

Problem Statement

Managing accommodations for diverse groups such as tourists, students, and working professionals is often challenging due to fragmented platforms and lack of personalization.

- **Tourists & Travelers** need short-term stays with flexibility.
- **Students & Professionals** require long-term, affordable, and reliable housing like PGs, hostels, or rentals.
Traditional property management systems lack centralized communication, efficient booking handling, transparent ratings/reviews, and host–guest coordination.

This results in:

- Inefficient property discovery
- Lack of trust between host and guest
- Manual management of payments & complaints
- Limited visibility of guest/host performance



StayEase CRM solves these issues by creating a **Salesforce-powered centralized platform** for property management, booking, guest handling, and payments.

Industry Type

- **B2C (Business-to-Consumer):**

StayEase directly connects property hosts with end-users (tourists, students, working professionals).

Target Users

1. **Hosts** – Individuals or businesses listing their properties.
2. **Guests** – Tourists, students, or professionals booking stays.
3. **Admins** – Oversee operations, resolve disputes, and manage CRM functionalities.

Key Objectives

- Centralize property listings, bookings, payments, and complaints.
- Provide a **trust-building mechanism** via guest ratings and host performance.
- Offer different stay types: **short-term (Airbnb-like)** and **long-term (student/professional rentals)**.
- Enhance **customer satisfaction** with transparent processes and automation

Value Proposition

- For **Guests** → Easy property discovery, secure payments, and transparent ratings.
- For **Hosts** → Better visibility, guest management, and simplified bookings.
- For **Admin/Organization** → A scalable CRM system on Salesforce that tracks all activities in one place.

◆ 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- **Unverified Listings** – Many platforms allow fake or outdated property details.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

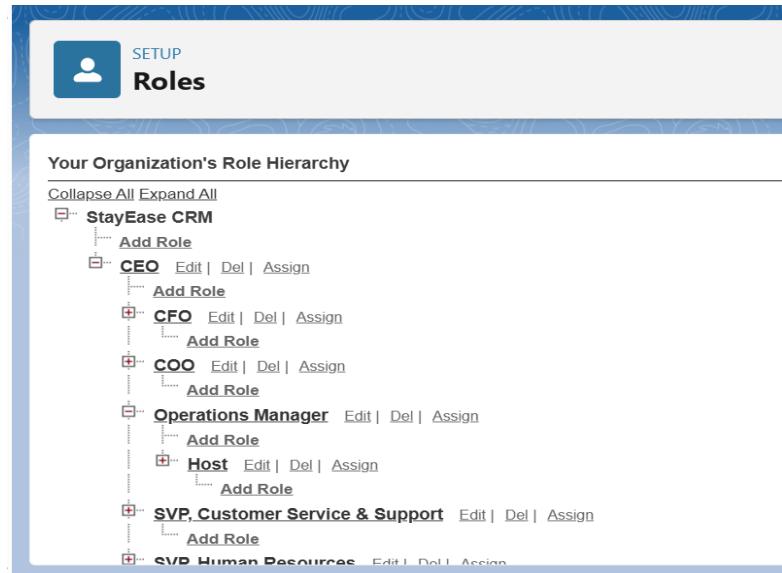
1. Create a **unified accommodation platform** with transparent workflows.
2. Support **short-term bookings** for tourists and **long-term rentals** for students/professionals.
3. Provide **role-based security** for data integrity.
4. Use Salesforce automation (Flows, Validation Rules, Approvals) to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



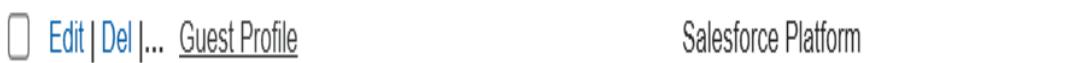
2.2 Profiles

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.

❖ 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.

❖ Phase 3 Documentation – StayEase CRM

The screenshot shows the StayEase CRM interface. At the top, there's a search bar with placeholder text "Search..." and a navigation bar with links for Guests, Hosts, Bookings, Complaints, Payments, and Properties. Below the navigation is a "Recently Viewed" section for Guests, which lists one item: "1 item • Updated a minute ago". The list table has columns for "Guest ID" and "Name". The first row shows "1" and "G-0001". On the right side of the list, there are several action buttons: New, Import, Change Owner, Assign Label, and others for filtering and sorting.

1. Host Object

Purpose: Represents the property owner.

Fields:

1. Total Properties

- Data Type: Roll-Up Summary
- Relationship: Master–Detail with Property
- Summary Type: COUNT
- Description: Counts how many properties the host has listed.

2. Overall Host Rating

- Data Type: Roll-Up Summary
- Relationship: Master–Detail with Property
- Summary Type: AVERAGE
- Field to Aggregate: Property Rating (on Property object)
- Description: Shows the average rating across all properties owned by the host.

The screenshot shows the StayEase CRM Host object details page for "H-0001". The page has tabs for "Related" and "Details". The "Details" tab is active and displays the following fields:

- Name: Sample host1
- Owner: Shruti Deshpande
- Host ID: H-0001
- Contact Email: hostemail@gmail.com
- Contact Phone: 65656565656
- Total Properties: 1
- Created By: Shruti Deshpande, 9/18/2025, 8:30 AM
- Last Modified By: Shruti Deshpande, 9/18/2025, 8:32 AM

2. Property Object

Purpose: Represents an accommodation unit listed on the platform.

Relationships:

- Master-Detail: Property → Host
- Master-Detail: Booking → Property

Fields:

1. Property Rating (Average)

- Data Type: Roll-Up Summary
- Summarized Object: Booking
- Summary Type: AVERAGE
- Field to Aggregate: Guest Rating
- Description: Average rating based on all guest reviews for this property.

2. Total Bookings

- Data Type: Roll-Up Summary
- Summarized Object: Booking
- Summary Type: COUNT
- Description: Number of times this property has been booked.

The screenshot shows a form for a property named 'Anant Villa'. The fields include:

- Property Name: Anant Villa
- Host: H-0001
- Property ID: P-0001
- Type: Villa
- Rent: ₹1,000.00
- Rent Type: Per Night
- Location: Ravi nagar , Nagpur
- Amenities: WiFi;AC;Kitchen;Cook;Parking;Well furnished
- Availability: (checkbox checked)
- Max Occupancy: 5
- Booking Ratings Sum: 0.0

3. Booking Object

Purpose: Represents the reservation details when a Guest books a Property.

Relationship:

- Master-Detail: Booking → Property

Fields:

1. Check-In Date

- Data Type: Date
- Description: Start date of booking.

2. Check-Out Date

- Data Type: Date
- Description: End date of booking.

3. Number of Guests

- Data Type: Number (2,0)
- Description: Total number of people staying in this booking.

4. Booking Status

- Data Type: Picklist
- Values: Requested, Confirmed, Cancelled, Completed
- Description: Tracks the current status of the booking.

The screenshot shows a form for a booking with ID 'B-0001'. The fields include:

- Booking ID: B-0001
- Property: Anant Villa
- Guest: G-0001
- Status: Requested
- Check In Date: 9/22/2025
- Check Out Date: 9/27/2025
- Number of Guests: 5
- Guest Rating: (checkbox)
- Total Rent: ₹5,000.00
- Created By: Shruti Deshpande, 9/18/2025, 8:34 AM
- Last Modified By: Shruti Deshpande, 9/18/2025, 8:34 AM

5. Guest Rating

- Data Type: Number (2,1), Range: 1–5
- Description: Rating provided by the Guest for this booking/property.

4. Guest Object

Fields:

- **Name** (Text)
- **Email** (Email)
- **Phone** (Phone)
- **Guest Type** (Picklist → Tourist, Student, Working Professional)
- **Total Bookings** (Roll-Up Summary → COUNT of related Bookings)
- **Average Rating Given** (Roll-Up Summary → AVERAGE of Guest Rating from Bookings)
- **Loyalty Points** (Number → optional/future use)

The screenshot shows the 'Details' tab of a Guest object record. The record ID is G-0001. The客 (Owner) is Shruti Deshpande. The客 (Guest ID) is G-0001. The客 (Full Name) is Sample guest 1. The客 (Email) is samplemail@gmail.com. The客 (Phone) is (657) 835-4834. The客 (Guest Type) is Tourists. The客 (Created By) is Shruti Deshpande, created on 9/17/2025, 11:44 PM. The客 (Last Modified By) is Shruti Deshpande, modified on 9/18/2025, 8:35 AM.

Relationships:

- **Master-Detail:** Booking → Guest (One Guest can have many Bookings)

5. Complaint Object

- **Complaint ID (Auto Number)** → e.g. C-0001
- **Owner (Standard field)** → User responsible
- **Guest (Lookup to Guest)** → Who raised the complaint
- **Property (Lookup to Property)** → Complaint related property
- **Complaint Type (Picklist)** → Options like *Cleanliness, Payment, Other*
- **Status (Picklist)** → *Open, In Progress, Resolved, Closed*
- **Description (Long Text Area)** → Guest explains the problem

The screenshot shows the 'Details' tab of a Complaint object record. The记录 ID is C-0001. The客 (Owner) is Shruti Deshpande. The客 (Complaint ID) is C-0001. The客 (Guest) is G-0001. The客 (Property) is Anant Villa. The客 (Complaint_Type) is Other. The客 (Status) is In Progress. The客 (Description) is (empty). The客 (Created By) is Shruti Deshpande, created on 9/19/2025, 1:05 AM. The客 (Last Modified By) is Shruti Deshpande, modified on 9/19/2025, 1:05 AM.

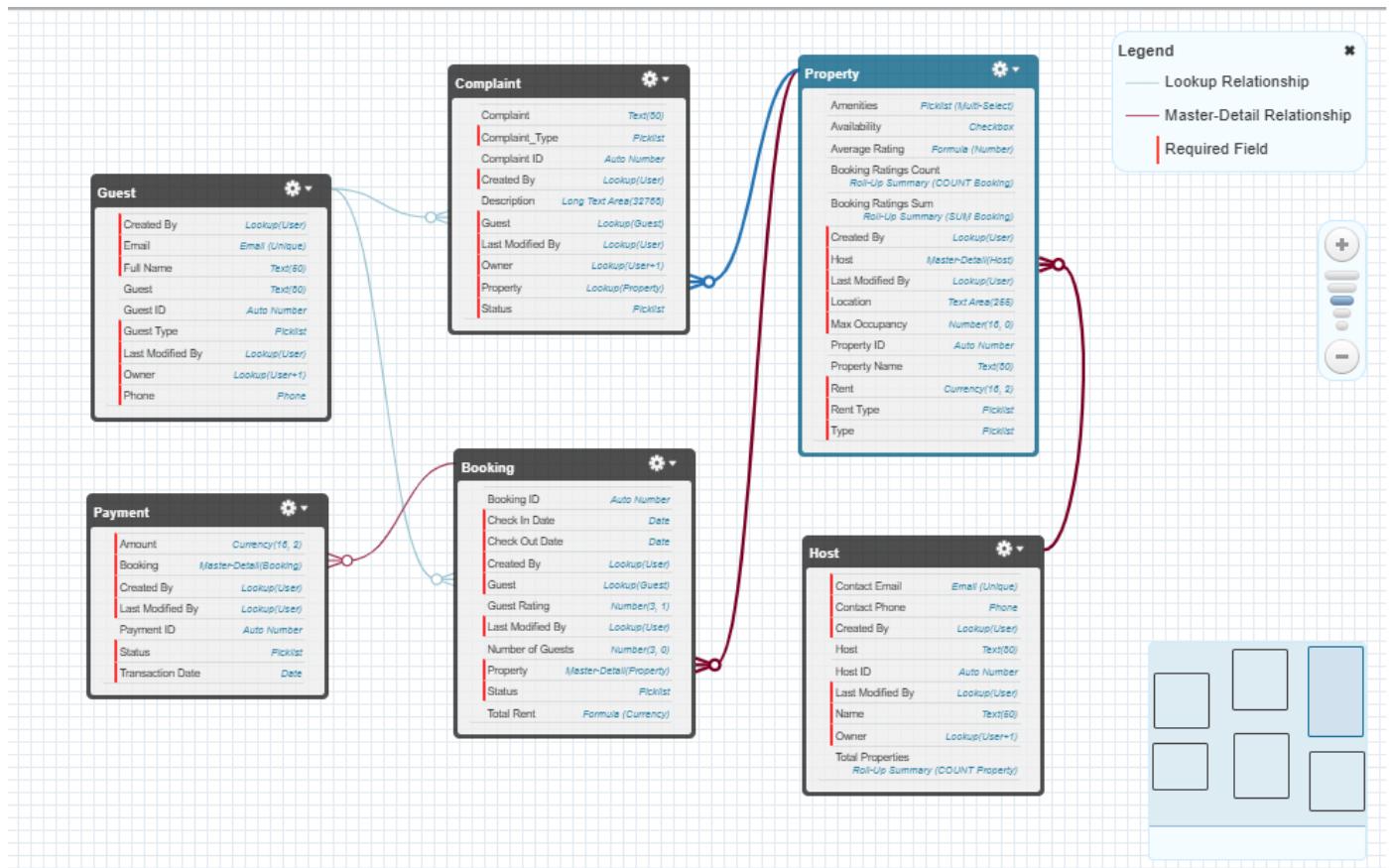


fig : Schema Diagram (Form Schema Builder)