Phase 1: Problem Understanding & Scope

Introduction

Accommodation management is a common challenge for multiple groups:

- **Tourists & Travellers** – need short-term stays for leisure, business trips, or events.
- **Students & Working Professionals** – require long-term affordable rentals such as PGs, hostels, or shared flats.

The **existing market** has several limitations:

- **Unverified Listings** – Many platforms allow fake or outdated property details.
- **Broker Dependence** – Manual, broker-driven processes inflate costs.
- **Communication Gaps** – Hosts and guests lack structured communication channels.
- **Low Transparency** – Guests face uncertainty regarding booking status, payments, and reviews.

Proposed Solution

StayEase CRM is a Salesforce-based CRM project designed to create a structured and trustworthy **accommodation and rental platform**. It takes inspiration from Airbnb's short-term rental model while extending to long-term housing for students and professionals.

Key Features of the Solution:

- Centralized database of verified properties.
- Seamless booking flow for **short-term** and **long-term** stays.
- Role-based data access for **Admin, Hosts, Guests, and Managers**.
- Automation for booking status, payments, and complaint resolution.
- Reports and dashboards to monitor occupancy, revenue, and service quality.

Objectives

1. Create a **unified accommodation platform** with transparent workflows.
2. Support **short-term bookings** for tourists and **long-term rentals** for students/professionals.
3. Provide **role-based security** for data integrity.
4. Use Salesforce automation (Flows, Validation Rules, Approvals) to improve efficiency.
5. Deliver **business insights** through dashboards and reports.

🔹 Phase 2: Org Setup & Security

To implement the foundation of StayEase CRM, **roles, profiles, org-wide defaults, and permissions** are configured.

2.1 Roles (Role Hierarchy)

- **CEO** → Top-level access; visibility into all records, reports, and dashboards.
- **Operations Manager** → Oversees all bookings, complaints, and escalations across the org.
- **Host** → Can only manage their own properties, bookings, and related complaints.
- **Guest (Traveller/Student/Professional)** → Can book properties and raise complaints related to their own bookings.
- **System Administrator** → Full access, controls setup and configurations (you).

2.2 Profiles

Profiles define **what a user can do** inside Salesforce.

- **Host Profile**
  - Can create, view, and edit only their properties.
  - Can view bookings related to their properties.
  - Limited access to complaints (only related ones).
- **Guest Profile**
  - Can create and manage only their own bookings.
  - Can raise complaints for their own bookings/properties.
  - Read-only access to property listings.
- **Operations Manager Profile**
  - Full access to all bookings and complaints.
  - Can intervene/escalate complaints.
  - Cannot perform system-level configurations.
- **CEO Profile**
  - Read-only access to all data.
  - Access to dashboards and reports for strategic decision-making.
- **System Admin Profile**
  - Full CRUD (Create, Read, Update, Delete) permissions on all objects.
  - Handles automation, security, and backend setup.

2.3 Org-Wide Defaults (OWD)

Default record-level security settings:

- **Property__c** → Private (Hosts can only see their properties).
- **Booking__c** → Private (Guests can see their own bookings, Hosts can see bookings related to their properties).
- **Complaint__c** → Private (Guests see their complaints; Ops Manager sees all).
- **Payment__c** → Private (Linked to bookings; visible only to Guest, Host, and Ops Manager).

2.4 Sharing Rules

- **Hosts** get automatic access to bookings made on their properties.
- **Ops Managers** get access to all complaints and escalated bookings.
- **CEO** gets read-only access to all objects for reporting.

2.5 Permission Sets

Additional flexibility beyond profiles:

- **Host_Extra_Access** → If needed, allow hosts to see limited guest details (like name & contact for bookings).
- **Ops_Manager_Analytics** → Access to advanced dashboards and reports.

2.6 Security Best Practices

- **Field-Level Security** → Hide sensitive fields (like Payment Transaction IDs) from Hosts/Guests.
- **Read-Only Fields** → Booking ID, Payment ID, Total Rent should be auto-generated and locked.
- **Validation Rules** → Prevent invalid data entry (e.g., check-in date after check-out date).

☑ **Outcome of Phase 2:**
At this stage, the org is structured with a secure role hierarchy, profiles, permissions, and OWD rules. Each actor (Admin, CEO, Operations Manager, Host, Guest) has **only the data visibility and access they need**, ensuring data privacy, integrity, and smooth system operation.