



Project Report: The Wild Oasis

Project Title: The Wild Oasis – Hotel Management Dashboard

Client: Sagar Patil, Software Architecture, UK News, London

Developer-Sudarshan Hingalje

Project Duration: June 1 – June 28, 2025

Jira Project:

<https://sudarshanhingalje1.atlassian.net/jira/software/projects/WO/summary>

Live App: <https://the-wild-oasis.netlify.app>

GitHub Repo: <https://github.com/SudarshanHingalje/the-wild-oasis>

Project Overview

The Wild Oasis is a full-featured hotel management dashboard built for internal hotel employees. It helps manage cabins, guests, bookings, and key performance insights efficiently via a clean UI powered by React and Supabase. The project follows Agile development managed via Jira Kanban board.

Objectives

- Build a real-world full-stack application with React + Supabase
- Apply concepts from advanced React course into a practical project
- Learn and integrate third-party libraries for production-ready features
- Practice modular folder structure, reusable components, and state management

Functional & Business Requirements

Authentication

- - Only hotel employees can log in and access the app.
- - New users can only be signed up inside the application.
- - Users can upload avatars, change names, and reset passwords.

Cabins Management

- - View cabins with photo, name, capacity, price, and discount.
- - Create, edit, delete cabins including image upload.

Bookings Management

- Table with booking status, dates, price, guest, and cabin.
- Filter bookings by status: Unconfirmed, Checked In, Checked Out.

Guest Data

- Guest info includes full name, email, ID, nationality, and country flag.

Booking Flow & Validation

- Check-in, check-out, and delete booking flows.
- Confirm payment manually; add breakfast for full stay during check-in.

Dashboard Requirements

- Show guests checking in/out today.
- Statistics on bookings, sales, occupancy, and stay duration.
- Filter data by last 7, 30, or 90 days.

Application Settings

- Adjust breakfast price, min/max nights, and max guests.

UI & UX

- Dark Mode support, responsive layout, reusable UI components.

Tech Stack

Frontend:

- React (Vite)
- Styled Components
- React Router DOM
- React Query (@tanstack/react-query)
- React Hook Form
- React Datepicker
- React Hot Toast
- React Icons

Backend:

- Supabase (PostgreSQL, Auth, API, Storage)
- JSON Server (for prototyping/mock API)

Utilities:

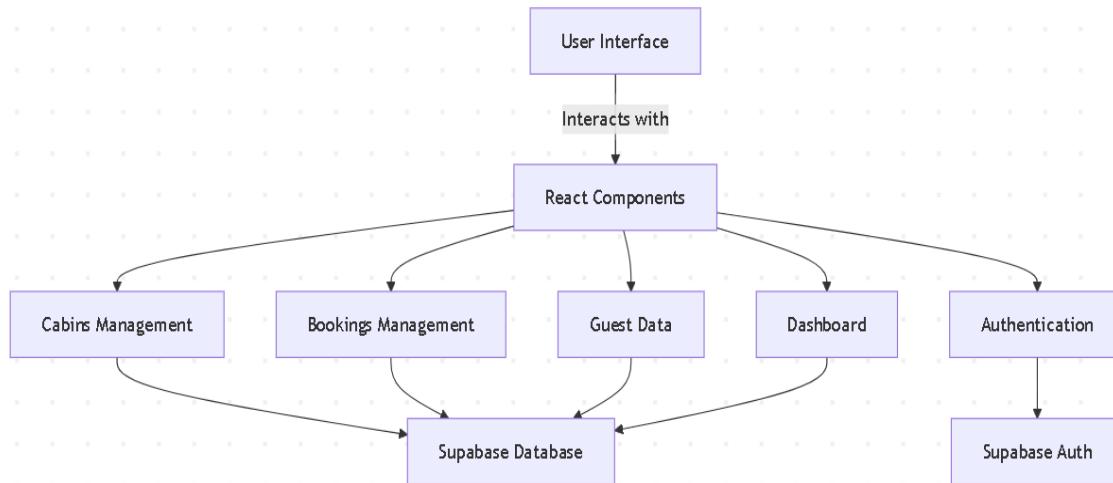
- date-fns
- react-error-boundary
- react-leaflet (mapping)
- Chart.js + react-chartjs-2

📁 Folder Structure Overview

- src/features: Modular domain-based folders (Cabins, Bookings, Guests)
- src/ui: Reusable UI components (Modals, Forms, Buttons)
- src/hooks: Custom React hooks
- src/services: API layer (Supabase service functions)
- src/pages: Route-specific layout
- src/context: Global state context (e.g., dark mode)

📊 System Diagrams

UML diagrams,

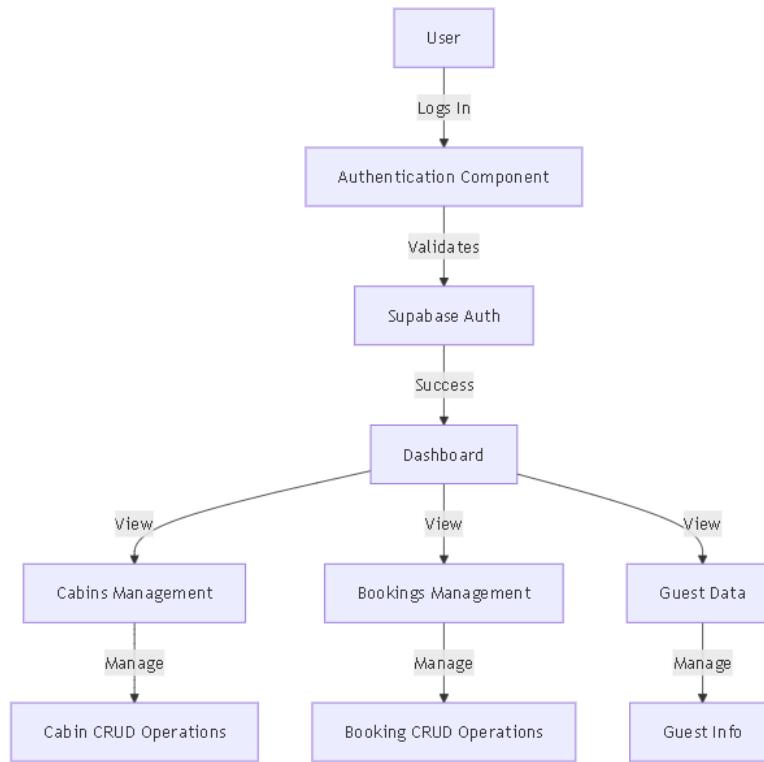


- **UI Overview:** The User Interface (UI) interacts with various React components for functionality.
- **Component Connections:** Several management components (Cabins & Bookings) and data components (Guest Data & Dashboard) are integrated.
- **Database Interaction:** The UI communicates with the Supabase database to fetch and manage information.
- **Authentication Flow:** Authentication is also a key component, linking to Supabase for user verification.

•

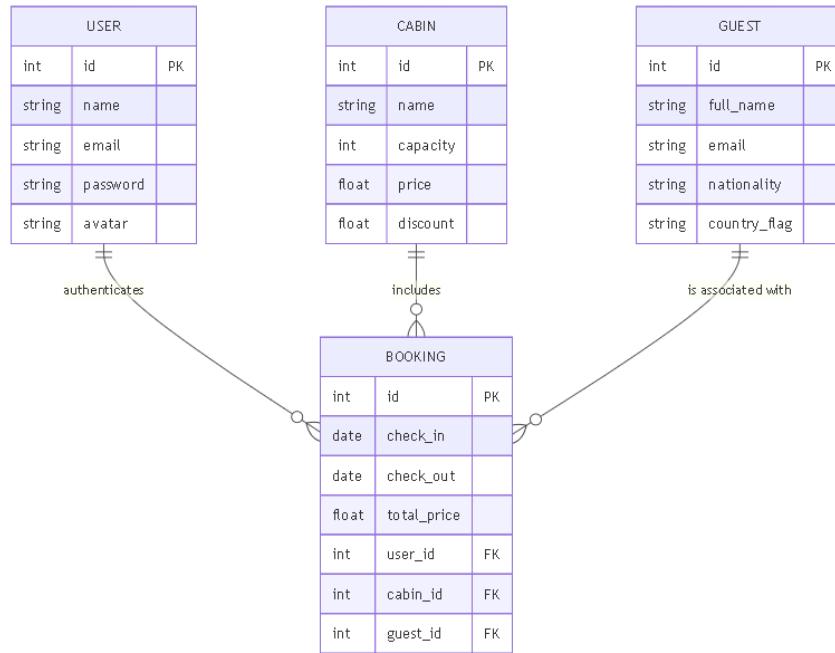
•

component flows



- **User Interaction:** The user initiates the process by logging in through an authentication component.
- **Validation:** The authentication component validates credentials via an supabase authentication.
- **Dashboard Access:** Upon successful validation, the user gains access to the dashboard.
- **Management Options:** The dashboard offers views for managing cabins, bookings, and guest data.

ER diagrams



Entities: Three main entities are defined: User, Cabin, and Guest, each with specific attributes.

Booking Relations: The Booking table serves as a junction to connect User, Cabin, and Guest, indicating relationships.

Data Types: Different data types (e.g., int, string, float) are specified for each attribute across the tables.

Foreign Key Links: Relationships are established via foreign keys, ensuring data integrity across linked tables.

Screenshots

Dashboard, Cabins, Add Cabin, Bookings, Create Booking, Guests, Guest Detail View, Statistics Panel, Settings Panel.

Screenshots available in the GitHub repository.

Third-Party Libraries Used

Date	02 / 30 / 2025
Page	1
 3rd-Party Libraries & NPM Plugins Used	
1. react-router-dom@6 - Routing purposes	
2. json-server - JSON data fetch(virtual backend)	
3. react-leaflet - Big open-source library for maps	
4. react-datepicker - Calendar/date picker component	
5. @reduxjs/toolkit - State management tool for large applications	
6. redux-thunk - Middleware for Redux	
7. react-redux - Redux bindings for React	
8. redux-devtools-extension - Debuggingextension for Redux	
9. reduxjs/toolkit - (Duplicate entry already listed at #5)	
10. react-icons - Icon library for React	
11. @tanstack/react-query@4 - Data fetching and caching	
12. date-fns - Utility functions for manipulating dates	
13. react-hot-toast - Toast notifications for React	
14. react-hook-form@7 - Lightweightform management	
15. flagcdnnet/download/api - Country flag API	
16. react-chartjs-2 - React wrapperfor Chartjs (for charts)	
17. react-error-boundary - Error handling boundaries for React	

See handwritten list in repo: <https://github.com/user-attachments/assets/521a1d5d-8230-4e90-8c36-e14382be4bd4>

Authentication & Authorization

Handled via Supabase Auth with protected routes using React Router. Admin-only dashboard access.

Stats, Charts & Reports

Rendered with Chart.js. Real-time data stats: total bookings, revenue, occupancy. Date filtering supported.

Deployment

Frontend hosted on Netlify.
Backend via Supabase.
Environment variables handled in `.env`.

Future Improvements

- Role-based access
- Mobile app version (React Native/PWA)
- Payment gateway integration
- Report export (PDF/Excel)

Jira Kanban Summary (Epics & Tasks)

EPIC: User Authentication

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---
Epic User Authentication Auth with Supabase 1 Jun 10 Jun — Done 
Task Create Signup Page Signup using Supabase 1 Jun 2 Jun 1 day Done 
"Signup looks smooth."
Subtask Form validation Required field check 1 Jun 2 Jun 0.5 day Done 
Subtask Error handling Show server error 1 Jun 2 Jun 0.5 day Done 
Task Create Login Page With validations and redirects 2 Jun 3 Jun 0.5 day Done 
"Add forgot password?"
Bug Auth fails on 500 error Server crashes without catch 3 Jun 4 Jun — Fixed 
"App crashed on retry."
Task Add Protected Routes Use React Router 3 Jun 4 Jun 0.5 day Done 
"Admin locked properly."
Task Handle Auth Token Store token in localStorage 4 Jun 5 Jun 1 day Done 
"Log out after inactivity?"
Spike Session persistence Investigate refresh behavior 5 Jun 5 Jun 0.5 day Resolved 

EPIC: Cabin Management

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---
Epic Cabin Management CRUD for cabins 11 Jun 18 Jun — Done <input checked="" type="checkbox"/>
Task Create Cabin Table Supabase schema 11 Jun 12 Jun 1 day Done <input checked="" type="checkbox"/> "Schema looks clean."
Subtask Add constraints Required fields + types 11 Jun 12 Jun 0.5 day Done <input checked="" type="checkbox"/>
Subtask Add storage bucket For cabin images 11 Jun 12 Jun 0.5 day Done <input checked="" type="checkbox"/>
Task Build Add/Edit Form React Hook Form with upload 12 Jun 13 Jun 1.5 days Done <input checked="" type="checkbox"/> "Love the layout."
Subtask Add validation Price and capacity fields 12 Jun 13 Jun 0.5 day Done <input checked="" type="checkbox"/>
Task Display Cabins List With edit/delete actions 14 Jun 14 Jun 0.5 day Done <input checked="" type="checkbox"/>
Task Image Upload Supabase storage integration 15 Jun 16 Jun 1 day Done <input checked="" type="checkbox"/> "Please compress large files."
Bug Image upload failed Bucket permission misconfig 15 Jun 16 Jun — Fixed <input checked="" type="checkbox"/> "Upload fails on mobile."

EPIC: Booking System

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---

| Epic | Booking System | Booking + guest forms + toggle paid | 19 Jun | 24 Jun | — | Done  | |

| Task | Design Booking Schema | check-in/out, guestId, cabinId | 19 Jun | 19 Jun | 1 day | Done  | |

| Task | Create Booking Form | Prefill based on cabin | 20 Jun | 21 Jun | 1.5 days | Done  | "Nice autofill." |

| Task | Show Guest List | Table with filter/search | 22 Jun | 22 Jun | 0.5 day | Done  | "Add phone no?" |

| Task | Payment Toggle | Paid/unpaid status | 23 Jun | 23 Jun | 0.5 day | Done  | "Can we tag overdue?" |

| Bug | Guest info not updating | Guest state not synced | 21 Jun | 21 Jun | — | Fixed  | |

| Subtask | Create test bookings | 5 test entries | 20 Jun | 20 Jun | 0.5 day | Done  | |

EPIC: Admin Dashboard

Type	Summary	Description	Start	Due	Est. Time	Status	Client Comment

| Epic | Admin Dashboard | Revenue stats, filters | 25 Jun | 28 Jun | — | In Progress  | |

| Task | Add Chart for Revenue | Area chart | 25 Jun | 25 Jun | 1 day | Done  | "Nice chart visuals." |

| Task | Add Stats Cards | This month/week stats | 26 Jun | 26 Jun | 0.5 day | Done  | "Add bookings count?" |

| Task | Add Date Filter | Custom range calendar | 27 Jun | 28 Jun | 1 day | In Progress  | |

| Spike | Evaluate chart libraries | Recharts vs Chart.js | 25 Jun | 25 Jun | 0.5 day | Resolved  | |

EPIC: UI/UX Design

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---
Epic UI/UX Design Layouts, responsiveness 6 Jun 28 Jun — In Progress 
Task Sidebar & Navigation App shell with routing 6 Jun 7 Jun 1 day Done 
“Feels like a real app.”
Task Mobile Responsiveness Media queries for small screens 8 Jun 12 Jun 2 days In Progress 
Task Toasts + Spinners Loading states + feedback 13 Jun 14 Jun 0.5 day To Do 

EPIC: Database Setup

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---
Epic Database Setup Schema, API, RLS policies 1 Jun 5 Jun — Done 
Task Create Supabase Project DB + keys 1 Jun 1 Jun 0.5 day Done 
Task Add RLS Policies Restrict access to guests only 2 Jun 2 Jun 0.5 day Done 
Task Seed Test Data Add 10 bookings/guests 3 Jun 4 Jun 1 day Done  “Helps with testing!”
Bug Guest table constraint Duplicate emails allowed 3 Jun 3 Jun — Fixed 

EPIC: Deployment & Testing

Type Summary Description Start Due Est. Time Status Client Comment
--- --- --- --- --- --- --- ---
Epic Deployment & Testing Vercel deploy + QA 25 Jun 28 Jun — In Progress 
Task Vercel Integration Auto deploy via GitHub 25 Jun 25 Jun 0.5 day Done 
Task Add Supabase Env Vars Public key & URL 26 Jun 26 Jun 0.5 day Done 
Task Test Booking Flows Full booking process run 27 Jun 28 Jun 1 day In Progress  "Add success message."

Conclusion

The Wild Oasis is a production-level dashboard demonstrating advanced proficiency in React, state management, authentication, and modern design patterns using styled-components.