	React (for building the component) Axios (for API calls) SCSS (for styling)

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	The application uses open-source frameworks for development and styling.	React, React Router, SCSS, Axios
2	Security Implementations	Implements JWT for authentication, input validation, and secure API calls.	JWT, HTTPS, Form Validation, OWASP Guidelines
3	Scalable Architecture	Built with a scalable architecture to handle growing user demands.	Component-Based Architecture, RESTful APIs
4	Availability	Ensures high availability with backend APIs and responsive design.	Load Balancers, Distributed Backend Servers
5	Performance	Optimized with caching, lazy loading, and efficient API calls.	SWR (Caching), CDN, Axios, React Optimization