









Hotel Booking App – Project Flow Overview

Total Pages (Frontend)

Page Name	Path	Purpose
 Home Page	/	List all rooms available for booking
 Room Details Page	/rooms/:id	Show full details of selected room
 Booking Page	/book/:id	User fills form to book room
 Admin Add Room Page	/admin/add-room	Admin adds new room
 All Bookings Page (Admin)	/admin/bookings	Admin sees list of all bookings
 Booking Confirmation	/success	After user submits form
 Login Page (Optional Phase 2)	/login	Admin/User login
 User Dashboard (Optional Phase 2)	/dashboard	User sees their bookings

Page-to-Page Flow (User Journey)

Phase 1 — Without Backend (Static Data)

1. User visits /
 - See all rooms from `rooms.js`
 - Click on any room
2. User lands on `/rooms/:id`
 - Room details shown
 - Button: "Book Now"
3. User clicks "Book Now" → `/book/:id`
 - Prefilled form for that room
 - Enters name, email, check-in, check-out

- Click submit

4. User submits form

- (Console logs booking data)
- Redirect to `/success` page

Data Flow (Frontend Only Phase)

```
data/rooms.js
↓
HomePage → map() → <RoomCard />
↓ click
RoomDetails → uses id → find room from rooms.js
↓ click
BookingPage → uses id → fill booking form
↓ submit
Redirect to /success
```

All pages read room data from `rooms.js` .
Bookings aren't stored yet — just logged.

Once Backend Comes (Phase 2+)

- Rooms will come from MongoDB via Express API
- Booking form will send data via POST request to backend
- Admin can log in and manage rooms/bookings
- User dashboard shows past bookings

Summary Visual (Simplified Flow)

```
[Home Page]
↓ click on room
[Room Details]
↓ Book Now
[Booking Page]
↓ Submit
[Success Page]
```

Admin side (Phase 2+):

[Admin Add Room]



Backend: Add room to MongoDB

[Admin Bookings Page]



Backend: Fetch all bookings