**events.json (12 Events Data)**

json

CopyEdit

[

{

"id": 1, "name": "Music Concert", "date": "2025-03-20",

"venue": "City Auditorium", "image": "assets/images/concert.jpg"

},

{

"id": 2, "name": "Tech Conference", "date": "2025-04-10",

"venue": "Tech Park Hall", "image": "assets/images/tech.jpg"

},

{

"id": 3, "name": "Art Exhibition", "date": "2025-05-05",

"venue": "Art Gallery", "image": "assets/images/art.jpg"

},

{

"id": 4, "name": "Food Festival", "date": "2025-06-15",

"venue": "City Square", "image": "assets/images/food.jpg"

},

{

"id": 5, "name": "Dance Show", "date": "2025-07-10",

"venue": "Grand Theatre", "image": "assets/images/dance.jpg"

},

{

"id": 6, "name": "Business Summit", "date": "2025-08-05",

"venue": "Business Park", "image": "assets/images/business.jpg"

},

{

"id": 7, "name": "Startup Meetup", "date": "2025-09-12",

"venue": "Co-working Space", "image": "assets/images/startup.jpg"

},

{

"id": 8, "name": "Gaming Expo", "date": "2025-10-30",

"venue": "Expo Center", "image": "assets/images/gaming.jpg"

},

{

"id": 9, "name": "Science Fair", "date": "2025-11-20",

"venue": "University Campus", "image": "assets/images/science.jpg"

},

{

"id": 10, "name": "Book Fair", "date": "2025-12-05",

"venue": "Library Hall", "image": "assets/images/book.jpg"

},

{

"id": 11, "name": "Film Festival", "date": "2026-01-10",

"venue": "Cinema Complex", "image": "assets/images/film.jpg"

},

{

"id": 12, "name": "Charity Run", "date": "2026-02-25",

"venue": "City Park", "image": "assets/images/charity.jpg"

}

]

## ****index.html (Landing Page with Sliders)****

html

CopyEdit

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Event Management</title>

<link rel="stylesheet" href="assets/styles.css">

</head>

<body>

<header>

<h1>Event Management System</h1>

<nav>

<a href="login.html">Login</a>

<a href="user.html">User Dashboard</a>

<a href="admin.html">Admin Dashboard</a>

</nav>

</header>

<section id="slider">

<div class="slide">

<img src="assets/images/concert.jpg" alt="Music Concert">

<div class="caption">Join Our Music Concert</div>

</div>

<div class="slide">

<img src="assets/images/tech.jpg" alt="Tech Conference">

<div class="caption">Attend the Tech Conference</div>

</div>

<div class="slide">

<img src="assets/images/food.jpg" alt="Food Festival">

<div class="caption">Explore the Food Festival</div>

</div>

</section>

<script src="assets/script.js"></script>

</body>

</html>

### **📌** login.html **(Login & Signup Page)**

html

CopyEdit

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Login & Signup</title>

<link rel="stylesheet" href="assets/styles.css">

</head>

<body>

<div class="login-container">

<h2>Login</h2>

<select id="role">

<option value="user">User</option>

<option value="admin">Admin</option>

</select>

<input type="text" id="login-username" placeholder="Username">

<input type="password" id="login-password" placeholder="Password">

<button onclick="login()">Login</button>

<p>Don't have an account? <a href="#" onclick="toggleForms()">Sign Up</a></p>

</div>

<div class="signup-container" style="display: none;">

<h2>Sign Up</h2>

<select id="signup-role">

<option value="user">User</option>

<option value="admin">Admin</option>

</select>

<input type="text" id="signup-username" placeholder="Username">

<input type="password" id="signup-password" placeholder="Password">

<button onclick="signup()">Sign Up</button>

<p>Already have an account? <a href="#" onclick="toggleForms()">Login</a></p>

</div>

<script src="assets/script.js"></script>

</body>

</html>

### **📌** script.js **(JavaScript for Login & Signup)**

javascript

CopyEdit

// Function to toggle between login and signup forms

function toggleForms() {

document.querySelector(".login-container").style.display =

document.querySelector(".login-container").style.display === "none" ? "block" : "none";

document.querySelector(".signup-container").style.display =

document.querySelector(".signup-container").style.display === "none" ? "block" : "none";

}

// Signup function

function signup() {

let username = document.getElementById("signup-username").value;

let password = document.getElementById("signup-password").value;

let role = document.getElementById("signup-role").value;

if (username === "" || password === "") {

alert("Please fill all fields");

return;

}

let users = JSON.parse(localStorage.getItem("users")) || [];

let existingUser = users.find(user => user.username === username);

if (existingUser) {

alert("Username already exists!");

return;

}

users.push({ username, password, role });

localStorage.setItem("users", JSON.stringify(users));

alert("Signup successful! Please login.");

toggleForms();

}

// Login function

function login() {

let username = document.getElementById("login-username").value;

let password = document.getElementById("login-password").value;

let role = document.getElementById("role").value;

let users = JSON.parse(localStorage.getItem("users")) || [];

let validUser = users.find(user => user.username === username && user.password === password && user.role === role);

if (validUser) {

alert("Login successful!");

if (role === "admin") {

window.location.href = "admin-dashboard.html"; // Redirect to admin dashboard

} else {

window.location.href = "user-dashboard.html"; // Redirect to user dashboard

}

} else {

alert("Invalid credentials! Please try again.");

}

}

document.addEventListener("DOMContentLoaded", () => {

let slides = document.querySelectorAll(".slide");

let index = 0;

function showSlide() {

slides.forEach(slide => slide.style.display = "none");

slides[index].style.display = "block";

index = (index + 1) % slides.length;

setTimeout(showSlide, 3000);

}

showSlide();

});

### **📌** styles.css **(Basic Styling for Forms)**

css

CopyEdit

body {

font-family: Arial, sans-serif;

text-align: center;

background-color: #f4f4f4;

}

.login-container, .signup-container {

width: 300px;

margin: 100px auto;

padding: 20px;

background: white;

box-shadow: 0px 0px 10px 0px gray;

border-radius: 10px;

}

input, select {

width: 90%;

padding: 10px;

margin: 10px 0;

display: block;

border: 1px solid #ddd;

border-radius: 5px;

}

button {

width: 100%;

padding: 10px;

background: #28a745;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

}

button:hover {

background: #218838;

}

p {

margin-top: 10px;

}

a {

color: #007bff;

cursor: pointer;

text-decoration: none;

}

## ****styles.css (Beautiful UI)****

css

CopyEdit

body {

font-family: Arial, sans-serif;

background: url('images/background.jpg') no-repeat center center/cover;

color: white;

text-align: center;

padding: 20px;

}

header {

background: rgba(0, 0, 0, 0.7);

padding: 15px;

}

nav a {

color: white;

margin: 0 15px;

text-decoration: none;

font-weight: bold;

}

#slider {

width: 80%;

margin: auto;

position: relative;

}

.slide {

display: none;

}

.slide img {

width: 100%;

border-radius: 10px;

}

.caption {

position: absolute;

bottom: 10px;

left: 50%;

transform: translateX(-50%);

background: rgba(0, 0, 0, 0.6);

color: white;

padding: 10px;

border-radius: 5px;

}

Here is the **complete HTML & JavaScript** for a **User Dashboard**, **Event Listing Page**, **Event Booking Page**, and **Bookings Page** with a **Cancel Booking** feature.

### **📌 Features Implemented:**

✅ **User Dashboard** – View available events  
✅ **View Events** – Click an event to go to the booking page  
✅ **Book Events** – Simple booking system  
✅ **View Bookings** – Shows all booked events  
✅ **Cancel Booking** – Remove a booked event

## ****1️⃣**** user-dashboard.html ****(User Dashboard - View Events)****

This page displays a list of events. Clicking an event redirects to the booking page.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>User Dashboard</title>

<link rel="stylesheet" href="assets/styles.css">

</head>

<body>

<h1>Welcome, User!</h1>

<h2>Available Events</h2>

<div id="event-list"></div>

<br>

<a href="view-bookings.html">View My Bookings</a> |

<a href="login.html">Logout</a>

<script src="assets/script.js"></script>

</body>

</html>

## ****2️⃣**** book-event.html ****(Event Booking Page)****

This page allows users to book an event.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Book Event</title>

<link rel="stylesheet" href="assets/styles.css">

</head>

<body>

<h1>Book Event</h1>

<div id="event-details"></div>

<button onclick="bookEvent()">Confirm Booking</button>

<br><br>

<a href="user-dashboard.html">Back to Dashboard</a>

<script src="assets/script.js"></script>

</body>

</html>

## ****3️⃣**** view-bookings.html ****(View & Cancel Bookings)****

This page displays all booked events with an option to cancel.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>My Bookings</title>

<link rel="stylesheet" href="assets/styles.css">

</head>

<body>

<h1>My Bookings</h1>

<div id="bookings-list"></div>

<br>

<a href="user-dashboard.html">Back to Dashboard</a>

<script src="assets/script.js"></script>

</body>

</html>

## ****4️⃣**** script.js ****(JavaScript for Events, Booking, and Canceling)****

document.addEventListener("DOMContentLoaded", function() {

if (window.location.pathname.includes("user-dashboard.html")) {

loadEvents();

} else if (window.location.pathname.includes("book-event.html")) {

showEventDetails();

} else if (window.location.pathname.includes("view-bookings.html")) {

loadBookings();

}

});

// Sample event data

const events = [

{ id: 1, name: "Tech Conference 2024", date: "2024-04-15", venue: "Hall A" },

{ id: 2, name: "Music Festival", date: "2024-05-20", venue: "Open Ground" },

{ id: 3, name: "Startup Pitching", date: "2024-06-10", venue: "Auditorium" }

];

// Function to display events in user dashboard

function loadEvents() {

let eventList = document.getElementById("event-list");

eventList.innerHTML = "";

events.forEach(event => {

let eventItem = document.createElement("div");

eventItem.innerHTML = `

<h3>${event.name}</h3>

<p>Date: ${event.date}</p>

<p>Venue: ${event.venue}</p>

<button onclick="redirectToBooking(${event.id})">Book Now</button>

<hr>

`;

eventList.appendChild(eventItem);

});

}

// Function to redirect to booking page

function redirectToBooking(eventId) {

localStorage.setItem("selectedEvent", JSON.stringify(events.find(e => e.id === eventId)));

window.location.href = "book-event.html";

}

// Function to show event details on booking page

function showEventDetails() {

let event = JSON.parse(localStorage.getItem("selectedEvent"));

let eventDetails = document.getElementById("event-details");

if (event) {

eventDetails.innerHTML = `

<h3>${event.name}</h3>

<p>Date: ${event.date}</p>

<p>Venue: ${event.venue}</p>

`;

} else {

eventDetails.innerHTML = "<p>No event selected.</p>";

}

}

// Function to book an event

function bookEvent() {

let event = JSON.parse(localStorage.getItem("selectedEvent"));

let bookings = JSON.parse(localStorage.getItem("bookings")) || [];

if (event) {

bookings.push(event);

localStorage.setItem("bookings", JSON.stringify(bookings));

alert("Booking Confirmed!");

window.location.href = "view-bookings.html";

} else {

alert("No event selected.");

}

}

// Function to display booked events

function loadBookings() {

let bookings = JSON.parse(localStorage.getItem("bookings")) || [];

let bookingList = document.getElementById("bookings-list");

bookingList.innerHTML = "";

if (bookings.length === 0) {

bookingList.innerHTML = "<p>No bookings yet.</p>";

return;

}

bookings.forEach((event, index) => {

let bookingItem = document.createElement("div");

bookingItem.innerHTML = `

<h3>${event.name}</h3>

<p>Date: ${event.date}</p>

<p>Venue: ${event.venue}</p>

<button onclick="cancelBooking(${index})">Cancel Booking</button>

<hr>

`;

bookingList.appendChild(bookingItem);

});

}

// Function to cancel a booking

function cancelBooking(index) {

let bookings = JSON.parse(localStorage.getItem("bookings")) || [];

bookings.splice(index, 1);

localStorage.setItem("bookings", JSON.stringify(bookings));

alert("Booking Canceled!");

loadBookings();

}

## ****5️⃣**** styles.css ****(Basic Styling)****

body {

font-family: Arial, sans-serif;

text-align: center;

background-color: #f4f4f4;

}

h1, h2 {

color: #333;

}

#event-list, #bookings-list {

width: 50%;

margin: auto;

background: white;

padding: 20px;

box-shadow: 0px 0px 10px 0px gray;

border-radius: 10px;

}

button {

padding: 10px;

background: #28a745;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

margin: 10px;

}

button:hover {

background: #218838;

}

a {

color: #007bff;

text-decoration: none;

font-weight: bold;

}

### **✅ How It Works:**

1. **User Dashboard** displays available events with a "Book Now" button.
2. Clicking "Book Now" redirects to the **Event Booking Page**.
3. The user confirms the booking, which gets saved in **Local Storage**.
4. **View Bookings Page** lists all bookings, with a "Cancel Booking" button.

### **🚀 Next Steps (Optional Enhancements)**

🔹 Use a **backend (Node.js, PHP) with a database** for real bookings.  
🔹 Add a **booking confirmation email**.  
🔹 Improve **UI with Bootstrap or Tailwind CSS**.

Would you like any modifications? 😊

Here’s a complete **Event Management System** with the following features:

✅ Admin and User functionalities  
✅ Event details stored in a JSON file  
✅ Beautiful UI with a **translucent background image**  
✅ Responsive layout with **HTML, CSS, and JavaScript**

### 📂 **Project File Structure**

event-management/

│── index.html # Home page (User can view events & book)

│── admin.html # Admin Dashboard (Manage events & bookings)

│── login.html # Login page (for Admin & User)

│── events.json # JSON file for storing event details

│── styles.css # Styling for all pages

│── script.js # JavaScript file for event handling

│── admin.js # JavaScript file for admin functionalities

│── images/ # Background and event images

│── README.md # Project documentation

## 📜 ****1. events.json (Event Data)****

[

{

"id": 1,

"name": "Music Concert",

"date": "2025-03-20",

"venue": "City Auditorium",

"image": "images/concert.jpg"

},

{

"id": 2,

"name": "Tech Conference",

"date": "2025-04-10",

"venue": "Tech Park Hall",

"image": "images/tech.jpg"

},

{

"id": 3,

"name": "Art Exhibition",

"date": "2025-05-05",

"venue": "Art Gallery",

"image": "images/art.jpg"

}

]

## 📜 ****2. index.html (User Page)****

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Event Management</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<header>

<h1>Welcome to Event Management System</h1>

<nav>

<a href="login.html">Login</a>

<a href="admin.html">Admin Dashboard</a>

</nav>

</header>

<section id="events">

<h2>Upcoming Events</h2>

<div id="event-list"></div>

</section>

<script src="script.js"></script>

</body>

</html>

## 📜 ****3. script.js (User Functionality)****

document.addEventListener("DOMContentLoaded", () => {

fetch("events.json")

.then(response => response.json())

.then(events => displayEvents(events))

.catch(error => console.error("Error loading events:", error));

});

function displayEvents(events) {

const eventList = document.getElementById("event-list");

eventList.innerHTML = "";

events.forEach(event => {

const eventCard = document.createElement("div");

eventCard.classList.add("event-card");

eventCard.innerHTML = `

<img src="${event.image}" alt="${event.name}">

<h3>${event.name}</h3>

<p><strong>Date:</strong> ${event.date}</p>

<p><strong>Venue:</strong> ${event.venue}</p>

<button onclick="bookEvent(${event.id})">Book Now</button>

`;

eventList.appendChild(eventCard);

});

}

function bookEvent(eventId) {

alert(`Event ${eventId} booked successfully!`);

}

## 📜 ****4. admin.html (Admin Dashboard)****

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Admin Dashboard</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<header>

<h1>Admin Dashboard</h1>

<nav>

<a href="index.html">Home</a>

<a href="login.html">Logout</a>

</nav>

</header>

<section id="admin-panel">

<h2>Manage Events</h2>

<form id="event-form">

<input type="text" id="event-name" placeholder="Event Name" required>

<input type="date" id="event-date" required>

<input type="text" id="event-venue" placeholder="Venue" required>

<button type="submit">Add Event</button>

</form>

<div id="admin-event-list"></div>

</section>

<script src="admin.js"></script>

</body>

</html>

## 📜 ****5. admin.js (Admin Functionalities)****

document.addEventListener("DOMContentLoaded", () => {

fetch("events.json")

.then(response => response.json())

.then(events => displayAdminEvents(events))

.catch(error => console.error("Error loading events:", error));

document.getElementById("event-form").addEventListener("submit", addEvent);

});

function displayAdminEvents(events) {

const adminEventList = document.getElementById("admin-event-list");

adminEventList.innerHTML = "";

events.forEach(event => {

const eventItem = document.createElement("div");

eventItem.classList.add("event-item");

eventItem.innerHTML = `

<h3>${event.name}</h3>

<p><strong>Date:</strong> ${event.date}</p>

<p><strong>Venue:</strong> ${event.venue}</p>

<button onclick="deleteEvent(${event.id})">Delete</button>

`;

adminEventList.appendChild(eventItem);

});

}

function addEvent(e) {

e.preventDefault();

alert("Event added! (JSON update required manually)");

}

function deleteEvent(eventId) {

alert(`Event ${eventId} deleted! (JSON update required manually)`);

}

## 📜 ****6. styles.css (Styling)****

body {

font-family: Arial, sans-serif;

background: url('images/background.jpg') no-repeat center center/cover;

color: white;

text-align: center;

padding: 20px;

}

header {

background: rgba(0, 0, 0, 0.7);

padding: 15px;

}

nav a {

color: white;

margin: 0 15px;

text-decoration: none;

font-weight: bold;

}

#events, #admin-panel {

background: rgba(0, 0, 0, 0.5);

padding: 20px;

border-radius: 10px;

width: 80%;

margin: auto;

}

.event-card, .event-item {

background: rgba(255, 255, 255, 0.2);

padding: 10px;

border-radius: 8px;

margin: 10px;

display: inline-block;

width: 250px;

}

.event-card img {

width: 100%;

border-radius: 5px;

}

button {

padding: 8px 12px;

background: darkblue;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

}

button:hover {

background: royalblue;

}

### ✅ **Features Implemented**

✔ **User Features:** View events, book events  
✔ **Admin Features:** Add, delete events (updates need manual JSON edits)  
✔ **Beautiful UI:** Translucent background, card-based layout

Would you like **more enhancements** such as **backend integration** or **user authentication**? 🚀