



CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE • INDIA

**DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING**

L&T

CIA 2&3 – Angular Coding Qs -Theory + Practical

**B. Tech – Computer Science and Engineering
School of Engineering and Technology,**

CHRIST (Deemed to be University),

Kumbalagodu, Bengaluru-560 074

Jan 2026

JavaScript Frontend Framework Project

Event Management and Booking System

Submitted By:

Team Member: Dinesh Babu R S

Roll Number: 2460360

College Email: dinesh.babu@btech.christuniversity.in

Team Member: Prisha S

Roll Number: 2460424

College Email: prisha.s@btech.christuniversity.in

Team Member: Jashwanth B S

Roll Number: 2460374

College Email: jashwanth.bs@btech.christuniversity.in

Team Member: Aishwarya S

Roll Number: 2460315

College Email: aishwartya.s@btech.christuniversity.in

Course: Advanced Angular Frontend JavaScript

Instructor: Mr. Balagopal / Mr. Athaulla

University: Christ University

Department: Btech in Computer Science Engineering

INDEX:

S.No	Section	Page No.
1	Abstract	3
2	Objectives	3
3	Scope of the Project	3
4	Tools & Technologies Used	4
5	HTML Structure Overview(With JavaScript/Angular Integration)	4
6	CSS Styling Strategy(With JavaScript/Angular Integration)	5
7	Key Features	5-6
8	Challenges Faced & Solutions	6
9	Outcome	6
10	Future Enhancements	6-7
11	Full Code	7-8
12	Screenshots of Final Output	8-9
13	Conclusion	9
14	References	9

Abstract:

The Event Management and Booking System is a modern Single Page Application (SPA) developed using Angular and TypeScript to simulate a real-world event discovery and ticket booking platform. The application enables users to browse upcoming events, view detailed event information, register or book tickets, and manage their bookings through an interactive and responsive user interface. The system leverages Angular's component-based architecture, routing, dependency injection, reactive programming with RxJS, and Angular Material UI to deliver a scalable and maintainable front-end solution.

Mock backend integration using JSON Server or static JSON files is employed to simulate real-time data retrieval and booking operations. The project demonstrates industry-standard Angular development practices and serves as a strong learning and portfolio project for modern front-end application development.

Objectives:

The primary objectives of the Event Management and Booking System are:

- To design and develop a fully functional Angular-based SPA.
- To apply TypeScript concepts such as interfaces, strong typing, and object-oriented design.
- To implement event browsing, booking, and management functionalities.
- To demonstrate Angular routing, services, and dependency injection.
- To build validated forms using both template-driven and reactive approaches.
- To create a responsive and professional UI using Angular Material.
- To simulate backend interactions using mock JSON data.

Scope of the Project:

Included Scope

- Viewing and filtering events
- Event detail display
- Ticket booking and registration
- Booking management
- Client-side validation
- Mock backend data integration

Excluded Scope

- Real payment gateway integration
- Live authentication servers
- Production database deployment

The scope is intentionally limited to focus on front-end architecture and learning objectives.

Tools & Technologies Used:

- Framework: Angular (v17 or latest)
- Language: TypeScript
- UI Library: Angular Material
- Markup & Styling: HTML5, CSS3
- State Management: RxJS Observables
- Mock Backend: JSON Server / Static JSON Files
- Development Tools:
 - Node.js
 - Angular CLI
 - Visual Studio Code
- Version Control: Git & GitHub

Application Architecture:

The application follows Angular's layered architecture:

- Components: Handle UI and user interactions
- Services: Manage data and business logic
- Models: Define data structures using TypeScript interfaces
- Routing Module: Controls navigation
- Mock Backend: Supplies event and booking data

This architecture ensures modularity, reusability, and scalability.

HTML Structure Overview:

- **App Component**
 - Root layout with <app-navbar> and <router-outlet>
- **Navbar Component**
 - Angular Material Toolbar
 - Navigation links to Events and Bookings
- **Event List Component**
 - Uses *ngFor to render event cards
 - Filters applied via custom pipes
 - **Event Detail Component**
 - Loads event details using route parameters
 - Displays schedules and descriptions
 - **Booking Form Component**
 - Uses reactive forms
 - Handles ticket booking and validation

CSS Styling Strategy:

- Component-scoped CSS for modular styling
- Angular Material themes for consistency
- Flexbox and grid layouts for responsiveness
- Conditional styling using [ngClass] and [ngStyle]
- Visual indicators for featured or sold-out events

Key Features:

- Event listing with category and date filtering
- Detailed event view with schedule information
- Ticket booking and registration
- Booking history management

- Angular Material dialogs for confirmations
- Snackbar notifications for success and error messages
- Responsive design for mobile and desktop
- Route protection for booking access (mock authentication)

Challenges Faced & Solutions:

Data Synchronization

Challenge: Sharing event and booking data across components

Solution: Centralized services with Observables

Form Validation

Challenge: Handling user input errors

Solution: Reactive forms with built-in validators

Routing Complexity

Challenge: Managing dynamic routes

Solution: Route parameters and guards

UI Responsiveness

Challenge: Device compatibility

Solution: Angular Material responsive layouts

Outcome:

The project resulted in a fully functional Angular-based Event Management and Booking System that demonstrates modern front-end development practices. The application offers a smooth user experience, clean architecture, and professional UI design. It significantly enhanced understanding of Angular concepts such as services, routing, forms, observables, and Material UI integration. The project also serves as a strong portfolio application suitable for interviews and academic evaluation.

Future Enhancements:

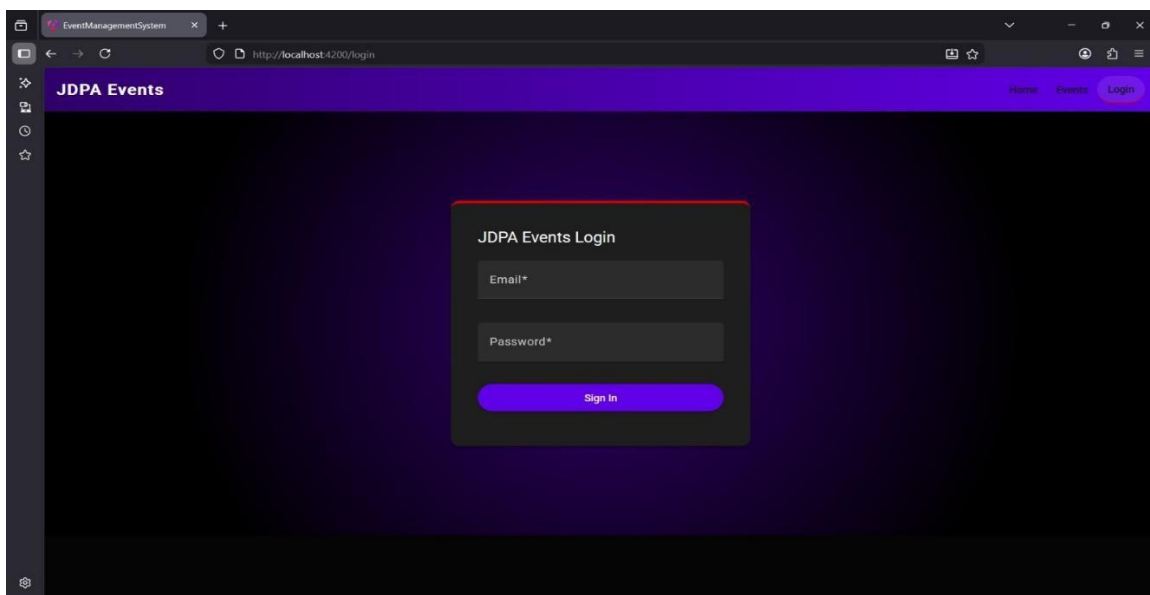
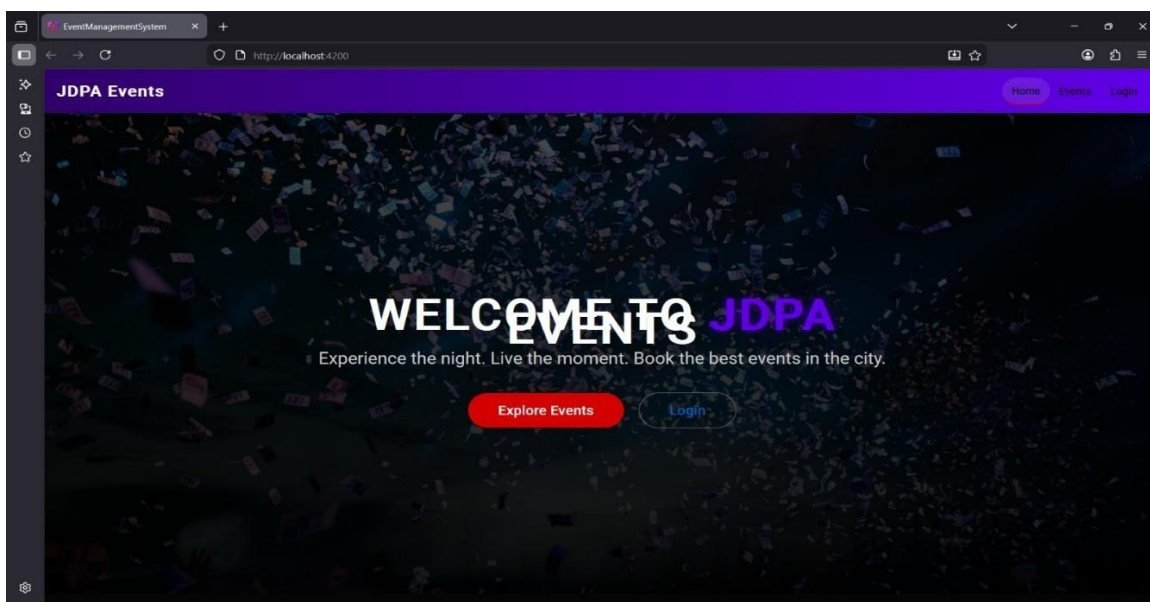
- Integration with real backend services (Node.js / Spring Boot)
- Secure authentication and role-based access
- Payment gateway integration

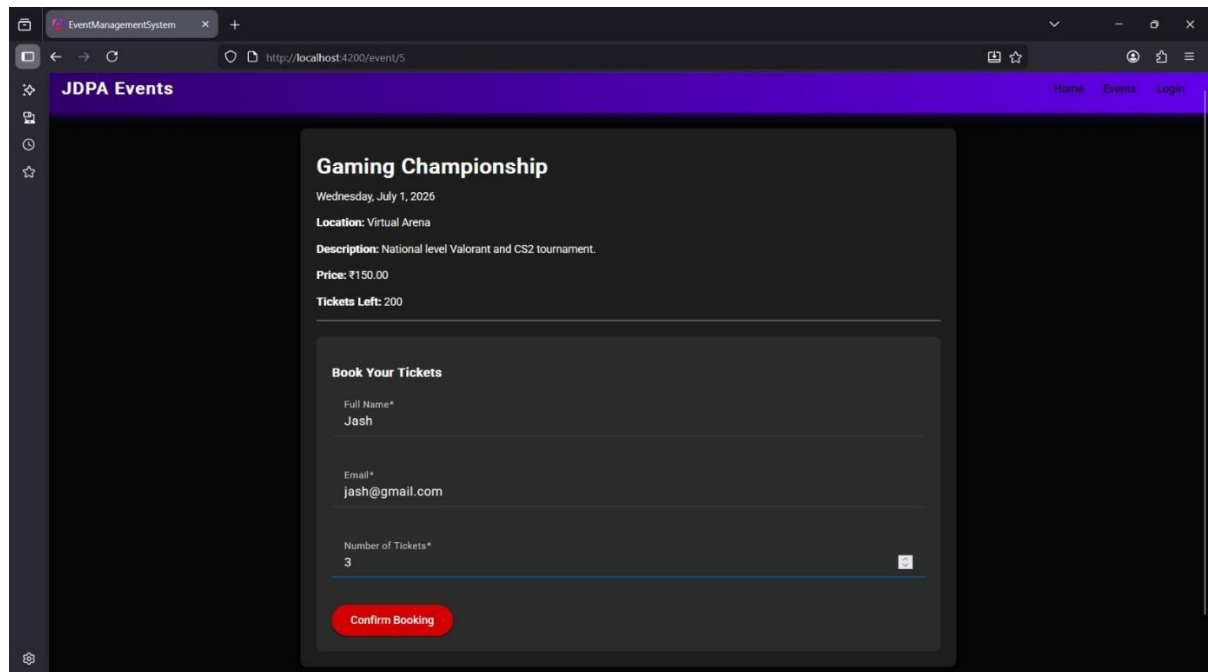
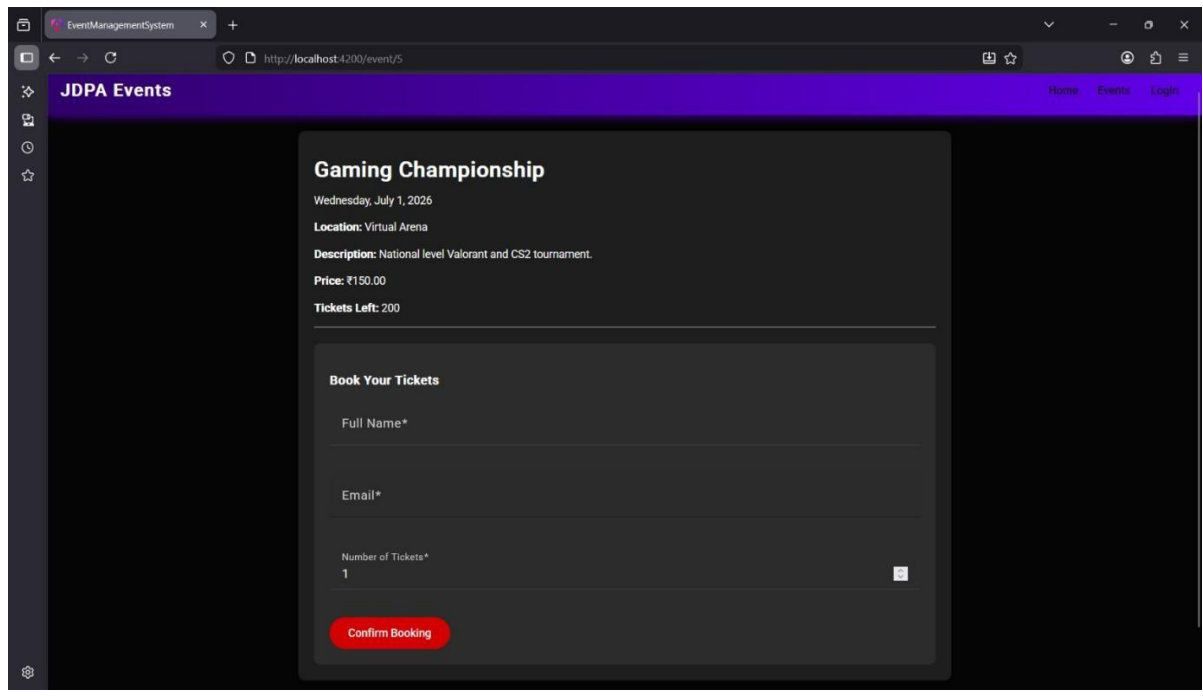
- Real-time event updates using WebSockets
- Admin dashboard for event management
- Cloud deployment (AWS / Firebase)

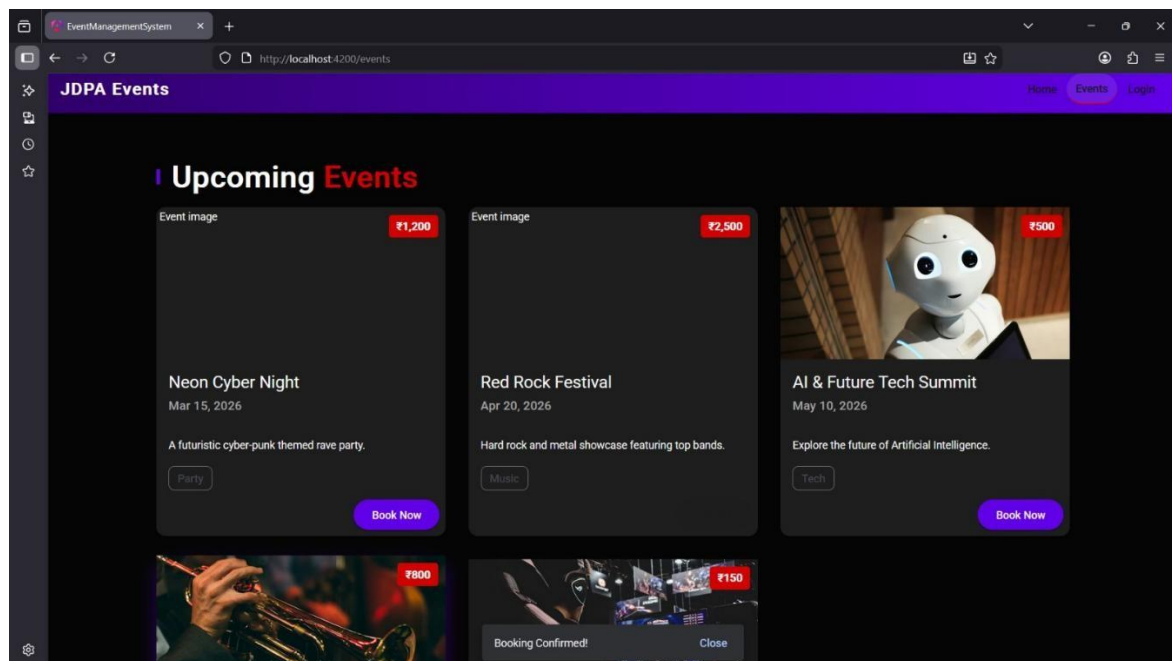
Full Code:

Github Link: <https://github.com/Devillance079/event-management-system>

Screenshots of Final Output:







Conclusion:

The Event Management and Booking System effectively bridges theoretical learning and practical implementation of Angular development. By incorporating modular architecture, robust UI design, and simulated backend interactions, the project demonstrates industry-relevant skills and best practices. It provides a strong foundation for extending the system into a full-scale production application.

References:

1. Angular Official Documentation <https://angular.io/docs>
2. TypeScript Documentation <https://www.typescriptlang.org/docs>
3. Angular Material Documentation <https://material.angular.io>
4. RxJS Documentation <https://rxjs.dev>
5. JSON Server Documentation <https://github.com/typicode/json-server>
6. MDN Web Docs <https://developer.mozilla.org>