

# Smart Event Booking System

---

A modern, full-stack MERN (MySQL) event booking platform with real-time seat availability, QR tickets, animations, and PWA support.

🔗 **Live Link:** <https://event-booking-app.vercel.app>

---

## Tech Stack

- **Frontend:** React.js, Tailwind CSS, Framer Motion, React Router, Socket.io-client
  - **Backend:** Node.js, Express.js, Socket.io, MySQL2
  - **Database:** MySQL (AlwaysData)
  - **Deployment:**
    - Frontend → Vercel
    - Backend → Render
    - Database → AlwaysData
- 

## Project Structure

```
event-booking/  
├── frontend/           # React app  
├── backend/            # Node.js + Express + APIs  
├── event_booking.sql/  # SQL schema & seed files  
└── Documentation(ReadMe) # Documentation
```

---

## Setup Guide

### 1. Clone Repo

```
git clone https://github.com/shibbu04/EventBook.git  
cd EventBook
```

---

### 2. Database (AlwaysData or Local MySQL)

```
-- If using local MySQL  
CREATE DATABASE event_booking;
```

```
-- Import schema
mysql -u root -p event_booking < database/event_booking.sql
```

Then, create a temporary admin and sample users:

```
cd backend
npm install
node scripts/createAdmin.js
```

---

### 3. Environment Variables Setup

#### ◇ Backend .env

Create a `.env` file inside `backend/`:

```
# Environment
NODE_ENV=development
PORT=5000

# Database
DB_HOST=localhost
DB_USER=root
DB_PASSWORD=pass
DB_NAME=event_booking
DB_PORT=3306

# JWT
JWT_SECRET=event-book

# Frontend URL
FRONTEND_URL=http://localhost:3000

# Email (optional)
EMAIL_HOST=smtp.gmail.com
EMAIL_PORT=587
EMAIL_USER=your_email@gmail.com
EMAIL_PASS=your_app_password

# Cloudinary (for images)
CLOUDINARY_CLOUD_NAME=
CLOUDINARY_API_KEY=
CLOUDINARY_API_SECRET=
```

---

#### ◇ Frontend .env

Create a `.env` file inside `frontend/`:

```
REACT_APP_BACKEND_URL=http://localhost:5000
```

---

## 4. Run Backend

```
cd backend
npm install
npm start
```

Runs on → <http://localhost:5000>

---

## 5. Run Frontend

```
cd frontend
npm install
npm run dev
```

Runs on → <http://localhost:3000>

---

## ✦ Features

- Landing page with animations + dark/light mode
  - Event search & filters
  - Real-time seat availability (WebSockets)
  - Interactive booking flow + QR code tickets
  - Admin dashboard (CRUD events, revenue stats)
  - PWA support
- 

## 🌐 API Endpoints

### Events

- `GET /api/events` → list events
- `GET /api/events/:id` → single event
- `POST /api/events` → create event (admin)
- `PUT /api/events/:id` → update event (admin)
- `DELETE /api/events/:id` → delete event (admin)

### Bookings

- `POST /api/bookings` → create booking
- `GET /api/bookings` → list bookings

---

## Remaining Work (if needed)

- Payment gateway integration
  - Email notifications
  - Multi-language support
- 

## Developer

**Name:** Shivam Singh **Role:** Full Stack Developer ✉ [shivamsingh57680@gmail.com](mailto:shivamsingh57680@gmail.com) 💻 [GitHub: @shibbu04](#)  
🌐 [Portfolio](#)