

Tours and Travels CRM

Salesforce Documentation

Project Overview

Tours and Travels CRM is a system that enables travel agencies to manage their entire business process in one platform. It includes customer management, bookings, payments, travel packages, employee coordination, and customer feedback. The CRM reduces reliance on manual work and provides automation for activities such as booking confirmations, payment reminders, and feedback collection. It also offers dashboards and reports to monitor revenue, customer satisfaction, and overall business performance. By integrating these features, the system ensures smoother operations, improved communication, and enhanced customer service.

Objectives

The overall objective of the CRM is to automate travel operations and improve efficiency for agencies. It seeks to:

- Provide better customer management by centralizing all customer information and bookings.
- Streamline and automate the booking and payment process to save time and minimize errors.
- Enhance customer satisfaction with instant confirmations, timely reminders, and feedback tracking.

- Enable employees (agents, guides, and finance teams) to collaborate more effectively with role-based access.
- Deliver actionable business insights with reports and dashboards, allowing managers to track revenue, performance, and customer retention.

Overall, the vision is to develop a scalable, User-friendly, and secure CRM to support both daily operations and long-term growth for travel agencies.

Phase 1: Requirement Analysis & Planning

Understanding Business Requirements

The initial task was to comprehend how travel agencies operate internationally. This involved learning about their day-to-day problems like processing bookings for individuals and groups, managing customer data, monitoring payments, and maintaining effective communication among agents, guides, and customers. I also examined existing manual system issues—such as late updates, lost feedback, and no proper tracking—which underscored the need for an automated CRM solution.

Defining Project Scope & Objectives

The project was to develop a scalable CRM that could be utilized by travel agencies all over the world. Its scope spanned the complete booking process—from customer inquiry to payments and feedback collection. The objectives were well-defined: minimize manual effort, enhance customer satisfaction, and provide managers with insights into revenue and performance through reports and dashboards. One of the essential aspects of the scope was also to ensure role-based access, so various teams (agents, guides, finance team, and admins) only view and deal with what is appropriate to them.

Gathering & Analyzing User Needs

In order to make the system useful, I researched the needs of various Users: customers, travel agents, guides, finance teams, and admins. Customers wished for easy onboarding, package selection, and immediate booking confirmation. Agents and guides required tools to handle bookings, tasks, and communication. Finance staff needed to see payments and reminders, whereas admins wanted reports for measuring performance. I collected these needs through Google Forms, internet research, Salesforce guides, and Trailhead modules, and the design was thus grounded in real-life scenarios.

Identifying Key Salesforce Features & Tools Required

Based on analysis, I enumerated the Salesforce tools that could effectively address these needs. Custom objects such as Booking, Travel Package, Customer, Employee, and Payment would store all key information. Automation features such as Flows, Workflow Rules, and Approval Processes would do repetitive tasks like reminders and confirmations. Apex code (Triggers, Batch Apex, Queueable Apex) would execute more complicated tasks. For the Users interface, Lightning Apps, Dynamic Forms, and LWC components would make it easy to use. On the security front, Profiles, Roles, Permission Sets, and Sharing Rules would provide adequate access.

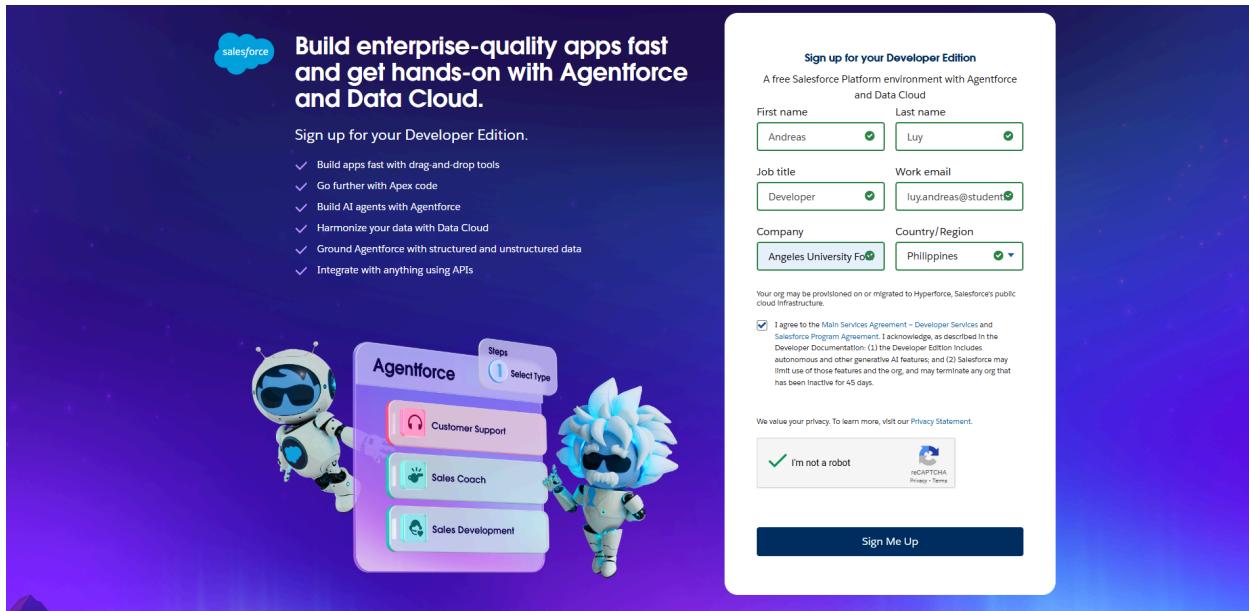
Designing Data Model and Security Model

The data model was created in order to connect important entities to one another: Bookings with Customers, Travel Packages, and Employees, and to monitor payments, feedback, and guest information. This provided a structure where all information related to bookings was logically connected. The security design was based on a straightforward hierarchy—Travel Managers in charge of Agents and Guides. Profiles managed object-level access, and permission sets gave flexibility to grant further access as required. Sensitive information, such as payment details, was limited to finance Users, and sharing rules permitted guides to view only records in relation to their work.

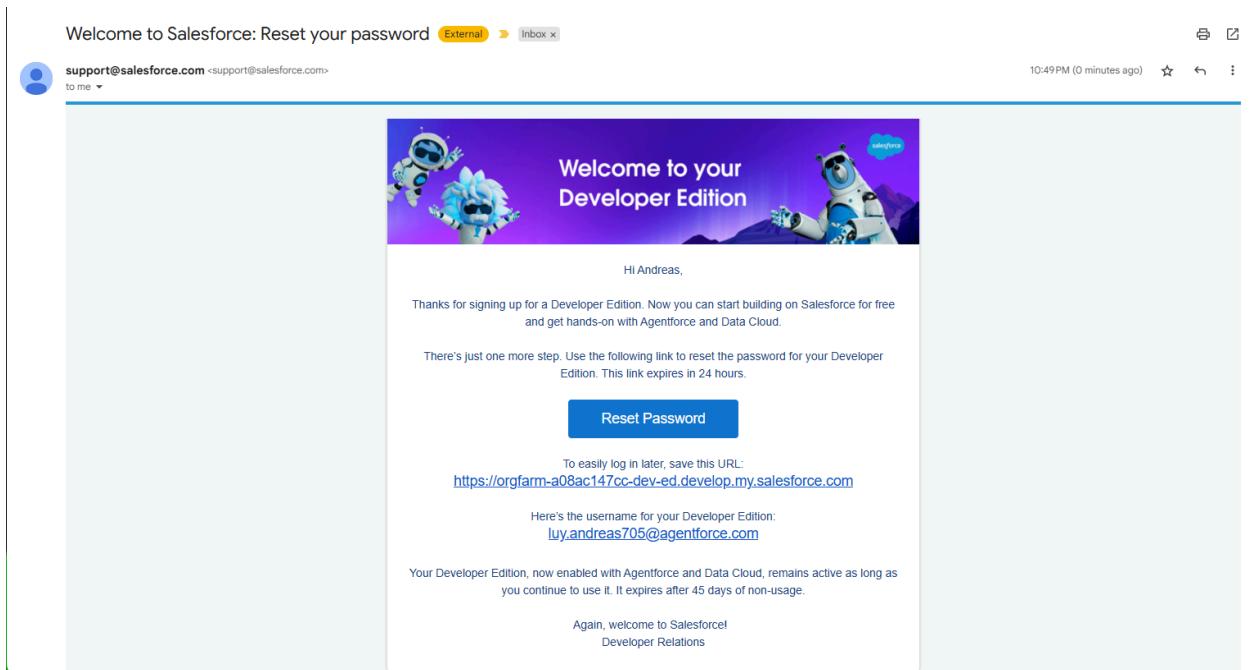
Phase 2: Salesforce Development – Backend & Configurations

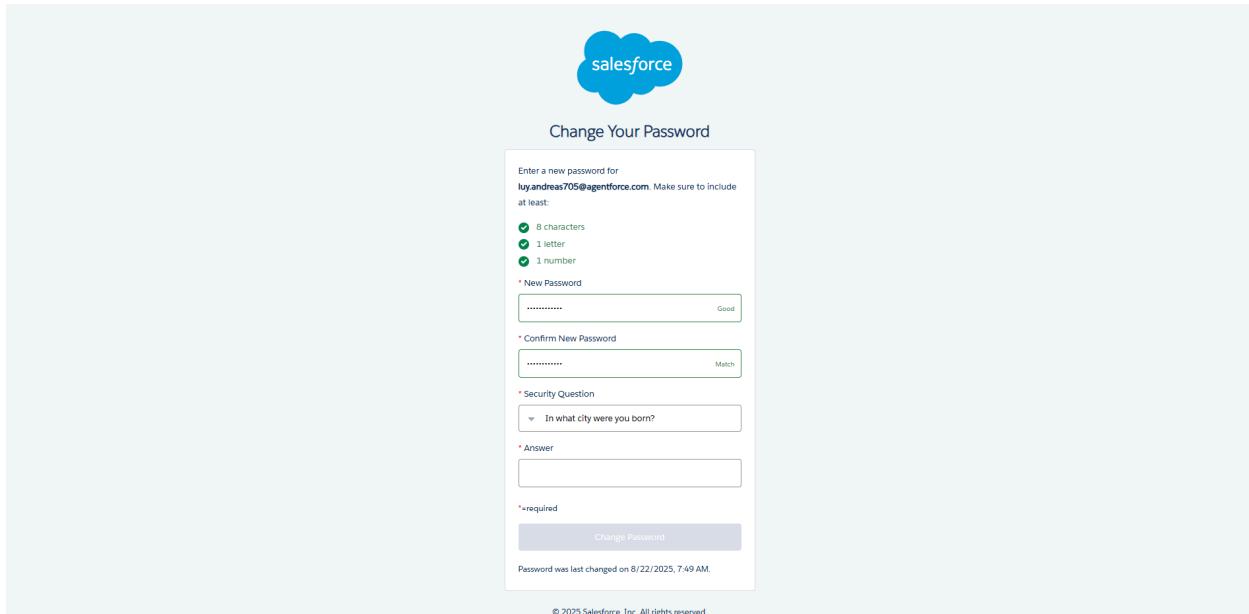
Milestone 1: Salesforce Account

Activity 1: Creating Developer Account

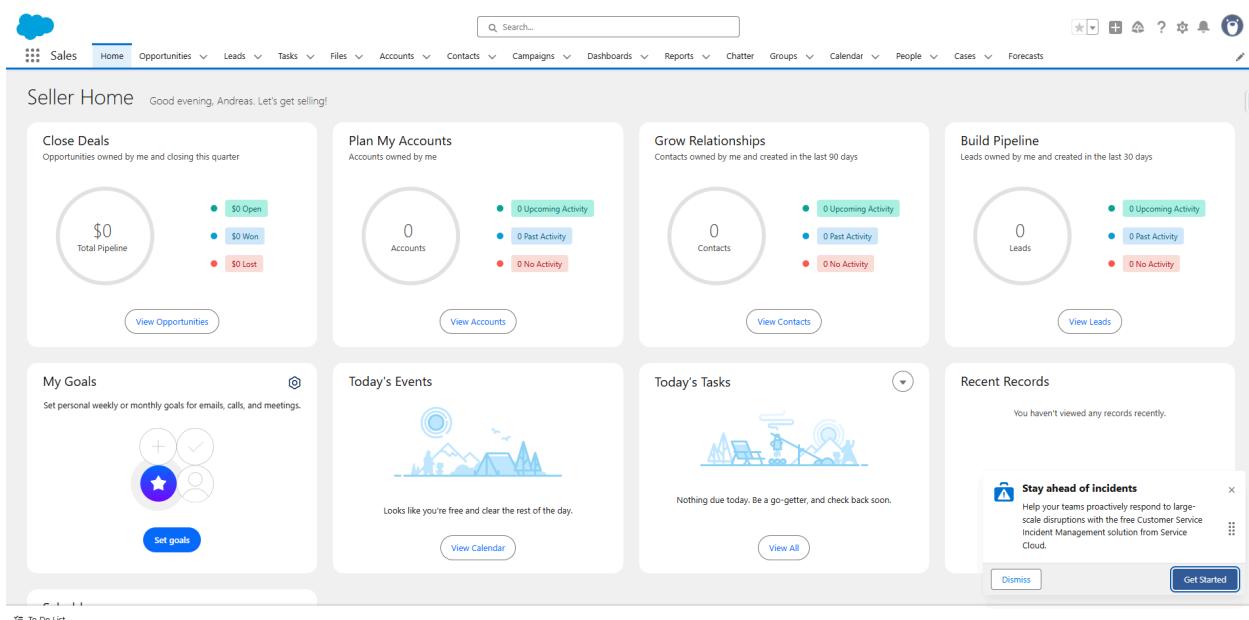


Activity 2: Account Activation





The screenshot shows the Salesforce 'Change Your Password' page. At the top is the blue cloud logo followed by the word 'salesforce'. Below it is the heading 'Change Your Password'. A form is displayed for entering a new password, including fields for 'New Password' and 'Confirm New Password', both of which are marked as 'Good' and 'Match' respectively. There are also fields for a 'Security Question' ('In what city were you born?') and an 'Answer'. A note at the bottom states 'Password was last changed on 8/22/2025, 7:49 AM.' At the very bottom of the page is a copyright notice: '© 2025 Salesforce, Inc. All rights reserved.'



The screenshot shows the Salesforce Seller Home dashboard. At the top is a navigation bar with icons for Sales, Home, Opportunities, Leads, Tasks, Files, Accounts, Contacts, Campaigns, Dashboards, Reports, Chatter, Groups, Calendar, People, Cases, Forecasts, and a search bar. Below the navigation is a greeting 'Seller Home Good evening, Andreas. Let's get selling!'. The dashboard features several cards: 'Close Deals' (Total Pipeline: \$0, Open: 0, Won: 0, Lost: 0), 'Plan My Accounts' (Accounts owned by me: 0, Upcoming Activity: 0, Past Activity: 0, No Activity: 0), 'Grow Relationships' (Contacts owned by me and created in the last 90 days: 0, Upcoming Activity: 0, Past Activity: 0, No Activity: 0), 'Build Pipeline' (Leads owned by me and created in the last 30 days: 0, Upcoming Activity: 0, Past Activity: 0, No Activity: 0), 'My Goals' (Set personal weekly or monthly goals for emails, calls, and meetings, with a 'Set goals' button), 'Today's Events' (Shows a calendar icon and the message 'Looks like you're free and clear the rest of the day.' with a 'View Calendar' button), 'Today's Tasks' (Shows a tasks icon and the message 'Nothing due today. Be a go-getter, and check back soon.' with a 'View All' button), and 'Recent Records' (Shows a message 'You haven't viewed any records recently.'). A 'Stay ahead of incidents' notification is present in the bottom right corner.

For this milestone, I began by setting up my own account on Salesforce Developer Org. To create my own account, I began by following the steps on the activity. Setting up the developer account was very straightforward and marked my first step to the Salesforce ecosystem. It gave me access to the system where I could start experimenting and exploring different features and functionalities. This is the first step of my learning journey.

Milestone 2: Objects Creation

Activity 1: Creating a Customer Info Object

Customer Info Object

The screenshot shows the 'New Custom Object' page in the Salesforce Setup interface. The 'Custom Object Information' section includes:

- Label: Customer Info (Example: Account)
- Plural Label: Customers Info (Example: Accounts)
- Starts with vowel sound:
- The Object Name is used when referencing the object via the API: Object Name: Customer_Info (Example: Account)
- Description: (Empty text area)
- Context-Sensitive Help Setting: Open the standard Salesforce.com Help & Training window (radio button selected)
- Content Name: None

The 'Enter Record Name Label and Format' section includes:

- The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.
- Record Name: Customer Name (Example: Account Name)
- Data Type: Text (Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.)

The 'Optional Features' section includes:

- Allow Reports:
- Allow Activities:
- Track Field History:
- Allow in Chatter Groups:

The screenshot shows the 'Customer Info' object details page in the Salesforce Object Manager. The 'Details' tab is selected, showing:

- Fields & Relationships
- Page Layouts
- Lightning Record Pages
- Buttons, Links, and Actions
- Compact Layouts
- Field Sets
- Object Limits
- Record Types
- Related Lookup Filters
- Search Layouts
- List View Button Layout
- Restriction Rules
- Scoping Rules
- Object Access
- Triggers
- Flow Triggers
- Validation Rules
- Conditional Field Formatting

The main panel displays the object's details:

Details	Value
Description	(Empty text area)
API Name	Customer_Info_c
Custom	<input checked="" type="checkbox"/>
Singular Label	Customer Info
Plural Label	Customers Info
Enable Reports	<input checked="" type="checkbox"/>
Track Activities	<input checked="" type="checkbox"/>
Track Field History	<input checked="" type="checkbox"/>
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

Buttons at the top right: Edit, Delete.

Booking Object

The screenshot shows the 'Object Manager' section of the Salesforce Setup. A message at the top indicates that permissions are disabled for all profiles by default. The 'Custom Object Definition Edit' page is displayed, with the object name 'Booking' entered in the 'Label' field. Other fields like 'Plural Label' (Bookings), 'Object Name' (Booking), and 'Description' are also filled out. The 'Record Name' field is set to 'Booking Name'. Under 'Optional Features', 'Allow Reports' and 'Allow Activities' are checked. The 'Data Type' is set to 'Text'. The right side of the screen shows a vertical scroll bar.

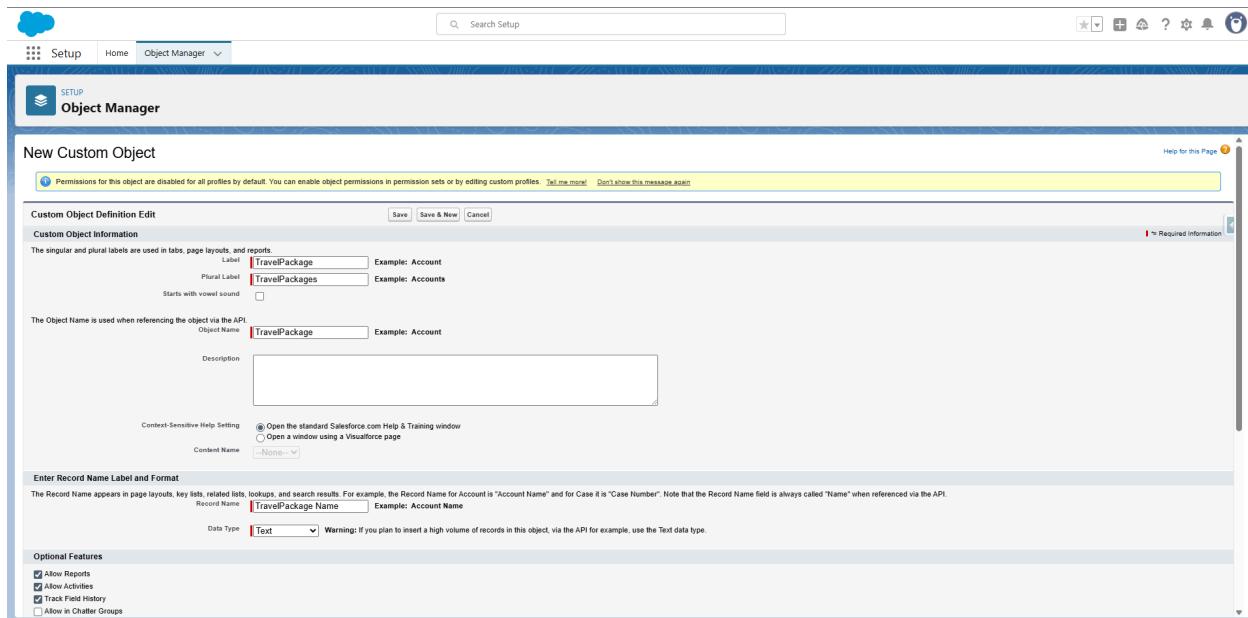
The screenshot shows the 'Booking' object details page in the Object Manager. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main 'Details' tab is selected, showing fields such as API Name (Booking__c), Singular Label (Booking), and Plural Label (Bookings). On the right, under 'Enable Reports', 'Track Activities', and 'Track Field History', several checkboxes are checked. Deployment status is listed as 'Deployed' with 'Help Settings' pointing to the Standard salesforce.com Help Window.

BookingGuest Object

The screenshot shows the 'Object Manager' section of the Salesforce Setup. A message at the top states: 'Permissions for this object are disabled for all profiles by default. You can enable object permissions in permission sets or by editing custom profiles.' Below this, the 'Custom Object Definition Edit' form is displayed. The 'Custom Object Information' section includes fields for 'Label' (BookingGuest), 'Plural Label' (BookingGuests), and 'Object Name' (BookingGuest). The 'Description' field is empty. Under 'Context-Sensitive Help Setting', there are two options: 'Open the standard Salesforce.com Help & Training window' (selected) and 'Open a window using a Visualforce page'. The 'Content Name' field contains 'None'. The 'Enter Record Name Label and Format' section specifies 'Record Name' (BookingGuest Name) and 'Data Type' (Text). The 'Optional Features' section has several checkboxes checked: 'Allow Reports', 'Allow Activities', 'Track Field History', and 'Allow in Chatter Groups'. At the bottom right of the form are 'Save', 'Save & New', and 'Cancel' buttons.

The screenshot shows the 'BookingGuest' object details in the Salesforce Object Manager. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main 'Details' tab is selected. The 'Fields & Relationships' section shows the API name 'BookingGuest_c'. The 'Custom' section shows the singular label 'BookingGuest' and plural label 'BookingGuests'. The 'Edit' and 'Delete' buttons are located at the top right of the details panel.

TravelPackage Object

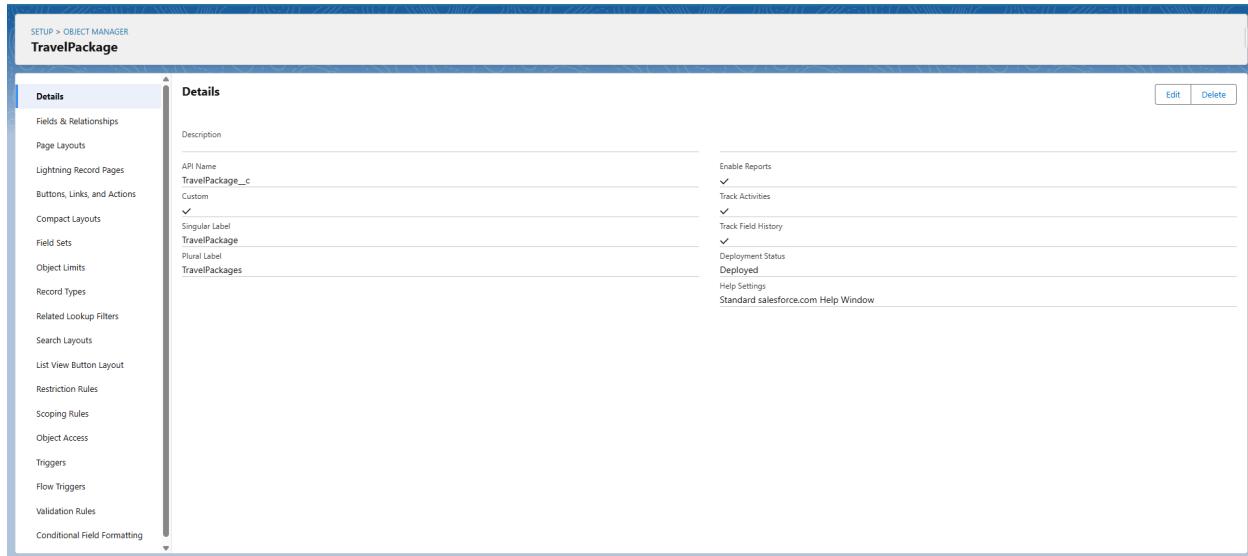


The screenshot shows the "Object Manager" section of the Salesforce Setup interface. A message at the top indicates that permissions for this object are disabled by default. The "Custom Object Definition Edit" screen is displayed, showing the following details:

- Custom Object Information:**
 - Label: TravelPackage (Example: Account)
 - Plural Label: TravelPackages (Example: Accounts)
 - Starts with vowel sound:
- The Object Name is used when referencing the object via the API:**
 - Object Name: TravelPackage (Example: Account)
- Description:** (Empty text area)
- Context-Sensitive Help Setting:**
 - Open the standard Salesforce.com Help & Training window
 - Open a window using a Visualforce page
- Content Name:** None
- Enter Record Name Label and Format:**

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

 - Record Name: TravelPackage Name (Example: Account Name)
 - Data Type: Text (Warning: If you plan to insert a high volume of records in this object, via the API for example, use the Text data type.)
- Optional Features:**
 - Allow Reports
 - Allow Activities
 - Track Field History
 - Allow in Chatter Groups



The screenshot shows the "TravelPackage" object details in the Salesforce Object Manager. The left sidebar lists various configuration tabs: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main "Details" tab is selected, displaying the following information:

Field	Value
Description	(Empty)
API Name	TravelPackage_c
Custom	✓
Singular Label	TravelPackage
Plural Label	TravelPackages
Enable Reports	✓
Track Activities	✓
Track Field History	✓
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

Booking Payment Object

The screenshot shows the 'Object Manager' section of the Salesforce Setup. A message at the top indicates that permissions are disabled for all profiles by default. The 'Custom Object Definition Edit' screen is displayed, with the object name set to 'Booking Payment'. Fields include 'Label' (Booking Payment), 'Plural Label' (Booking Payments), and 'Object Name' (Booking_Payment). A 'Description' field is empty. Under 'Context-Sensitive Help Setting', the 'Open a standard Salesforce.com Help & Training window' option is selected. The 'Record Name' is set to 'Booking Payment Name'. The 'Data Type' is 'Text'. Under 'Optional Features', 'Allow Reports' and 'Allow Activities' are checked. The right side of the screen shows a sidebar with various setup categories.

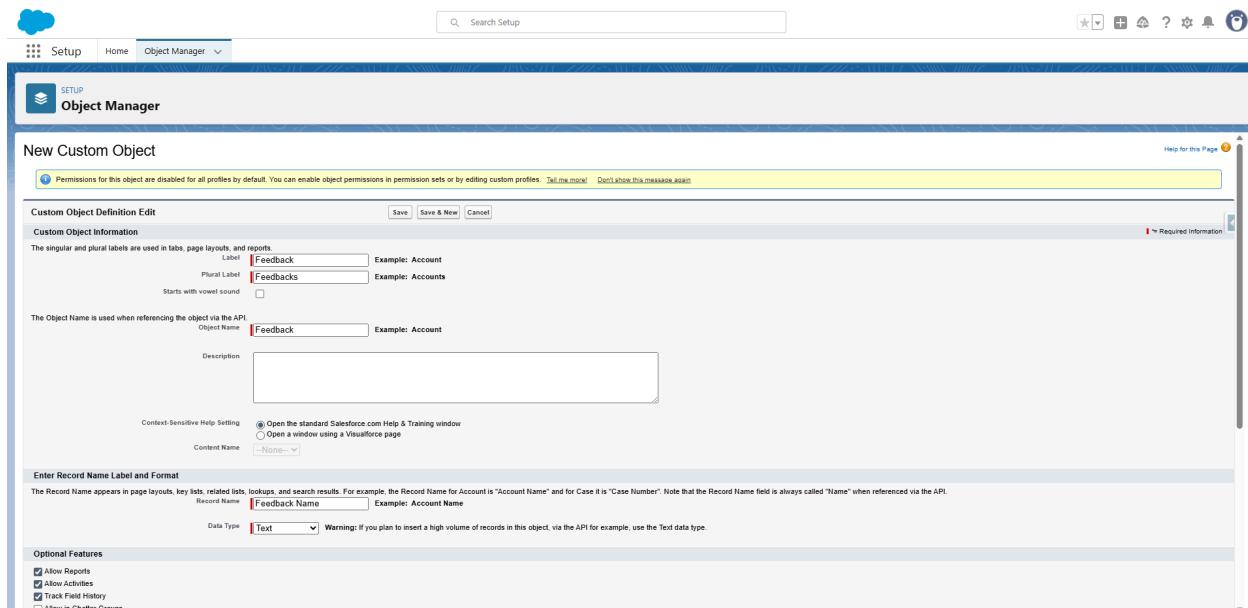
The screenshot shows the 'Details' page for the 'Booking Payment' object. The left sidebar lists various configuration tabs: Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main 'Details' section shows the API name 'Booking_Payment_c', a 'Custom' checkbox, and singular/plural labels. On the right, under 'Enable Reports', 'Track Activities', and 'Track Field History', checkboxes for 'Deployment Status' and 'Deployed' are checked. The 'Help Settings' field contains the URL 'Standard salesforce.com Help Window'.

Employee Object

The screenshot shows the 'Object Manager' section of the Salesforce Setup. A message at the top indicates that permissions are disabled by default. The 'Custom Object Definition Edit' screen is displayed, with the 'Label' field set to 'Employee' and the 'Plural Label' field set to 'Employees'. The 'Object Name' field is also set to 'Employee'. The 'Description' field is empty. Under 'Optional Features', the 'Allow Reports' checkbox is checked. The 'Record Name' field is set to 'Employee Name'. The 'Data Type' is selected as 'Text'. The 'API Name' is 'Booking_Payment__c'. The 'Singular Label' is 'Booking Payment' and the 'Plural Label' is 'Booking Payments'. The 'Content Name' is set to 'None'. The 'Context-Sensitive Help Setting' is set to open the standard Salesforce.com Help & Training window.

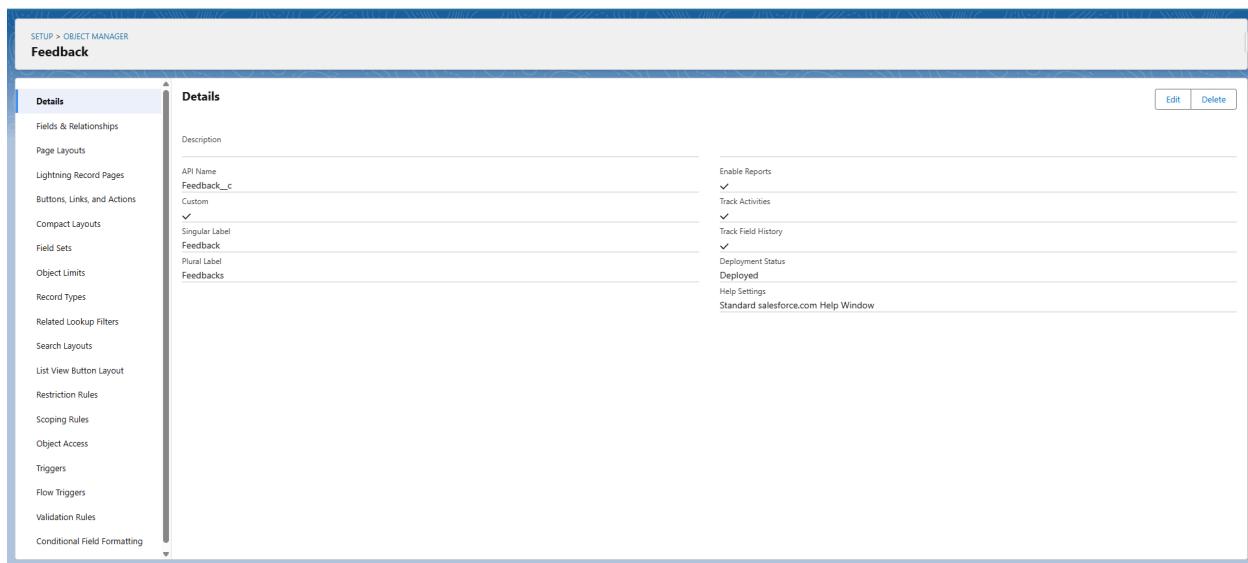
The screenshot shows the 'Details' tab for the 'Booking Payment' object. The 'Description' field is empty. The 'API Name' is 'Booking_Payment__c'. The 'Singular Label' is 'Booking Payment' and the 'Plural Label' is 'Booking Payments'. On the right side, under 'Optional Features', the following checkboxes are checked: 'Enable Reports', 'Track Activities', 'Track Field History', and 'Deployed'. The 'Deployment Status' is 'Deployed'. The 'Help Settings' link points to the 'Standard salesforce.com Help Window'.

Feedback Object



The screenshot shows the 'Object Manager' section of the Salesforce Setup. A new custom object is being created with the following details:

- Label:** Feedback
- Plural Label:** Feedbacks
- Description:** (Empty)
- Object Name:** Feedback
- Record Name:** Feedback Name
- Data Type:** Text
- Optional Features:** Allow Reports, Allow Activities, Track Field History, Allow in Chatter Groups (all checked)



The 'Details' tab of the Feedback object page displays the following configuration:

- Description:** (Empty)
- API Name:** Feedback_c
- Singular Label:** Feedback
- Plural Label:** Feedbacks
- Enable Reports:** ✓
- Track Activities:** ✓
- Track Field History:** ✓
- Deployment Status:** Deployed
- Help Settings:** Standard salesforce.com Help Window

In Salesforce, objects are like the database tables in MySQL. They store the data and act as the foundation of the CRM's data model. Each object helps organize and manage the types of info that is relevant to the Tours and Travels CRM app. This Milestone helped me understand how Salesforce objects are made and how they form the backbone of the app's data model. By creating the different custom objects, I can somewhat visualize how they could be connected together in the Tours and Travels CRM.

Milestone 3: Tabs

Activity 1: Creating a tab for Customer Info Object

Edit Custom Object Tab
Customers Info

Fill in the fields below to define the custom tab.

Custom Tab Definition Edit

Custom Object Tab Information

Tab Label	Customers Info
Object	Customer Info
Tab Style	Pencil

(Optional) Choose a Home Page Custom Link to show as a splash page the first time your users click on this tab.
Splash Page Custom Link:

Enter a short description.

Description:

Activity 2: Creating Remaining Tabs

Custom Object Tabs		<input type="button" value="New"/> What Is This?
Action	Label	Tab Style
Edit Del	<u>BookingGuests</u>	Safe
Edit Del	<u>Booking Payments</u>	Treasure chest
Edit Del	<u>Bookings</u>	Books
Edit Del	<u>Customers Info</u>	Pencil
Edit Del	<u>Employees</u>	Chess piece
Edit Del	<u>Feedbacks</u>	Mail
Edit Del	<u>TravelPackages</u>	Airplane

In Salesforce, the Tabs are the main entry points to access the object's data. They appear in the navigation bar of the application. Creating tabs will ensure that Users can easily access what they need. For this milestone, I learned how the creation of tabs brought the objects to life. Until now, the objects only existed in the background, but with tabs, each of them became directly accessible to Users. It took me some time to pick the correct tab style for each object, it was fun though!

Milestone 4: Fields & Relationships

Picklist Value Sets

The screenshot shows the 'Picklist Value Sets' page in the Salesforce Setup interface. At the top, there's a header with the page title and a 'Help for this Page' link. Below the header, a sub-header reads 'Global Value Sets'. A note states: 'Global picklist value sets let you share the values across objects. Base custom picklist fields on a global value set to inherit its values. The value set is restricted so users can't add unapproved values through the API.' Below this, there are buttons for 'View' and 'Create New View'. A 'New' button is located at the top right of the main table area. The main content is a table titled 'Global Value Sets' with columns for 'Action', 'Label', and 'Description'. The table lists several global value sets, each with an 'Edit' or 'Delete' link. The listed items include: City, Country, Languages, MemberShip, PackageType, PreferredAccommodation, and TripType. At the bottom of the table, there's a link 'Delete Global Value Sets (0)'.

Activity 1: Creation of fields for the Customer Info object

Customer Info Object Fields

The screenshot shows the 'Fields & Relationships' section for the 'Customer Info' object in the Object Manager. On the left, a sidebar lists various configuration options: Details, Fields & Relationships (selected), Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, Search Layouts, List View Button Layout, Restriction Rules, Scoping Rules, Object Access, Triggers, Flow Triggers, Validation Rules, and Conditional Field Formatting. The main area displays a table titled 'Fields & Relationships' with 10 items, sorted by Field Label. The columns are: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The data includes:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Age	Age__c	Formula (Number)		
City	City__c	Picklist		
Country	Country__c	Picklist		
Created By	CreatedById	Lookup(User)		
Customer Name	Name	Text(80)		✓
Date Of Birth	DateOfBirth__c	Date		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓
Phone	Phone__c	Phone		

Activity 2: Creation of fields for the Booking Object

Booking Object Fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Accommodation Amount per Person per Day	Accommodation_Amount_per_Person_per_Day__c	Formula (Currency)		
Lightning Record Pages	Approval Status	Approval_Status__c	Picklist		
Buttons, Links, and Actions	Booking Date	Booking_Date__c	Date		
Compact Layouts	Booking Name	Name	Text(80)		✓
Field Sets	Booking Number	Booking_Number__c	Auto Number		
Object Limits	Booking Status	Booking_Status__c	Picklist		
Record Types	Cancel Confirmation	Cancel_Confirmation__c	Checkbox		
Related Lookup Filters	Cancellation Date	Cancellation_Date__c	Date		
Search Layouts	Cancellation Reason	Cancellation_Reason__c	Text Area(255)		
List View Button Layout	Created By	CreatedBy	Lookup(User)		
Restriction Rules	Customer	Customer__c	Lookup(Customer Info)		✓
Scoping Rules	Customer Email	Customer_Email__c	Email		
Object Access	Guide Assigned	Guide_Assigned__c	Lookup(Employee)		✓
Triggers	Include Travel Insurance	Include_Travel_Insurance__c	Checkbox		
Flow Triggers	Last Modified By	LastModifiedBy	Lookup(User)		
Validation Rules	Membership Chosen	Membership_Chosen__c	Picklist		
Conditional Field Formatting					

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Membership Chosen	Membership_Chosen__c	Picklist		
Lightning Record Pages	No of Booking Guests Info Available	No_of_Booking_Guests_Info_Available__c	Roll-Up Summary (COUNT BookingGuest)		
Buttons, Links, and Actions	Number of Travelers	Number_of_Travelers__c	Number(18, 0)		
Compact Layouts	Owner	OwnerId	Lookup(User/Group)		✓
Field Sets	Preferred Accommodation	Preferred_Accommodation__c	Picklist		
Object Limits	Preferred Guide Language	Preferred_Guide_Language__c	Picklist		
Record Types	Require Tour Guide	Require_Tour_Guide__c	Checkbox		
Related Lookup Filters	Require Visa Assistance	Require_Visa_Assistance__c	Checkbox		
Search Layouts	Total Accommodation Amount	Total_Accommodation_Amount__c	Formula (Currency)		
List View Button Layout	Total Billing Amount	Total_Billing_Amount__c	Formula (Currency)		
Restriction Rules	Total Travel Amount	Total_Travel_Amount__c	Formula (Currency)		
Scoping Rules	Travel Cost Per Person	Travel_Cost_Per_Person__c	Formula (Currency)		
Object Access	Travel Package	Travel_Package__c	Lookup(TravelPackage)		
Triggers	Travelling End Date	Travelling_End_Date__c	Formula (Date)		
Flow Triggers	Travelling Start Date	Travelling_Start_Date__c	Date		
Validation Rules	Trip Type	Trip_Type__c	Picklist		
Conditional Field Formatting					

Activity 3: Creation of fields for the Booking Guest Object

Booking Guest Object Fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Age	Age__c	Number(18, 0)		
Lightning Record Pages	Age Category	Age_Category__c	Formula (Text)		
Buttons, Links, and Actions	Booking	Booking__c	Master-Detail(Booking)		
Compact Layouts	BookingGuest Name	Name	Text(80)		
Field Sets	City	City__c	Picklist		
Object Limits	Country	Country__c	Picklist		
Record Types	Created By	CreatedById	Lookup(User)		
Related Lookup Filters	Gender	Gender__c	Picklist		
Search Layouts	Last Modified By	LastModifiedById	Lookup(User)		
List View Button Layout	Passport Number	Passport_Number__c	Text(11)		
Restriction Rules	Relation with Customer	Relation_with_Customer__c	Picklist		
Scoping Rules	Special Needs	Special_Needs__c	Long Text Area(22768)		
Object Access	Visa Required	Visa_Required__c	Checkbox		
Triggers					
Flow Triggers					
Validation Rules					
Conditional Field Formatting					

Activity 4: Creation of fields for the TravelPackage Object

TravelPackage Object Fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Additional Notes	Additional_Notes__c	Text Area(255)		
Lightning Record Pages	Availability Status	Availability_Status__c	Picklist		
Buttons, Links, and Actions	Average Rating	Average_Rating__c	Number(18, 0)		
Compact Layouts	Country	Country__c	Picklist		
Field Sets	Created By	CreatedById	Lookup(User)		
Object Limits	Duration in Days	Duration_in_Days__c	Number(18, 0)		
Record Types	Guide Included	Guide_Included__c	Checkbox		
Related Lookup Filters	Guide LanguagesPlaces Covered	Guide_LanguagesPlaces_Covered__c	Picklist (Multi-Select)		
Search Layouts	Insurance Included	Insurance_Included__c	Checkbox		
List View Button Layout	Last Modified By	LastModifiedById	Lookup(User)		
Restriction Rules	Maximum Group Size	Maximum_Group_Size__c	Number(18, 0)		
Scoping Rules	Meals Included	Meals_Included__c	Picklist		
Object Access	Membership	Membership__c	Picklist (Multi-Select)		
Triggers	Owner	OwnerId	Lookup(User/Group)		
Flow Triggers	Package Type	Package_Type__c	Picklist (Multi-Select)		
Validation Rules	Places Covered	Places_Covered__c	Text Area(255)		
Conditional Field Formatting					

Fields & Relationships			
	Field Label	Type	
Page Layouts	Guide Included	Guide_Included__c	Checkbox
Lightning Record Pages	Guide LanguagesPlaces Covered	Guide_LanguagesPlaces_Covered__c	Picklist (Multi-Select)
Buttons, Links, and Actions	Insurance Included	Insurance_Included__c	Checkbox
Compact Layouts	Last Modified By	LastModifiedById	Lookup(User)
Field Sets	Maximum Group Size	Maximum_Group_Size__c	Number(18, 0)
Object Limits	Meals Included	Meals_Included__c	Picklist
Record Types	Membership	Membership__c	Picklist (Multi-Select)
Related Lookup Filters	Owner	OwnerId	Lookup(User, Group)
Search Layouts	Package Type	Package_Type__c	Picklist (Multi-Select)
List View Button Layout	Places Covered	Places_Covered__c	Text Area(255)
Restriction Rules	Preferred Guide Language	Preferred_Guide_Language__c	Picklist
Scoping Rules	Price Per Person	Price_Per_Person__c	Currency(16, 2)
Object Access	Region	Region__c	Picklist
Triggers	Transportation Modes	Transportation_Modes__c	Picklist (Multi-Select)
Flow Triggers	TravelPackage Name	Name	Text(80)
Validation Rules	Visa Assistance	Visa_Assistance__c	Checkbox

Activity 5: Creation of fields for the Employee Object

Employee Object Fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Address	Address__c	Text Area(255)		
Lightning Record Pages	Assigned Region	Assigned_Region__c	Picklist		
Buttons, Links, and Actions	Availability Status	Availability_Status__c	Picklist		
Compact Layouts	City	City__c	Picklist		
Field Sets	Country	Country__c	Picklist		
Object Limits	Created By	CreatedBy	Lookup(User)		
Record Types	Department	Department__c	Picklist		
Related Lookup Filters	Email	Email__c	Email		
Search Layouts	Employee ID	Employee_ID__c	Auto Number		
List View Button Layout	Employee Name	Name	Text(80)		
Restriction Rules	Employment Type	Employment_Type__c	Picklist		
Scoping Rules	Joining Date	Joining_Date__c	Date		
Object Access	Languages Spoken	Languages_Spoken__c	Picklist (Multi-Select)		
Triggers	Last Modified By	LastModifiedBy	Lookup(User)		
Flow Triggers	Owner	OwnerId	Lookup(User, Group)		
Validation Rules	Phone	Phone__c	Phone		

Fields & Relationships			
	FIELD NAME	DATA TYPE	
Page Layouts	City	City__c	Picklist
Lightning Record Pages	Country	Country__c	Picklist
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)
Compact Layouts	Department	Department__c	Picklist
Field Sets	Email	Email__c	Email
Object Limits	Employee ID	Employee_ID__c	Auto Number
Record Types	Employee Name	Name	Text(80)
Related Lookup Filters	Employment Type	Employment_Type__c	Picklist
Search Layouts	Joining Date	Joining_Date__c	Date
List View Button Layout	Languages Spoken	Languages_Spoken__c	Picklist (Multi-Select)
Restriction Rules	Last Modified By	LastModifiedById	Lookup(User)
Scoping Rules	Owner	OwnerId	Lookup(User/Group)
Object Access	Phone	Phone__c	Phone
Triggers	Profile Picture	Profile_Picture__c	Rich Text Area(32768)
Flow Triggers	Role	Role__c	Picklist
Validation Rules	Salary	Salary__c	Currency(16, 2)

Activity 6: Creation of fields for the Booking Payment Object

Booking Payment Object Fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Booking	Booking__c	Lookup(Booking)		✓
Lightning Record Pages	Booking Payment Name	Name	Text(80)		✓
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)		
Compact Layouts	Customer Name	Customer_Name__c	Formula (Text)		
Field Sets	Last Modified By	LastModifiedById	Lookup(User)		
Object Limits	Notes	Notes__c	Text Area(255)		
Record Types	Owner	OwnerId	Lookup(User/Group)		✓
Related Lookup Filters	Payment Date	Payment_Date__c	Date		
Search Layouts	Payment Method	Payment_Method__c	Picklist		
List View Button Layout	Payment Mode Details	Payment_Mode_Details__c	Text Area(255)		
Restriction Rules	Payment Number	Payment_Number__c	Auto Number		
Scoping Rules	Payment Receipt Sent	Payment_Receipt_Sent__c	Checkbox		
Object Access	Payment Reference Number	Payment_Reference_Number__c	Text(50)		
Triggers	Payment Status	Payment_Status__c	Picklist		
Flow Triggers	Total Bill Amount	Total_Bill_Amount__c	Formula (Currency)		

Feedback Object fields

Fields & Relationships					
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Accommodation Rating	Accommodation_Rating__c	Picklist		✓
Lightning Record Pages	Booking	Booking__c	Lookup(Booking)		✓
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)		✓
Compact Layouts	Customer	Customer__c	Lookup(Customer)		✓
Field Sets	Feedback Date	Feedback_Date__c	Date		✓
Object Limits	Feedback Name	Name	Text(80)		✓
Record Types	Feedback No	Feedback_No__c	Auto Number		✓
Related Lookup Filters	Last Modified By	LastModifiedById	Lookup(User)		✓
Search Layouts	Overall Rating	Overall_Rating__c	Picklist		✓
List View Button Layout	Owner	OwnerId	Lookup(User/Group)		✓
Restriction Rules	Package Satisfaction	Package_Satisfaction__c	Picklist		✓
Scoping Rules	Response Status	Response_Status__c	Picklist		✓
Object Access	Suggestions / Comments	Suggestions_Comments__c	Long Text Area(32768)		✓
Triggers	Tour Guide Rating	Tour_Guide_Rating__c	Picklist		✓
Flow Triggers	Transport Rating	Transport_Rating__c	Picklist		✓
Validation Rules					
Conditional Field Formatting					

In Salesforce, the fields and relationships are like the primary key, unique key, and the foreign key in MySQL. They determine how the data is captured, stored, and connected across the system. I also made use of the Picklist value sets in order to standardize the values across multiple objects. It ensures that the options are consistent and reduces redundancy.

This milestone is very important as it transformed the custom objects into meaningful data. By defining the custom fields and relationships, I built the structure of how information will flow through the CRM. It felt like creating an ERD where I connect the dots between the Customer Info, Bookings, Payments, and Feedback.

Milestone 5: Field Dependencies

Activity 1: Creating a Field Dependency in BookingGuest Object

The screenshot shows the Salesforce Setup interface with the 'Object Manager' tab selected. A modal window titled 'BookingGuest Field Dependencies' is open, showing two dependent picklist tables. The first table lists cities in India, Philippines, Saudi Arabia, and USA. The second table lists cities in Saudi Arabia, USA, and Switzerland. The 'Country' column is bolded in both tables. The 'City' column is bolded in the second table. The 'Philippines' row in the first table is highlighted with yellow. The 'Switzerland' row in the second table is highlighted with yellow. The 'Legend' at the top right indicates that yellow cells represent 'Included Value'.

BookingGuest Field Dependencies

[Back to Custom Object: BookingGuest](#)

This page allows you to define dependencies between fields (e.g., dependent picklists).

Field Dependencies			New
Action	Controlling Field	Dependent Field	Modified By
Edit Delete	Country	City	Andreas Luy , 8/23/2025, 10:29 AM

Activity 2: Creating a Field Dependency in Booking Object

The screenshot shows the 'Edit Field Dependency' page in the Salesforce Setup. The 'Controlling Field' is 'Membership Chosen' and the 'Dependent Field' is 'Preferred Accommodation'. The dependency grid lists several accommodation types: Hotel, Resort, Villa, Cabin, Cottage, Luxury Hotel, and No Preference. Each row has an 'Include Value' or 'Exclude Value' button.

Booking Field Dependencies

[Back to Custom Object: Booking](#)

This page allows you to define dependencies between fields (e.g., dependent picklists).

Field Dependencies			
Action	Controlling Field	Dependent Field	Modified By
Edit Del	Membership Chosen	Preferred Accommodation	Andreas Luy, 8/23/2025, 10:31 AM

Activity 3: Creating a Field Dependency in Employee Object

The screenshot shows the 'Edit Field Dependency' page in the Salesforce Setup. The 'Controlling Field' is 'Department' and the 'Dependent Field' is 'Role'. The dependency grid lists several roles: Travel Agent, Finance Executive, Support Staff, Customer Service Executive, Operations Supervisor, Admin, Logistics Coordinator, Accommodation Manager, and Transportation Manager. Each role has an 'Include Value' or 'Exclude Value' button.

Employee Field Dependencies

[« Back to Custom Object: Employee](#)

This page allows you to define dependencies between fields (e.g., dependent picklists).

Field Dependencies			New
Action	Controlling Field	Dependent Field	Modified By
Edit Del	Department	Role	Andreas Luy , 8/23/2025, 10:33 AM

Activity 4: Creating another Field Dependency in Employee Object

The screenshot shows the Salesforce Setup interface for creating field dependencies. The main area displays two dependent picklist tables side-by-side. The left table has columns for City (Controlling Field) and Country (Dependent Field). The right table has columns for Controlling Field (Dependent Field) and Dependent Field (Modified By). Both tables show lists of cities and countries with checkboxes for selecting values.

City:		Country:	
Hyderabad	Mumbai	Hyderabad	Mumbai
Mumbai	Delhi	Mumbai	Delhi
Delhi	Bangalore	Delhi	Bangalore
Bangalore	Chennai	Bangalore	Chennai
Chennai	Alaminos	Chennai	Alaminos
Alaminos	Angono City	Alaminos	Angono City
Angono City	Pasig City	Angono City	Pasig City
Pasig City	Quezon City	Pasig City	Quezon City
Quezon City	San Juan	Quezon City	San Juan
San Juan	Riyadh	San Juan	Riyadh
Riyadh	Jeddah	Riyadh	Jeddah
Jeddah	Tayma	Jeddah	Tayma
Tayma	Dammam	Tayma	Dammam
Dammam	Makkah	Dammam	Makkah
Makkah	New York	Makkah	New York
New York	Los Angeles	New York	Los Angeles
Los Angeles	Chicago	Los Angeles	Chicago
Chicago	Houston	Chicago	Houston
Houston	Columbus	Houston	Columbus
Columbus	Saudi Arabia	Columbus	Saudi Arabia
Saudi Arabia	USA	Saudi Arabia	USA
USA	Switzerland	USA	Switzerland
Switzerland	Geneva	Switzerland	Geneva
Geneva	Basel	Geneva	Basel
Basel	Lausanne	Basel	Lausanne
Lausanne	Bern	Lausanne	Bern
Bern	Luzern	Bern	Luzern
Luzern		Luzern	

Employee Field Dependencies

[« Back to Custom Object: Employee](#)

This page allows you to define dependencies between fields (e.g., dependent picklists).

Field Dependencies			New
Action	Controlling Field	Dependent Field	Modified By
Edit Del	Country	City	Andreas Luy , 8/23/2025, 10:35 AM
Edit Del	Department	Role	Andreas Luy , 8/23/2025, 10:33 AM

In Salesforce, the field dependencies are used to make the forms more dynamic by controlling which values appear in a dependent picklist that is also based on the controlling field.

While creating the dependencies across the different objects such as BookingGuest, Booking, and Employee. I noticed that the forms are much cleaner than before. Instead of leaving the Users feel intimidated with how many fields they should enter. It felt like a small change but had a big impact in terms of usability.

Milestone 6: Validation Rules

Activity 1: To create a Validation rules to a Customer Info Object

Customer Info Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit

Rule Name: Save Save & New Cancel

Active:

Description:

Error Condition Formula

Example: More Examples...
Display an error if Discount is more than 30%
If this formula expression is true, display the text defined in the Error Message area

Insert Field Insert Operator

Functions: ABS, ACOS, ADDMONTHS, AND, ASCII, ASIN
Insert Selected Function
ABS(number)
Returns the absolute value of a number: a number without its sign
Help on this function

Check Syntax: No errors found

Error Message

Example:
This message will appear when Error Condition formula is true

Error Message:

This error message can either appear at the top of the page or below a specific field on the page

Error Location: Top of Page Field [Phone]

Save Save & New Cancel

Help for this Page

Quick Tips: Operators & Functions

Customer Info Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit

Rule Name: Save Save & New Cancel

Active:

Description:

Error Condition Formula

Example: More Examples...
Display an error if Discount is more than 30%
If this formula expression is true, display the text defined in the Error Message area

Insert Field Insert Operator

Functions: ABS, ACOS, ADDMONTHS, AND, ASCII, ASIN
Insert Selected Function
ABS(number)
Returns the absolute value of a number: a number without its sign
Help on this function

Check Syntax: No errors found

Error Message

Example:
This message will appear when Error Condition formula is true

Error Message:

This error message can either appear at the top of the page or below a specific field on the page

Error Location: Top of Page Field [Email]

Save Save & New Cancel

Help for this Page

Quick Tips: Operators & Functions

Customer Info Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

The screenshot shows the 'Validation Rule Edit' interface. At the top, there are buttons for Save, Save & New, and Cancel. A 'Quick Tips' section with a link to 'Operators & Functions' is visible. The 'Rule Name' field contains 'Prevent_Future_DOB'. The 'Active' checkbox is checked. The 'Description' field is empty. In the 'Error Condition Formula' section, the formula 'Date_of_Birth > TODAY()' is entered. A dropdown menu for functions shows options like ABS, ACOS, ADOMONTHS, AND, ASCII, ASIN, etc. The 'Error Message' section contains the message 'Date of Birth cannot be in the future'. The 'Error Location' is set to 'Top of Page' at field 'Date Of Birth'. The bottom of the screen shows standard save buttons.

After adding the validations:

Validation Rules				
RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Email_Valid_Address	Email	Please Enter Valid Email Address	✓	Andreas Luy, 8/23/2025, 10:40 AM
Phone_Number_Must_10_Digits	Phone	Phone Number Must Be 10 Digits	✓	Andreas Luy, 8/23/2025, 10:39 AM
Prevent_Future_DOB	Date Of Birth	Date of Birth cannot be in the future	✓	Andreas Luy, 8/23/2025, 10:41 AM

Activity 2: To create a Validation rules to a BookingGuest Object

BookingGuest Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

The screenshot shows the 'Validation Rule Edit' interface. At the top, there are buttons for Save, Save & New, and Cancel. A 'Quick Tips' section with a link to 'Operators & Functions' is visible. The 'Rule Name' field contains 'Age_must_be_greater_than_0'. The 'Active' checkbox is checked. The 'Description' field is empty. In the 'Error Condition Formula' section, the formula 'Age <= 0' is entered. A dropdown menu for functions shows options like ABS, ACOS, ADOMONTHS, AND, ASCII, ASIN, etc. The 'Error Message' section contains the message 'Age must be greater than 0'. The 'Error Location' is set to 'Top of Page' at field 'Age'. The bottom of the screen shows standard save buttons.

Validation Rules					
1 items, Sorted by Rule Name					
RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY	
Age_must_be_greater_than_0	Age	Age must be greater than 0	✓	Andreas Luy, 8/23/2025, 10:43 AM	Edit

Activity 3: To create a Validation rules to an Employee Object

Employee Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit		Save Save & New Cancel	Quick Tips	
Rule Name	Email_is_mandatory		• Operators & Functions	
Active	<input checked="" type="checkbox"/>			
Description				
Error Condition Formula				
<p>Example: Discount_Percent_c>0.30 More Examples...</p> <p>Display an error if Discount is more than 30%</p> <p>If this formula expression is true, display the text defined in the Error Message area</p> <p>Insert Field Insert Operator ▾</p> <p>AND(OR(ISPICKVAL(Role__c, "Finance Executive"), ISPICKVAL(Role__c, "Admin")), ISBLANK(Email__c))</p> <p>Functions</p> <ul style="list-style-type: none"> - All Function Categories - ▾ ABS ACOS ADDMONTHS AND ASCII ASIN <p>Insert Selected Function</p> <p>ABS(number)</p> <p>Returns the absolute value of a number: a number without its sign</p> <p>Help on this function</p>				
<p>Check Syntax No errors found</p>				
Error Message				
<p>Example: Discount percent cannot exceed 30% This message will appear when Error Condition formula is true</p> <p>Error Message You Must Enter the Email</p> <p>This error message can either appear at the top of the page or below a specific field on the page</p> <p>Error Location <input type="radio"/> Top of Page <input checked="" type="radio"/> Field [Email] Help</p>				
<p>Save Save & New Cancel</p>				

Employee Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit		Save Save & New Cancel	Quick Tips	
Rule Name	Language_Must_be_Selected		• Operators & Functions	
Active	<input checked="" type="checkbox"/>			
Description				
Error Condition Formula				
<p>Example: Discount_Percent_c>0.30 More Examples...</p> <p>Display an error if Discount is more than 30%</p> <p>If this formula expression is true, display the text defined in the Error Message area</p> <p>Insert Field Insert Operator ▾</p> <p>AND(ISPICKVAL(Role__c, "Guide"), ISBLANK(Languages__c))</p> <p>Functions</p> <ul style="list-style-type: none"> - All Function Categories - ▾ ABS ACOS ADDMONTHS AND ASCII ASIN <p>Insert Selected Function</p> <p>ABS(number)</p> <p>Returns the absolute value of a number: a number without its sign</p> <p>Help on this function</p>				
<p>Check Syntax No errors found</p>				
Error Message				
<p>Example: Discount percent cannot exceed 30% This message will appear when Error Condition formula is true</p> <p>Error Message You Must Select Language</p> <p>This error message can either appear at the top of the page or below a specific field on the page</p> <p>Error Location <input type="radio"/> Top of Page <input checked="" type="radio"/> Field [Languages Spoken] Help</p>				
<p>Save Save & New Cancel</p>				

Validation Rules					
2 items, Sorted by Rule Name					
RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY	
Email_is_mandatory	Email	You Must Enter the Email	✓	Andreas Luy, 8/23/2025, 10:45 AM	Edit
Language_Must_be_Selected	Languages Spoken	You Must Select Language	✓	Andreas Luy, 8/23/2025, 10:45 AM	Edit

Activity 4: To create a Validation rules to a Booking Object.

Booking Validation Rule

Define a validation rule by specifying an error condition and a corresponding error message. The error condition is written as a Boolean formula expression that returns true or false. When the formula expression returns true, the save will be aborted and the error message will be displayed. The user can correct the error and try again.

Validation Rule Edit

Rule Name: Enforce_Pending_Status_On_Create

Active:

Description:

Error Condition Formula

Example: Discount_Percent_c>0.30 More Examples...
Display an error if Discount is more than 30%
If this formula expression is true, display the text defined in the Error Message area

Functions: All Function Categories

ABS
ACOS
ADDMONTHS
AND
ASIN
ASIN
Insert Selected Function
ABS(Number)
Returns the absolute value of a number, a number without its sign

Insert Field Insert Operator
AND
ISNUM()
NOT([SPICKVAL:Booking_Status__c, "Pending"])

Check Syntax No errors found

Error Message

Example: Discount percent cannot exceed 30%
This message will appear when Error Condition formula is true

Error Message: Booking Status must be 'Pending' when a new record is created.

This error message can either appear at the top of the page or below a specific field on the page

Error Location: Top of Page Field [Booking Status]

Save Save & New Cancel

In Salesforce, validation rules can help maintain data integrity by preventing the Users from saving records that do not meet the criteria. Ensuring that the information saved is accurate, complete, and consistent.

Upon creating the validation rules across different objects. It's important to note that we should always check if the API is correct as it can cause errors.

This activity helped me realize how important it is to check and check in order to avoid mistakes like incorrect values.

Milestone 7: Approval Process

Activity 1: Create an Approval Process for Booking Cancellation Approval

Classic Email Templates

Text Email Template
Booking Cancellation Approval Notification
Back to List: Profiles

Preview your email template below.

Email Template Detail

Email Templates from Salesforce	Unified Public Classic Email Templates
Email Template Name	Booking_Cancellation_Approval_Notification
Template Unique Name	Booking_Cancellation_Approval_Notification
Encoding	Unicode (UTF-8)
Author	Andreas.Lux [Change]
Description	
Created By	Andreas Lux 8/23/2025, 7:10 PM
Modified By	Andreas Lux 8/23/2025, 7:10 PM

Email Template

Send Test and Verify Merge Fields

Subject: Approval Needed: Booking ({Booking__c.Name})

Plain Text Preview

Hi {User.FullName},
You have a new account awaiting your approval.
Booking ID: {Booking__c.Id} ({Booking__c.TravelPackage__c})
Status: {Booking__c.Status}
Please log in to Salesforce and review the record.

Thanks,
Salesforce Approval

Attachments

Attach File

No records to display

Classic Email Templates

Text Email Template
Cancellation Approved
Back to List: Profiles

Preview your email template below.

Email Template Detail

Email Templates from Salesforce	Unified Public Classic Email Templates
Email Template Name	Cancellation_Approved
Template Unique Name	Cancellation_Approved
Encoding	Unicode (UTF-8)
Author	Andreas.Lux [Change]
Description	
Created By	Andreas Lux 8/23/2025, 7:13 PM
Modified By	Andreas Lux 8/23/2025, 7:13 PM

Email Template

Send Test and Verify Merge Fields

Subject: Cancellation Request Approved - Booking ({Booking__c.Name})

Plain Text Preview

Dear {Booking__c.Customer__c},
We have received and reviewed your cancellation request for the following booking:
Travel Package ({Booking__c.TravelPackage__c})
Booking ID: {Booking__c.Id}
Travel Start Date: {Booking__c.Travel_Start_Date__c}
We're confirming that your request has been APPROVED.
All funds will now be processed according to our policy. You will receive the refund shortly.
If you have any questions or need further assistance, feel free to contact our support team.
Best regards,
Travel Services Team.

Attachments

Attach File

No records to display

Text Email Template
Cancellation Rejected
[+ Back to List: Profiles](#)

Preview your email template below.

Email Template Detail

Email Templates from Salesforce	Unified Public Classic Email Templates
Email Template Name	Cancellation_Rejected
Template Unique Name	Cancellation_Rejected
Encoding	Unicode (UTF-8)
Author	Andreas Luy [Chaos]
Description	
Created By	Andreas Luy 8/23/2025, 7:15 PM
Modified By	Andreas Luy 8/23/2025, 7:15 PM

Email Template

Subject: Update on Your Cancellation Request – Booking [(Booking__c.Name)]
[Send Test and Verify Merge Fields](#)

Plain Text Preview:

```
Dear [(Booking__c.Customer__r)],

Thank you for submitting your cancellation request for:
-Travel Package [(Booking__c.TravelPackage__c)]
-Booking Number: [(Booking__c.Name)]
-Travel Date: [(Booking__c.Travel_Start_Date__c)]

We regret to inform you that your cancellation request has been REJECTED as it does not meet the required conditions.
You have the option to reschedule or cancel the booking if you are unable to travel on your scheduled trip.
If you have any questions or need further assistance, please contact your assigned travel agent.

Warm regards,
Travel Services Team
```

Attachments

No records to display

SETUP Approval Processes

Approval Processes
Booking: Booking Cancellation Approval
[+ Back to Approval Process List](#)

Process Definition Detail

Process Name	Booking Cancellation Approval	Active	<input checked="" type="checkbox"/>
Unique Name	Booking_Cancellation_Approval	Next Automated Approver Determined By	Manager of Record Submitter
Description	Process to approve booking cancellation		
Entry Criteria	(Booking: Booking_Status EQUALS Cancelled) AND (Booking: Cancel_Confirmation EQUALS True)		
Record Editability	Administrator OR Current Approver	Allow Submitters to Recall Approval Requests <input type="checkbox"/>	
Approval Assignment Email Template	Booking_Cancellation_Approval_Notification		
Initial Submitters	Booking Owner, Role: Travel Agent		
Created By	Andreas Luy 8/23/2025, 7:24 PM	Modified By Andreas Luy 8/23/2025, 7:37 PM	

Initial Submission Actions

Action	Type	Description
Record Lock		Lock the record from being edited
Edit Remove	Field Update	Set Booking_Status to Pending

Approval Steps

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions	1	Travel Agent Manager Approval			User: Michael Jackson	Final Rejection

Final Approval Actions

Action	Type	Description
Edit Remove	Record Lock	Lock the record from being edited
Edit Remove	Field Update	Set Approval_Status to Approved
Edit Remove	Email Alert	Send Email to Booking Customer

Final Rejection Actions

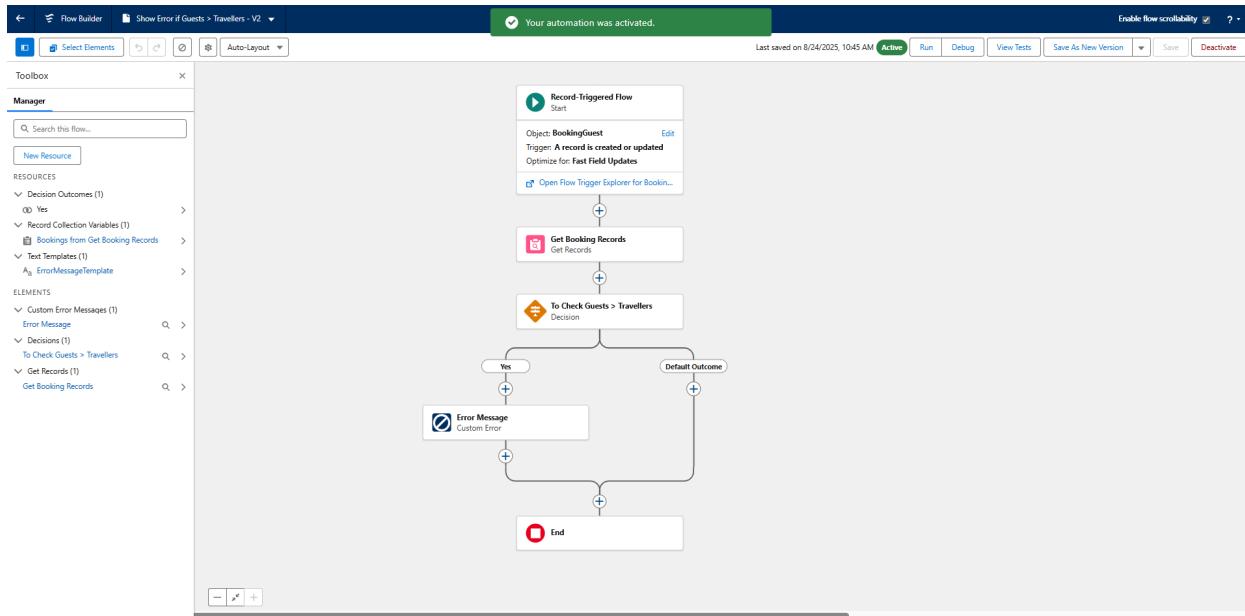
Action	Type	Description
Edit Remove	Record Lock	Unlock the record for editing
Edit Remove	Field Update	Booking_Status to Confirmed
Edit Remove	Field Update	Approval_Status to Rejected
Edit Remove	Email Alert	Send Approval Rejection Email to the User

In Salesforce, the approval process can streamline the record by automatically routing the submissions to designed approvers. Ensuring that the critical actions like booking, cancellation, and discounts are going through proper authorization before being finalized.

This milestone helped me understand the difference between creating the approval processes instead of staying in manual follow-up processes.

Milestone 8: Flows

Activity 1



In Salesforce, flows will allow the automation of a business using a visual drag-and-drop interface. They can update records, get records, send notifications, and even guide Users through screens without writing code.

Creating the flows gave me a deeper appreciation of Salesforce's no code automation. I could automate repetitive tasks like record updates and notifications quickly! It actually became one of my favorite features to work on in Salesforce.

Milestone 9: Workflows

Activity 1

The screenshot shows the Salesforce Setup interface under the 'Workflow Rules' section. A specific workflow rule, 'Follow-up Task After Booking Completion', is selected. The rule is triggered by the condition 'Booking: Booking Status equals Completed'. It is set to evaluate the rule when a record is created or edited. The rule has one immediate workflow action: a task titled 'Follow-up feedback' assigned to 'Andrea Lux'. A note at the bottom states that time-dependent actions cannot be added because the evaluation criteria is 'Every time a record is created or edited'.

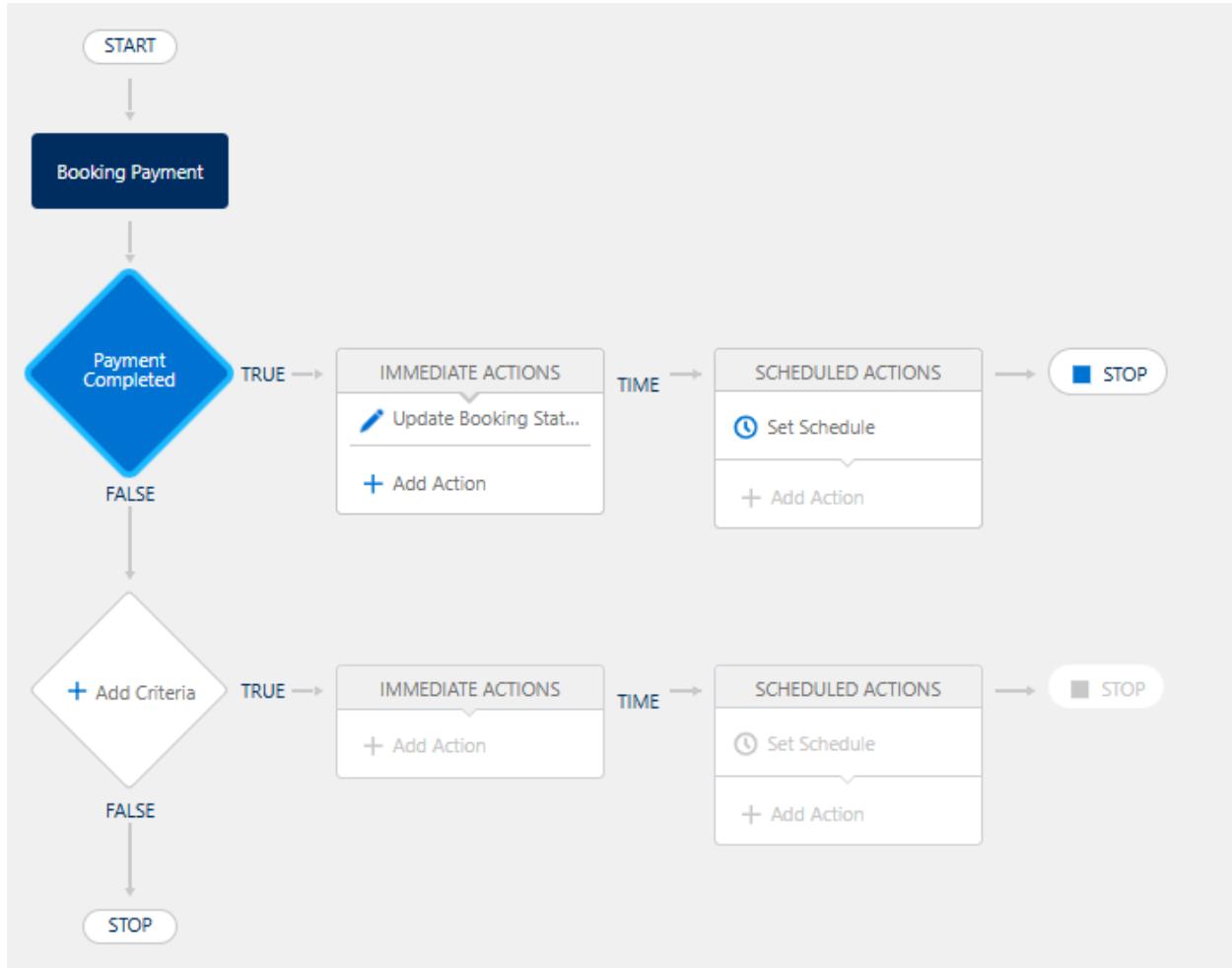
In Salesforce, the workflow automates repetitive tasks by triggering actions when specific conditions are met. They can update fields, send email alerts, and even create tasks which can help streamline the day-to-day processes.

Working with this milestone showed me how simple it is to make rules that could make a huge difference in efficiency. By automating the actions like field updates, I'll be able to reduce the manual effort and ensure that the important tasks are not going to be missed.

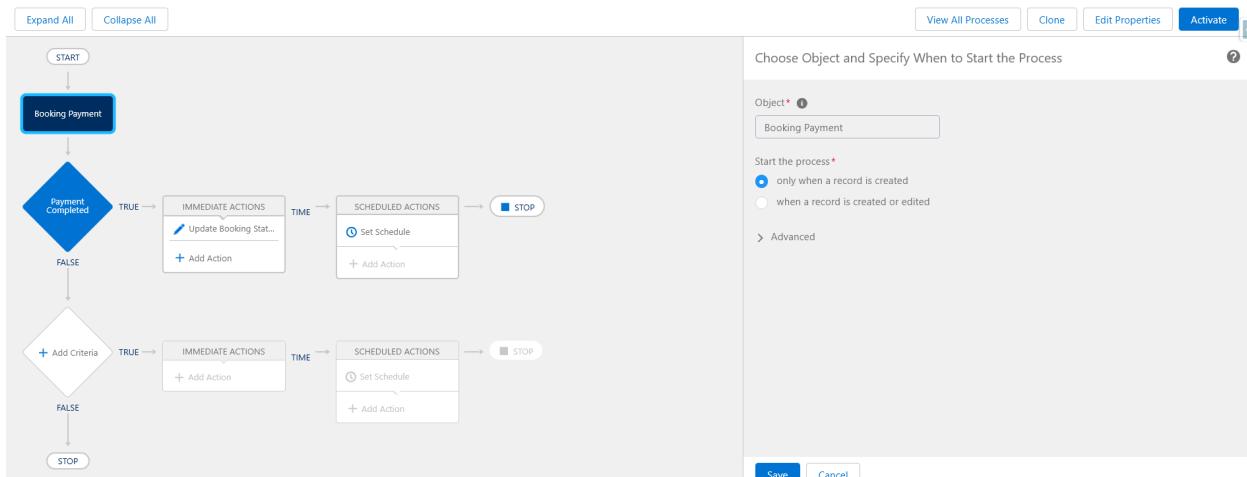
Milestone 10: Process Builder

Activity 1

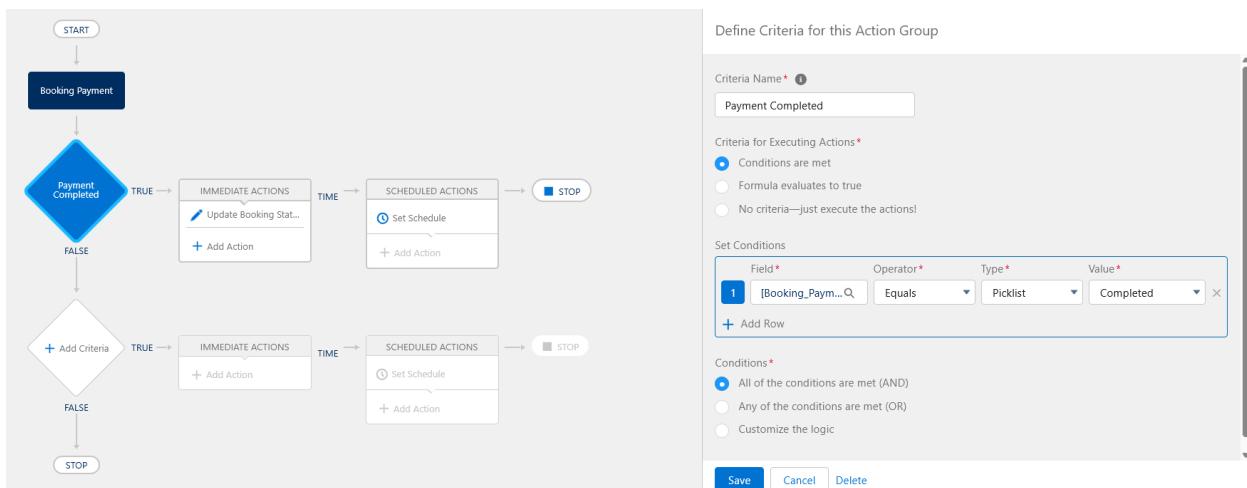
Entire Process



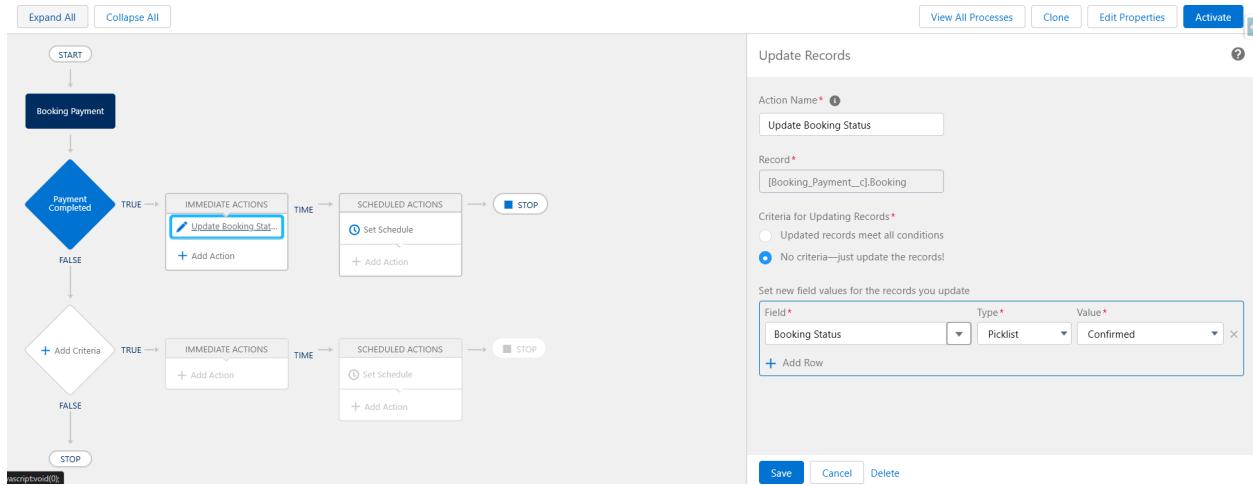
Booking Payment Object



Criteria



Immediate Actions



In Salesforce, the process builder is a point and click automation tool that uses the if/then logic to execute the actions within a single process. Building a process in Process Builder was confusing at first but it helped me gain an insight into how Salesforce automation evolved before flows. Even though Flows are my favorite, working with Process Builder was an important step in seeing how automation has evolved.

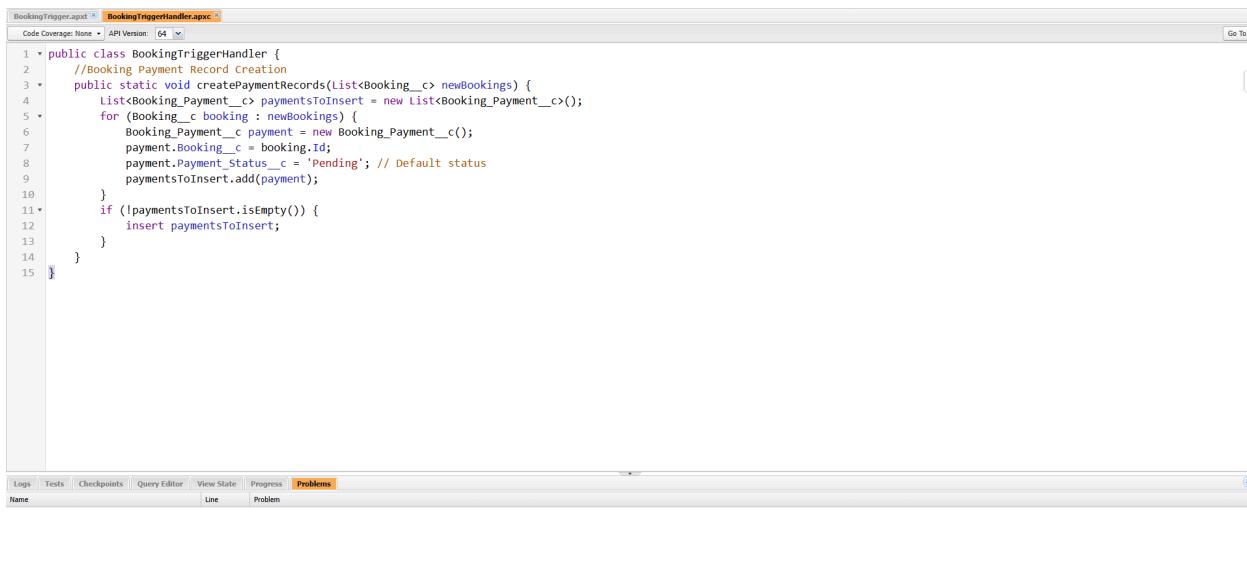
Milestone 11: Triggers

Activity 1 : Auto-create Booking Payment Record



The screenshot shows the Salesforce IDE interface with the BookingTrigger.apotr file open. The code defines a trigger named BookingTrigger on the Booking_c object. It checks if the trigger is after an insert and if the trigger is insert. If both conditions are met, it calls the createPaymentRecords method from the BookingTriggerHandler class.

```
1 * trigger BookingTrigger on Booking__c (after insert) {
2 *     if (Trigger.isAfter && Trigger.isInsert) {
3 *         // creates Booking Payment records
4 *         BookingTriggerHandler.createPaymentRecords(Trigger.new);
5 *     }
6 }
```



The screenshot shows the Salesforce IDE interface with the BookingTriggerHandler.apxc file open. The code defines a public class BookingTriggerHandler. It contains a static method createPaymentRecords that takes a list of new bookings as input. It initializes a list of payment records, iterates through each booking, creates a new payment record, sets its booking ID, and adds it to the list. Finally, it inserts all the payment records into the database if the list is not empty.

```
1 * public class BookingTriggerHandler {
2 *     //Booking Payment Record Creation
3 *     public static void createPaymentRecords(List<Booking__c> newBookings) {
4 *         List<Booking_Payment__c> paymentsToInsert = new List<Booking_Payment__c>();
5 *         for (Booking__c booking : newBookings) {
6 *             Booking_Payment__c payment = new Booking_Payment__c();
7 *             payment.Booking__c = booking.Id;
8 *             payment.Payment_Status__c = 'Pending'; // Default status
9 *             paymentsToInsert.add(payment);
10 *         }
11 *         if (!paymentsToInsert.isEmpty()) {
12 *             insert paymentsToInsert;
13 *         }
14 *     }
15 }
```

Activity 2 : Automate Guest Records Creation Based on Traveller Count

The screenshot shows two tabs open in the Salesforce code editor: 'BookingTrigger.apxc' and 'BookingTriggerHandler.apxc'. Both tabs have 'Code Coverage: None' and 'API Version: 64' set. The 'BookingTrigger.apxc' tab contains the following Apex code:

```
1 trigger BookingTrigger on Booking__c (after insert) {
2     if (Trigger.isAfter && Trigger.isInsert) {
3         // creates Booking Payment records
4         BookingTriggerHandler.createPaymentRecords(Trigger.new);
5
6         // creates BookingGuests records
7         BookingTriggerHandler.createBookingGuests(Trigger.new);
8     }
9 }
```

The 'BookingTriggerHandler.apxc' tab contains the following Apex code:

```
1 public class BookingTriggerHandler {
2     //Booking Payment Record Creation
3     public static void createPaymentRecords(List<Booking__c> newBookings) {
4         List<Booking_Payment__c> paymentsToInsert = new List<Booking_Payment__c>();
5
6         for (Booking__c booking : newBookings) {
7             Booking_Payment__c payment = new Booking_Payment__c();
8             payment.Booking__c = booking.ID;
9             payment.Payment_Status__c = 'Pending'; // Default status
10            paymentsToInsert.add(payment);
11        }
12        if (!paymentsToInsert.isEmpty()) {
13            insert paymentsToInsert;
14        }
15    }
16
17    //BookingGuests records Creation
18    public static void createBookingGuests(List<Booking__c> bookings) {
19        List<BookingGuest__c> guestsToInsert = new List<BookingGuest__c>();
20
21        for (Booking__c booking : bookings) {
22            Integer count = (Integer)booking.Number_of_Travelers__c;
23
24            for (Integer i = 1; i < count; i++) {
25                BookingGuest__c guest = new BookingGuest__c();
26                guest.Booking__c = booking.ID;
27            }
28        }
29    }
30 }
```

In Salesforce, the Apex triggers are pieces of code that run automatically before or after the database events such as insert, update, or delete. With triggers, the administrators and developers can gain a deeper control over how data is going to behave and interact across objects. Implementing the triggers for the Booking object gave me hands-on experience. It felt like a major step forward because it showed me how to go beyond clicks and use code to design accurate and customized workflows. It's more technical compared to the earlier milestones, but feels very rewarding when the state in the logs is Success!

Milestone 12: Asynchronous Apex

Activity 1 : Future Asynchronous Apex

The screenshot shows the code editor for the `BookingConfirmationEmailer` class. The code implements a static void method `sendBookingConfirmation` that takes a set of booking IDs. It queries the `Booking__c` object to get the names, customer emails, and total billing amounts. Then, it loops through each booking, creates a `Messaging.SingleEmailMessage`, sets the recipient to the customer email, sets the subject to 'Booking Confirmed', and sets the plain text body with a greeting, confirmation message, booking ID, name, total bill amount paid, and a thank you note. Finally, it adds the message to a list of emails and sends them if the list is not empty.

```
1 * public class BookingConfirmationEmailer {
2
3     @future(callout=false)
4
5     public static void sendBookingConfirmation(Set<Id> bookingIds) {
6         List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
7         List<Booking__c> bookings = [SELECT Id, Name, Customer_Email__c, Total_Billing_Amount__c
8             FROM Booking__c
9             WHERE Id IN :bookingIds];
10
11        for (Booking__c booking : bookings) {
12            if (String.isNotBlank(booking.Customer_Email__c)) {
13                Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
14                mail.setToAddresses(new String[] { booking.Customer_Email__c });
15                mail.setSubject('Booking Confirmed: ' + booking.Name);
16                mail.setPlainTextBody(
17                    'Dear Customer,' + '\n\n' +
18                    'Your booking has been confirmed. Please find the details below:\n' +
19                    'Booking ID: ' + booking.Name + '\n' +
20                    'Total Bill Amount Paid: $' + booking.Total_Billing_Amount__c + '\n\n' +
21                    'Thank you for booking with us!');
22            }
23            emails.add(mail);
24        }
25    }
26    if (!emails.isEmpty()) {
27        Messaging.sendEmail(emails);
28    }
29 }
30 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

The screenshot shows the code editor for the `BookingTrigger` apex trigger. It handles insert and update events on the `Booking__c` object. If the trigger is after an insert, it creates payment records using the `BookingTriggerHandler.createPaymentRecords` method. If the trigger is after an update, it creates booking guest records using the `BookingTriggerHandler.createBookingGuests` method. It then checks if the booking status changed to 'Confirmed'. If so, it retrieves the old booking record from the trigger's old map and adds its ID to a set of booking IDs to send. Finally, it checks if there are any booking IDs to send and calls the `BookingConfirmationEmailer.sendBookingConfirmation` method to send the emails.

```
1 * trigger BookingTrigger on Booking__c (after insert, after update) {
2
3     if (Trigger.isAfter && Trigger.isInsert) {
4         // creates Booking Payment records
5         BookingTriggerHandler.createPaymentRecords(Trigger.new);
6
7         // creates Booking Guests records
8         BookingTriggerHandler.createBookingGuests(Trigger.new);
9     }
10
11    else if (Trigger.isAfter && Trigger.isUpdate) {
12        Set<Id> bookingIdsToSend = new Set<Id>();
13
14        for (Booking__c booking : Trigger.new) {
15            Booking__c oldBooking = Trigger.oldMap.get(booking.Id);
16
17            // Check for status change to "Confirmed"
18            if (booking.Booking_Status__c == 'Confirmed' && oldBooking.Booking_Status__c != 'Confirmed') {
19                bookingIdsToSend.add(booking.Id);
20            }
21        }
22
23        if (!bookingIdsToSend.isEmpty()) {
24            // Call the Future Method
25            BookingConfirmationEmailer.sendBookingConfirmation(bookingIdsToSend);
26        }
27    }
28 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Activity 2 : Queueable and Schedulable Asynchronous Apex

The screenshot shows the Salesforce code editor with the tab 'BookingReminderQueueable.apc' selected. The code implements the Queueable interface and performs the following steps:

```
1 * public class BookingReminderQueueable implements Queueable {
2     List<Booking__c> bookingsToRemind;
3
4     public BookingReminderQueueable(List<Booking__c> bookings) {
5         this.bookingsToRemind = bookings;
6     }
7
8     public void execute(QueueableContext context) {
9         List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
10
11        for (Booking__c booking : bookingsToRemind) {
12            if (!String.isBlank(booking.Customer_Email__c)) {
13                Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
14                mail.setToAddresses(new String[] { booking.Customer_Email__c });
15                mail.setSubject('Reminder: Your Tour Starts Soon!');
16                mail.setPlainTextBody(
17                    'Hello,\n\nThis is a friendly reminder that your tour is starting on ' +
18                    booking.Travelling_Start_Date__c.format() +
19                    '. Please make necessary arrangements.\n\nThank you for choosing us!'
20                );
21                emails.add(mail);
22            }
23        }
24
25        if (!emails.isEmpty()) {
26            Messaging.sendEmail(emails);
27        }
28    }
29 }
```

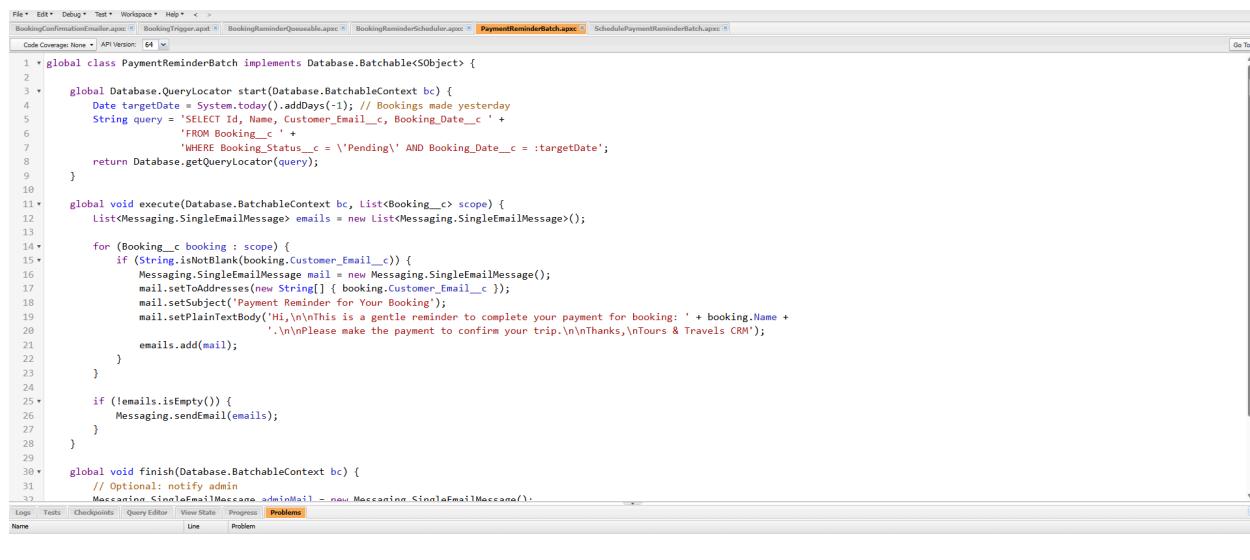
The code editor includes tabs for 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'ViewState', 'Progress', and 'Problems'. The 'Problems' tab is currently selected.

The screenshot shows the Salesforce code editor with the tab 'BookingReminderScheduler.apc' selected. The code implements the Schedulable interface and performs the following steps:

```
1 * public class BookingReminderScheduler implements Schedulable {
2     public void execute(SchedulableContext context) {
3         Date reminderDate = Date.today().addDays(3);
4         List<Booking__c> upcomingBookings = [
5             SELECT Id, Travelling_Start_Date__c, Customer_Email__c
6             FROM Booking__c
7             WHERE Travelling_Start_Date__c = :reminderDate
8             AND Booking_Status__c = 'Confirmed'
9         ];
10        if (!upcomingBookings.isEmpty()) {
11            System.enqueueJob(new BookingReminderQueueable(upcomingBookings));
12        }
13    }
14 }
```

The code editor includes tabs for 'Logs', 'Tests', 'Checkpoints', 'Query Editor', 'ViewState', 'Progress', and 'Problems'. The 'Problems' tab is currently selected.

Activity 3 : Batchable and Schedulable Asynchronous Apex



The screenshot shows the Salesforce IDE interface with the 'PaymentReminderBatch.apex' file open. The code implements the Database.Batchable<SObject> interface. It queries bookings made yesterday where the status is pending, then sends an email reminder to each customer. Finally, it sends an email to an admin. The code coverage is at 64%.

```
1 * global class PaymentReminderBatch implements Database.Batchable<SObject> {
2
3 *     global Database.QueryLocator start(Database.BatchableContext bc) {
4         Date targetDate = System.today().addDays(-1); // Bookings made yesterday
5         String query = 'SELECT Id, Name, Customer_Email__c, Booking_Date__c '
6             'FROM Booking__c '
7             'WHERE Booking_Status__c = \'Pending\' AND Booking_Date__c = :targetDate';
8         return Database.getQueryLocator(query);
9     }
10
11 *     global void execute(Database.BatchableContext bc, List<Booking__c> scope) {
12     List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
13
14     for (Booking__c booking : scope) {
15         if (String.isNotBlank(booking.Customer_Email__c)) {
16             Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
17             mail.setToAddresses(new String[] { booking.Customer_Email__c });
18             mail.setSubject('Payment Reminder for Your Booking');
19             mail.setPlainTextBody('Hi,\n\nThis is a gentle reminder to complete your payment for booking: ' + booking.Name +
20                 '.\n\nPlease make the payment to confirm your trip.\n\nThanks,\nTours & Travels CRM');
21             emails.add(mail);
22         }
23     }
24
25     if (!emails.isEmpty()) {
26         Messaging.sendEmail(emails);
27     }
28 }
29
30 *     global void finish(Database.BatchableContext bc) {
31     // Optional: notify admin
32     Mecanico.SingleEmailMessage adminMail = new Mecanico.SingleEmailMessage();
33 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Name Line Problem



The screenshot shows the Salesforce IDE interface with the 'SchedulePaymentReminderBatch.apex' file open. The code implements the Schedulable interface, executing a batch job every 200 seconds. The code coverage is at 64%.

```
1 * public class SchedulePaymentReminderBatch implements Schedulable {
2 *
3 *     public void execute(SchedulableContext sc) {
4         PaymentReminderBatch batch = new PaymentReminderBatch();
5         Database.executeBatch(batch, 200);
6     }
6 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Name Line Problem

The screenshot shows the Salesforce Apex Classes page. At the top, there's a header with a gear icon labeled "SETUP" and "Apex Classes". Below the header, a section titled "Apex Classes" is displayed. It includes a message about Apex Code usage: "Percent of Apex Used: 0.12% You are currently using 7,186 characters of Apex Code (excluding comments and @isTest annotated classes) in your organization, out of an allowed limit of 6,000,000 characters. Note that the amount in use includes both Apex Classes and Triggers defined in your organization." There are buttons for "Estimate your organization's code coverage" and "Compile all classes". A "View" dropdown is set to "All" with a "Create New View" option. A navigation bar at the top right includes links for A through Z and "Other".

Action	Name	Namespace Prefix	Api Version	Status	Size Without Comments	Last Modified By	Has Trace Flags
Edit Del Security	BookingConfirmationEmailer		64.0	Active	1,347	Andreas Luy	8/23/2025, 8:22 PM
Edit Del Security	BookingReminderQueueable		64.0	Active	1,171	Andreas Luy	8/23/2025, 8:28 PM
Edit Del Security	BookingReminderScheduler		64.0	Active	569	Andreas Luy	8/23/2025, 8:28 PM
Edit Del Security	BookingTriggerHandler		64.0	Active	1,188	Andreas Luy	8/23/2025, 8:12 PM
Edit Del Security	PaymentReminderBatch		64.0	Active	1,880	Andreas Luy	8/23/2025, 8:34 PM
Edit Del Security	SchedulePaymentReminderBatch		64.0	Active	232	Andreas Luy	8/23/2025, 8:36 PM

Dynamic Apex Classes

Dynamic Apex extends your programming reach by interacting with Lightning Platform components.

Class Name	Namespace Prefix	Api Version	Created By	Last Modified By
No records to display				

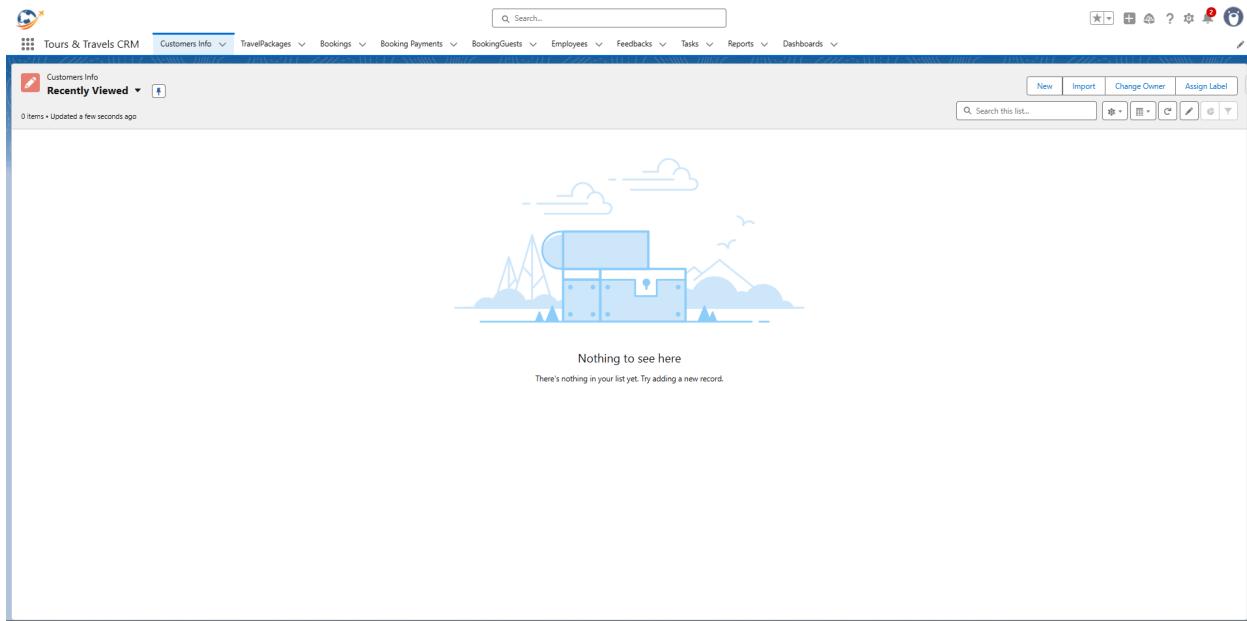
In Salesforce, Asynchronous Apex is designed to handle background tasks that are long-running or resource-intensive.

This milestone was one of the most technical yet rewarding parts of my journey. I implemented asynchronous processes such as automatically sending booking confirmation emails, scheduling tour reminders, and running daily payment reminder batches. It showed me a different way Salesforce can handle complex and large-scale operations efficiently.

Phase 3: UI/UX Development & Customization

Milestone 13: The Lightning App

Activity 1: Create a Lightning App for “ Tours & Travels CRM ”



In Salesforce, a Lightning App is a customizable collection of objects, tabs, and utilities. It provides a modern interface tailored to specific roles.

Creating the Tours & Travels CRM felt like bringing all the earlier requirements together! By including the custom objects I have created earlier together in a one unified space, it felt like I was able to design the app by myself. This is like the front door of the project as it gave the application a polished and professional look.

Milestone 14: Editing of Page Layouts

Activity 1: To edit a Page Layout in Customer Info Object

SETUP > OBJECT MANAGER
Customer Info

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers

Save ▾ Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields Quick Find Field Name

Section	Country	Date Of Birth	Phone
Blank Space	Created By	Email	
Age	Customer ID	Last Modified By	
City	Customer Name	Owner	

Actions in this section are currently inherited from the global publisher layout. You can [override the global publisher layout](#) to set a customized list of actions for the publisher pages that use this layout.

Salesforce Mobile and Lightning Experience Actions

Actions in this section are predefined by Salesforce. You can [override the predefined actions](#) to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions in the Quick Actions in the Salesforce Classic Publisher section, and have saved the layout, then this section inherits that set of actions by default when you click to override.

Customer Info Detail

Standard Buttons Edit Delete Clone Change Owner Change Record Type Printable View Sharing Sharing Hierarchy Edit Labels Custom Buttons

Information (Header visible on edit only)
Customer ID: GEN-2004-001234
Customer Name: Sample Text
Email: sarah.sample@company.com
Phone: 1-415-555-1212
Country: Sample Text
City: Sample Text

Date Of Birth: 8/23/2025
Age: 23,994
Owner: Sample Text

System Information (Header visible on edit only)
Created By: Sample Text

Last Modified By: Sample Text

Custom Links (Header visible on edit only)

Activity 2: To edit a Page Layout in BookingGuest Object

SETUP > OBJECT MANAGER
BookingGuest

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers

Save ▾ Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields Quick Find Field Name

Section	Booking	Created By	Relation with Cus...
Blank Space	BookingGuest Name	Gender	Special Needs
Age	City	Last Modified By	Visa Required
Age Category	Country	Passport Number	

Actions in this section are currently inherited from the global publisher layout. You can [override the global publisher layout](#) to set a customized list of actions for the publisher pages that use this layout.

BookingGuest Detail

Standard Buttons Edit Delete Clone Change Owner Change Record Type Printable View Edit Labels Custom Buttons

Information (Header visible on edit only)
Booking: Sample Text
BookingGuest Name: Sample Text
Age: 75,989
Gender: Sample Text
Relation with Customer: Sample Text
Age Category: Sample Text
Special Needs: Sample Text

Country: Sample Text
City: Sample Text
Passport Number: Sample Text
Visa Required: ✓

System Information (Header visible on edit only)
Created By: Sample Text

Last Modified By: Sample Text

Custom Links (Header visible on edit only)

Mobile Cards (Salesforce mobile only)

Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.

Related Lists

Activity 3: To edit a Page Layout in TravelPackage Object

SETUP > OBJECT MANAGER
TravelPackage

Page Layouts

TravelPackage Detail

Information (Header visible on edit only)

- Section: Sample Text
- Average Rating: Sample Text
- Country: Sample Text
- Maximum Group Size: Sample Text
- Package Type: Sample Text
- Region: Sample Text
- Guide Included: Sample Text
- Guide Languages: Sample Text
- Meals Included: Sample Text
- Places Covered: Sample Text
- Preferred Guide: Sample Text
- TravelPackage Name: Sample Text
- Additional Notes: Sample Text
- Created By: Sample Text
- Insurance Included: Sample Text
- Membership: Sample Text
- Preferred Guide: Sample Text
- TravelPackage Name: Sample Text
- Availability Status: Sample Text
- Duration in Days: Sample Text
- Last Modified By: Sample Text
- Owner: Sample Text
- Price Per Person: Sample Text
- Visa Assistance: Sample Text

System Information (Header visible on edit only)

- Created By: Sample Text
- Last Modified By: Sample Text

Mobile Cards (Salesforce mobile only)

Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.

Activity 4: To edit a Page Layout in Employee Object

SETUP > OBJECT MANAGER
Employee

Page Layouts

Employee Detail

Information (Header visible on edit only)

- Employee ID: GEN-2004-001234
- Employee Name: Sarah Sample
- Email: sarah.sample@company.com
- Phone: 1-415-555-1212
- Role: Sample Text
- Department: Sample Text
- Employment Type: Sample Text
- Salary: \$123.45
- Joining Date: 8/23/2025
- Availability Status: Sample Text
- Languages Spoken: Sample Text
- Assigned Region: Sample Text

Address Details

- Country: Sample Text
- City: Sample Text

System Information (Header visible on edit only)

- Created By: Sample Text
- Last Modified By: Sample Text

Custom Links (Header visible on edit only)

Activity 5: To edit a Page Layout in Booking Object

SETUP > OBJECT MANAGER
Booking

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers

Save ▾ Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields

	Field Name	
Section	Booking Date	Cancel Confirmation
Blank Space	Booking Name	Customer
Accommodation Amo...	Booking Number	Cancellation Date
Approval Status	Cancellation Reason	Customer Email
	Booking Status	Last Modified By
		Owner
		Require Visa Assis...
		Total Acc...
		Label, Requ...
		Visa Assistance
		Total Bill...
		Type: Checkbox
		Total Tra...
		This item is currently in use (click to locate)

Quick Find Field Name *

Information (Header visible on edit only)

Booking Number GEN-2004-001234

Booking Name Sample Text
 Customer Sample Text
 Customer Email sarah.sample@company.com
 Travel Package Sample Text
 Booking Date 8/23/2025
 Travelling Start Date 8/23/2025
 Travelling End Date 8/23/2025
 Trip Type Sample Text
 Membership Chosen Sample Text
 Preferred Accommodation Sample Text
 Include Travel Insurance ✓
 Require Visa Assistance ✓
 Require Tour Guide ✓
 Guide Assigned Sample Text
 Preferred Guide Language Sample Text
 No of Booking Guests 88,488
 Info Available

Number of Travellers 23,214
 Booking Status Sample Text
 Approval Status Sample Text
 Cancellation Date 8/23/2025
 Cancel Confirmation ✓
 Travel Cost Per Person \$123.45
 Total Travel Amount \$123.45
 Accommodation Amount per Person per Day \$123.45
 Cancellation Reason Sample Text
 Total Accommodation Amount \$123.45
 Total Billing Amount \$123.45
 Owner Sample Text

System Information: (Header visible on edit only)

Activity 6: To edit a Page Layout in Booking Payments Object

SETUP > OBJECT MANAGER
Booking Payment

Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Scoping Rules
Object Access
Triggers
Flow Triggers

Save ▾ Quick Save Preview As... Cancel Undo Redo Layout Properties

Fields

	Created By	Owner	Payment Number	Total Bill Amount
Section	Customer Name	Payment Date	Payment Receipt Sent	
Booking	Last Modified By	Payment Method	Payment Reference...	
Booking Payment Name	Notes	Payment Mode Details	Payment Status	

Quick Actions in the Salesforce Classic Publisher

Actions in this section are currently inherited from the global publisher layout. You can override the global publisher layout to set a customized list of actions for the publisher on pages that use this layout.

Salesforce Mobile and Lightning Experience Actions

Actions in this section are predefined by Salesforce. You can override the predefined actions to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions in the Quick Actions in the Salesforce Classic Publisher section, and have saved the layout, then this section inherits that set of actions by default when you click to override.

Booking Payment Detail

Standard Buttons Edit Delete Clone Change Owner Change Record Type Printable View Sharing Sharing Hierarchy Edit Labels Custom Buttons

Information (Header visible on edit only)

Payment Number GEN-2004-001234
 Customer Name Sample Text
 Booking Sample Text
 Payment Date 8/23/2025
 Total Bill Amount \$123.45
 Payment Status Sample Text
 Notes Sample Text

Payment Method Sample Text
 Payment Reference Number Sample Text
 Payment Mode Details Sample Text
 Payment Receipt Sent ✓
 Owner Sample Text

Activity 7: To edit a Page Layout in Feedbacks Object

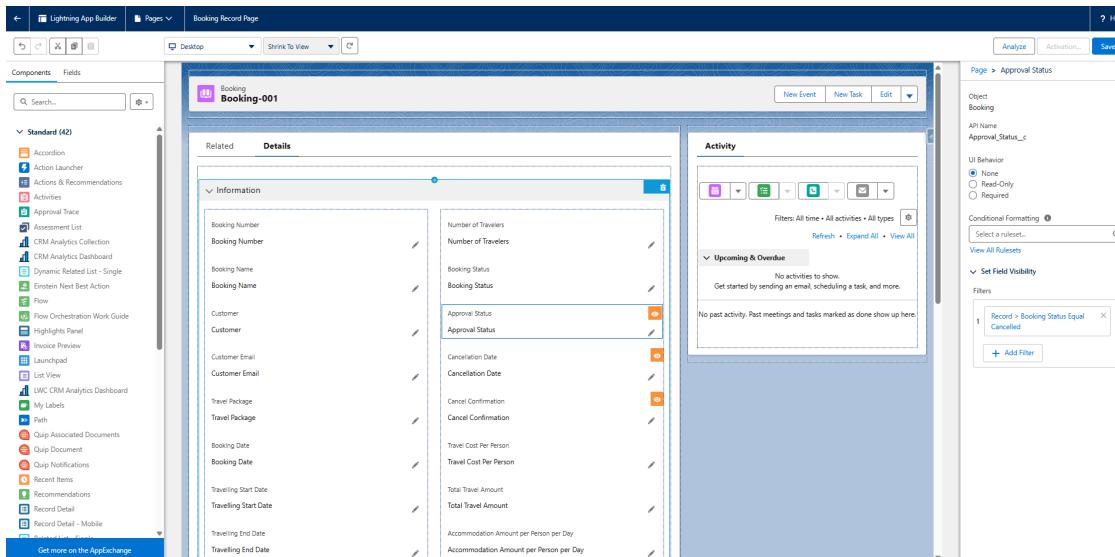
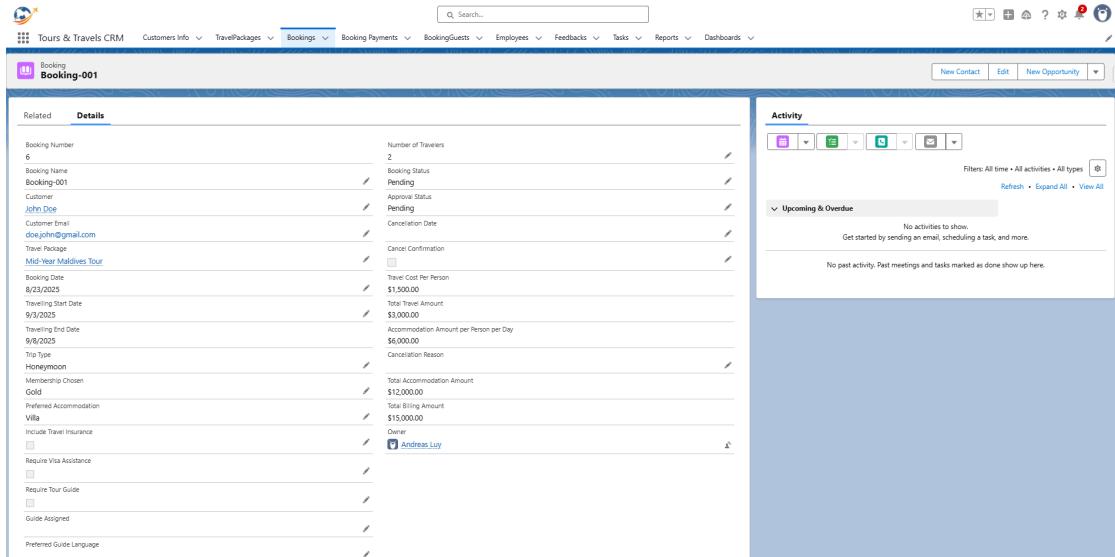
The screenshot shows the Salesforce Setup interface with the 'Object Manager' selected. Under the 'Feedback' object, the 'Page Layouts' tab is active. The main area displays the 'Feedback Detail' page layout. The layout includes sections for 'Information (Header visible on edit only)', 'System Information (Header visible on edit only)', and 'Custom Links (Header visible on edit only)'. Fields visible in the layout include Feedback No, Created By, Last Modified By, Response Status, Overall Rating, Tour Guide Rating, Transport Rating, Accommodation Rating, and Owner. A note at the bottom says 'Drag expanded lookups and mobile-enabled Visualforce pages here to display them as mobile cards.'

In Salesforce, Page layout can define how the records will appear to the Users by controlling the placement and visibility of fields, section, and other elements.

Upon editing the page layout across different objects like Customer Info, Bookings, and others helped me customize the CRM for a cleaner and Users friendly experience. This milestone surely helped make the application look more professional and intuitive.

Milestone 15: Dynamic Forms

Activity 1: To create a Dynamic Form in Booking Object



In Salesforce, Dynamic forms will let us configure record pages at the field level within the Lightning App Builder. Unlike the traditional page layouts, it can allow us the conditional visibility of fields and sections based on profile, record values, or logic. Implementing a dynamic form showed me how powerful it can be. It also helped me reduce the clutter on the record page and made the interface more intuitive.

Milestone 16: Users

Activity 1: Create User

The screenshot shows the 'User Detail' page for a user named Michael Jackson. The user's name is listed at the top. Below the name, there are sections for 'User Detail' and 'Role'. Under 'User Detail', fields include Name (Michael Jackson), Alias (mjack), Email (luy.andreas@student.auf.edu.ch), Username (luy.andreas.michael123@salesforce.com), Nickname (michjack), Title (Manager), Company (None), Department (None), Division (None), Address (None), Time Zone (GMT-07:00 Pacific Daylight Time (America/Los_Angeles)), Locale (English (United States)), Language (English), Delegated Approver (None), Manager (None), Receive Approval Request Emails (Only if I am an approver), Federation ID (None), App Registration: One-Time Password Authenticator (None), App Registration: Salesforce Authenticator (None), Security Key (2DF or WebAuthn) (None), and Lightning Login (None). A note says 'Temporary Verification Code (Expires in 1 to 24 Hours)' followed by a [Generate] button. At the bottom, there is a note about 'User ProfileHelp for this Page'. On the right side, there are tabs for 'Role' (Travel Agent Manager), 'User License Profile' (Salesforce Platform Travel Agent Profile), and other user settings like Active (checked), Marketing User (unchecked), Offline User (unchecked), Knowledge User (unchecked), Flow User (unchecked), Service Cloud User (unchecked), Site.com Contributor User (unchecked), Site.com Publisher User (unchecked), WDC User (unchecked), Mobile Push Registrations (View), Data.com User Type (None), Accessibility Mode (Classic Only) (unchecked), Debug Mode (unchecked), High-Contrast Palette on Charts (unchecked), Load Lightning Pages While Scrolling (checked), Salesforce CRM Content User (checked), Receive Salesforce CRM Content Email Alerts (checked), Receive Salesforce CRM Content Alerts as Daily Digest (checked), Make Setup My Default Landing Page (unchecked), Allow Forecasting (unchecked), No MRU Updates (unchecked), Call Center (unchecked), Phone (unchecked), Extension (unchecked), Fax (unchecked), Mobile (unchecked), Email Encoding (Unicode (UTF-8)), Employee Number (None), and Used Data Space (0 B (Used)).

Activity 2: creating other Users

The screenshot shows the 'User Detail' page for a user named Andreas Luy. The user's name is listed at the top. Below the name, there are sections for 'User Detail' and 'Role'. Under 'User Detail', fields include Name (Andreas Luy), Alias (anduy), Email (luy.andreas@student.auf.edu.ch), Username (luy.blue@salesforce.com), Nickname (andreasblue), Title (Manager), Company (None), Department (None), Division (None), Address (None), Time Zone (GMT-07:00 Pacific Daylight Time (America/Los_Angeles)), Locale (English (United States)), Language (English), Delegated Approver (None), Manager (None), Receive Approval Request Emails (Only if I am an approver), Federation ID (None), App Registration: One-Time Password Authenticator (None), App Registration: Salesforce Authenticator (None), Security Key (2DF or WebAuthn) (None), and Lightning Login (None). A note says 'Temporary Verification Code (Expires in 1 to 24 Hours)' followed by a [Generate] button. At the bottom, there is a note about 'User ProfileHelp for this Page'. On the right side, there are tabs for 'Role' (Travel Agent), 'User License Profile' (Salesforce Platform Travel Agent Profile), and other user settings like Active (checked), Marketing User (unchecked), Offline User (unchecked), Knowledge User (unchecked), Flow User (unchecked), Service Cloud User (unchecked), Site.com Contributor User (unchecked), Site.com Publisher User (unchecked), WDC User (unchecked), Mobile Push Registrations (View), Data.com User Type (None), Accessibility Mode (Classic Only) (unchecked), Debug Mode (unchecked), High-Contrast Palette on Charts (unchecked), Load Lightning Pages While Scrolling (checked), Salesforce CRM Content User (checked), Receive Salesforce CRM Content Email Alerts (checked), Receive Salesforce CRM Content Alerts as Daily Digest (checked), Make Setup My Default Landing Page (unchecked), Allow Forecasting (unchecked), No MRU Updates (unchecked), Call Center (unchecked), Phone (unchecked), Extension (unchecked), Fax (unchecked), Mobile (unchecked), Email Encoding (Unicode (UTF-8)), Employee Number (None), and Used Data Space (0 B (Used)).

User Detail

Andres Bonifazio

Name: Andres Bonifazio
Alias: andbon
Email: luvandreas@student.auf.edu.ch [Read Only]
Username: boni.andreseyellow@salesforce.com
Nickname: andebonifazio [i]
Title:
Company:
Department:
Division:
Address:
Time Zone: (GMT-07:00) Pacific Daylight Time (America/Los_Angeles)
Locale: English (United States)
Language: English
Delegated Approver:
Manager:
Receive Approval Request Emails: Only if I am an approver
Federation ID:
App Registration: One-Time Password Authenticator [i]
App Registration: Salesforce Authenticator [i]
Security Key (2FA or WebAuthn) [i]
Lightning Login [i]
Temporary Verification Code (Expires in 1 to 24 Hours): [General] [i]

Role: Travel Agent
User License: Salesforce Platform
Profile: Travel Agent Profile
Active: ✓
Marketing User:
Offline User:
Knowledge User:
Flow User:
Service Cloud User:
Site.com Contributor User:
Site.com Publisher User:
WDC User:
Mobile Push Registrations: View
Data.com User Type:
Accessibility Mode (Classic Only): [i]
Debug Mode: [i]
High-Contrast Palette on Charts: [i]
Load Lightning Pages While Scrolling: ✓ [i]
Salesforce CRM Content User: ✓ [i]
Receive Salesforce CRM Content Email Alerts: ✓ [i]
Receive Salesforce CRM Content Alerts as Daily Digest: ✓ [i]
Make Setup My Default Landing Page:
Allow Forecasting:
No MRU Updates: [i]
Call Center:
Phone:
Fax:
Mobile:
Email Encoding: Unicode (UTF-8)
Employee Number:
Used Data Space: 0 B [View]

All Users

All Users

On this page you can create, view, and manage users.

To get more licenses, use the Your Account app. [Let's Go](#)

View: [All Users](#) [Edit](#) | [Create New View](#)

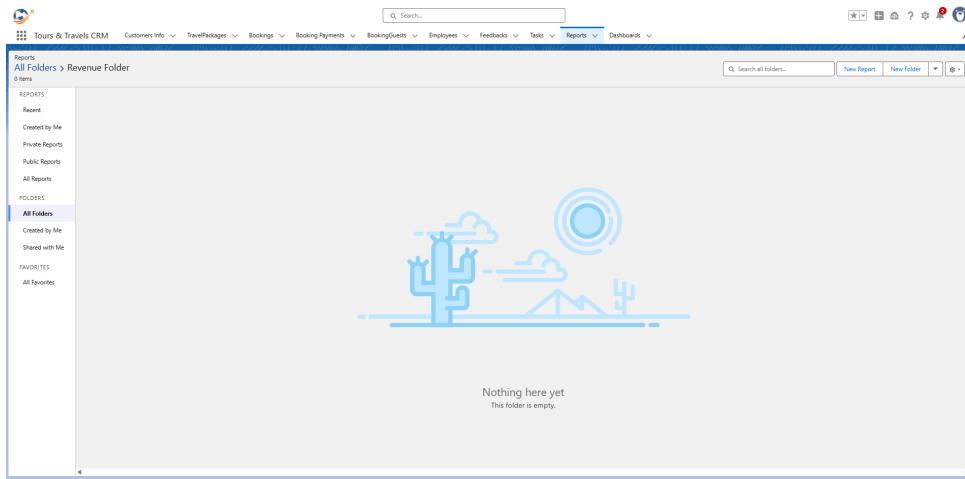
Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Bonifazio, Andres	andbon	boni.andreseyellow@salesforce.com	Travel Agent	✓	Travel Agent Profile
<input type="checkbox"/> Edit	Chatter_Expert	Chatter	chatty.00da000000000000@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit	EPIC_OrgFarm	EPIC	epic.0c29508a80@orgfarm.salesforce.com		✓	System Administrator
<input type="checkbox"/> Edit	Jackson, Michael	black	luv.andreas.michael123@salesforce.com	Travel Agent Manager	✓	Travel Agent Profile
<input type="checkbox"/> Edit	Luv Andreas	lex	luv.andreas705@salesforce.com		✓	System Administrator
<input type="checkbox"/> Edit	Luv Andreas	andluv	luv.luv@salesforce.com	Travel Agent	✓	Travel Agent Profile
<input type="checkbox"/> Edit	User_Integration	integr8	integration.00d0000000000000@prod.dream.com		✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	integr8secuity@00d0000000000000@prod.dream.com		✓	Analytics Cloud Security User

In Salesforce, Users represent the people who can log in and interact with the system. Each Users will be assigned a role, profile, and a license, which will together define their privilege or level of access and permissions.

Creating the Users such as travel agents and managers gave the project a more realistic feel. This milestone highlighted how Salesforce can balance flexibility with security, ensuring that the data is available to those who need it.

Milestone 17: Reports

Activity 1: Create a Report folder



Activity 2: Sharing a Report folder

Share folder

These sharing settings apply to all subfolders in this folder.

Share With:

Names: Travel Agent Manager, Finance Officer Role | Access: View

Share

Who Can Access:

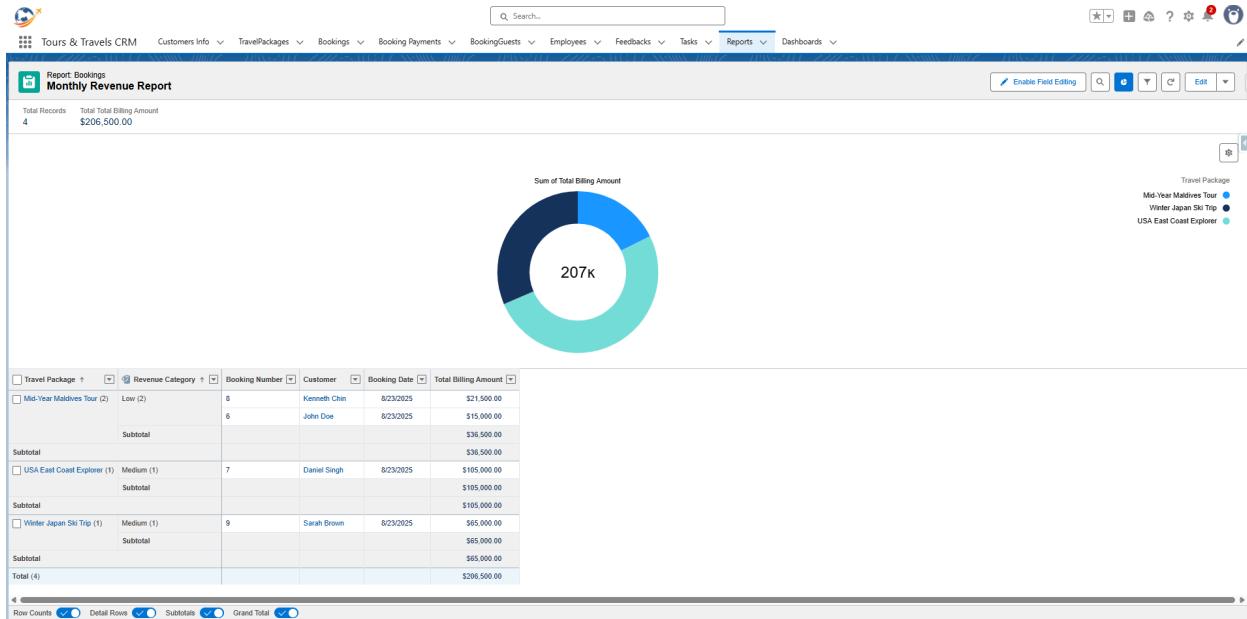
Andreas Luy (Users) | Manage

Done

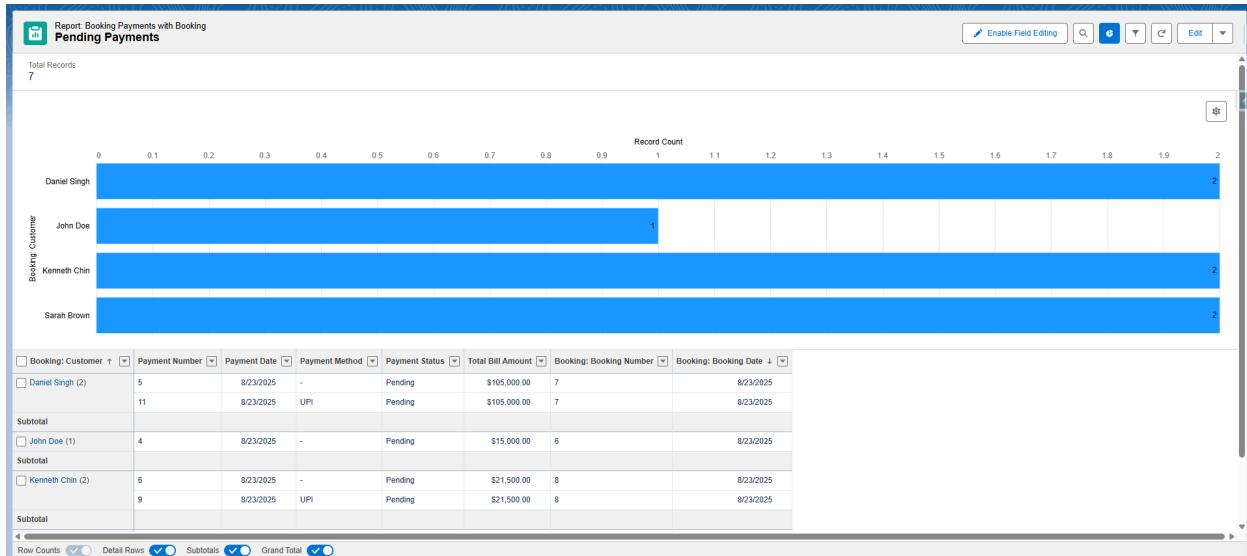
Who Can Access:

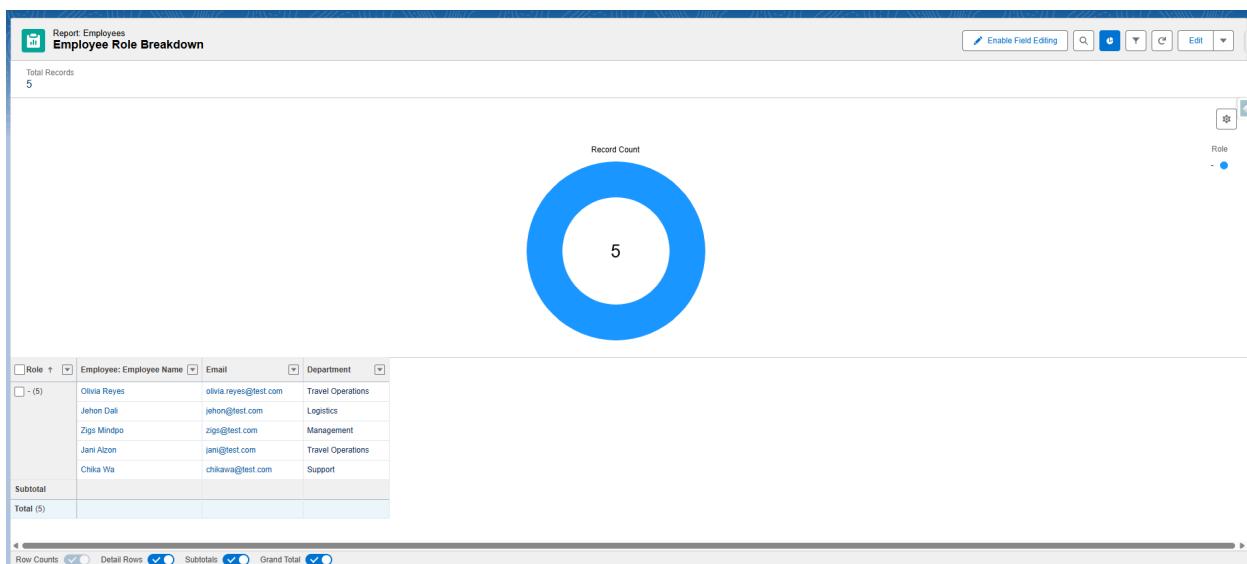
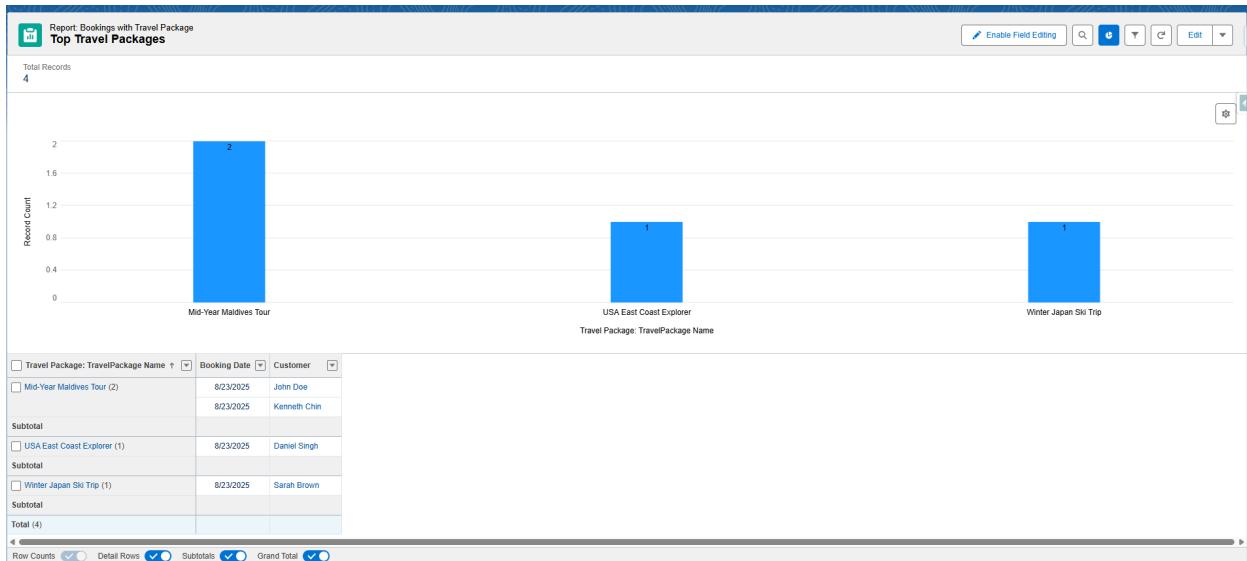
Andreas Luy (Users), Finance Officer Role (Roles), Travel Agent Manager (Roles)

Activity 3: Create Report



Activity 4: Create Other Reports





Reports

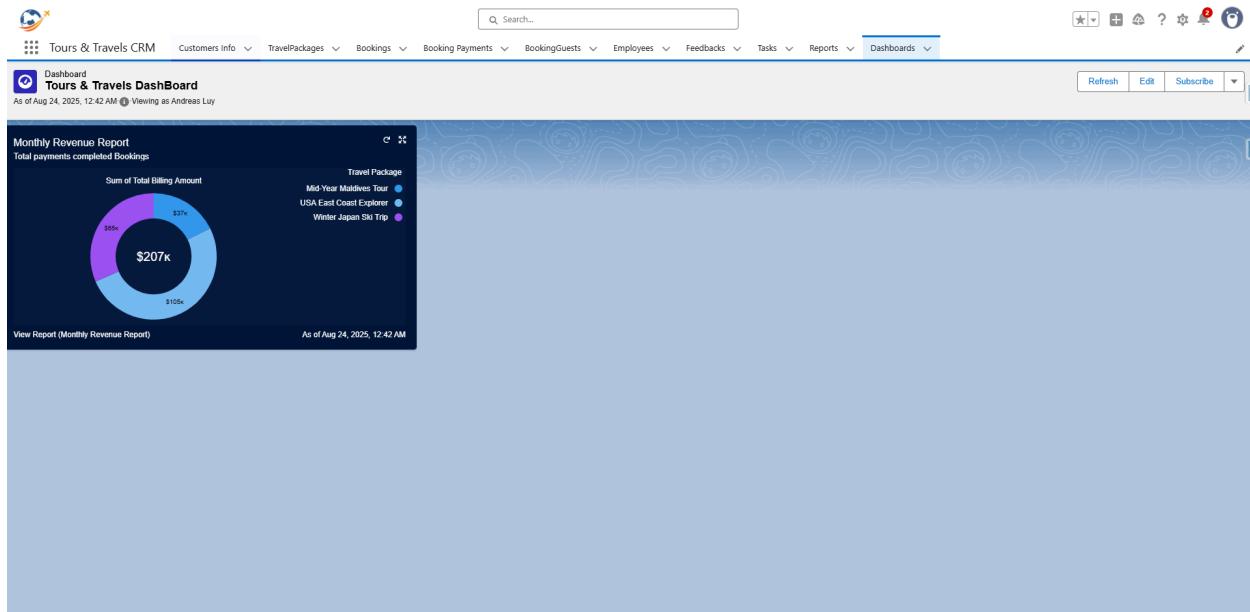
Reports						
Recent		9 items				
REPORTS	Report Name	Description	Folder	Created By	Created On	Subscribed
Recent	Customer Demographics by Country	Customer Reports	Andreas Luy	8/24/2025, 12:30 AM	<input checked="" type="checkbox"/>	
Created by Me	Bookings by Country	Customer Reports	Andreas Luy	8/24/2025, 12:26 AM	<input checked="" type="checkbox"/>	
Private Reports	Customer Booking Summary	Customer Reports	Andreas Luy	8/24/2025, 12:20 AM	<input checked="" type="checkbox"/>	
Public Reports	Revenue by Travel Package	Packages Folder	Andreas Luy	8/24/2025, 12:23 AM	<input checked="" type="checkbox"/>	
All Reports	Top Travel Packages	Packages Folder	Andreas Luy	8/24/2025, 12:05 AM	<input checked="" type="checkbox"/>	
Created by Me	Pending Payments	Revenue Folder	Andreas Luy	8/23/2025, 11:59 PM	<input checked="" type="checkbox"/>	
FOLDERS	Employee Role Breakdown	Employee Folder	Andreas Luy	8/24/2025, 12:11 AM	<input checked="" type="checkbox"/>	
All Folders	Monthly Revenue Report	Revenue Folder	Andreas Luy	8/23/2025, 11:21 PM	<input checked="" type="checkbox"/>	
Created by Me						
Shared with Me						
FAVORITES						
All Favorites						

During this Milestone, I created several reports across customers, bookings, payments, employees, and travel packages. Each of the reports are organized into their proper folders. At first, I missed the note about creating records first before creating the reports, which caused delays and blank results until I added some dummy date. I also ran into a few errors when saving a record (in some objects), but I was able to fix them later after rechecking the field requirements.

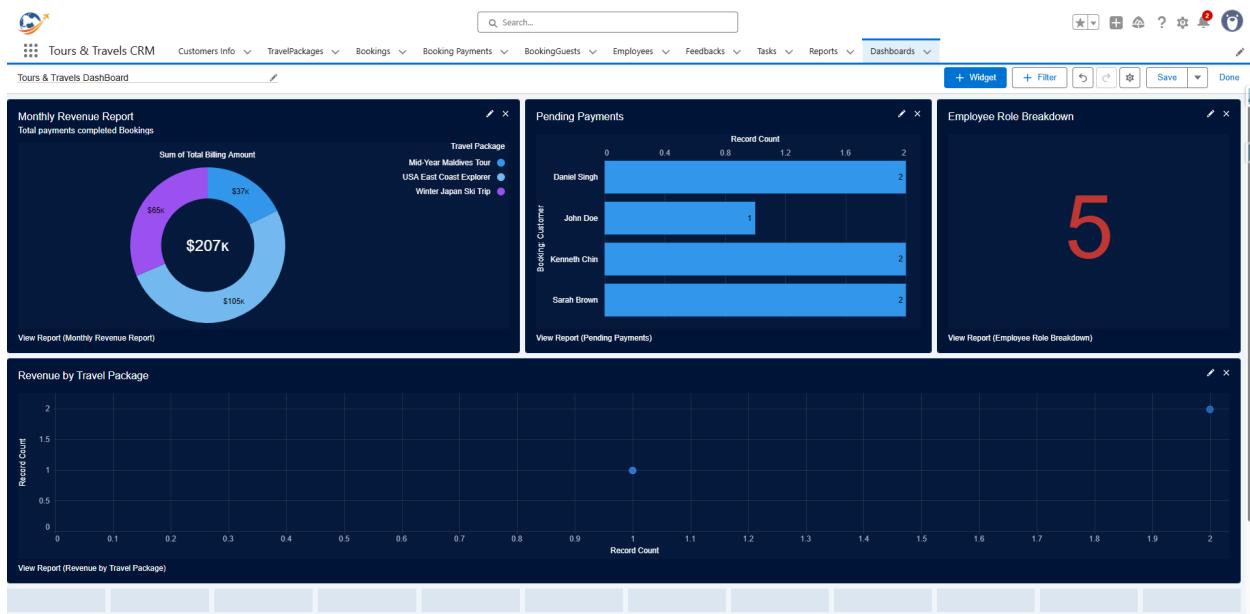
Despite these small challenges, the process gave me hands-on experience with grouping, summarizing, and adding charts in Salesforce's Reports. Overall, this milestone really helped me become familiar with the Salesforce UI and even understand how reports can turn raw data into a useful insight. Now for the next challenge, I think I'll display the reports into the dashboard? Maybe...

Milestone 18: Dashboards

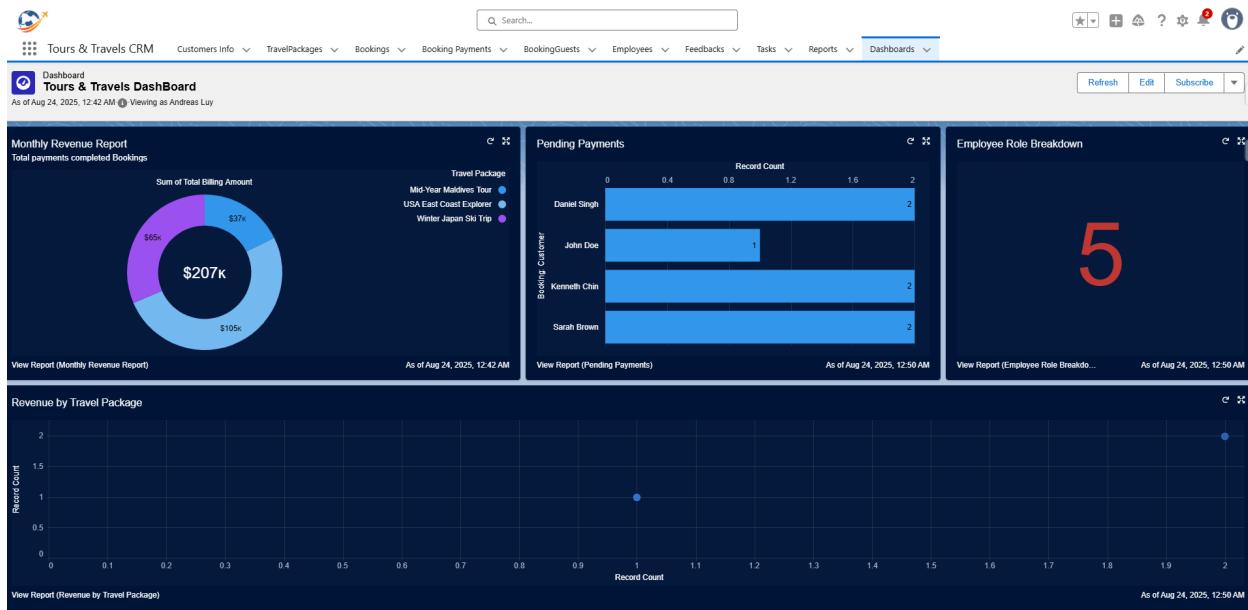
Activity 1: - Create Dashboard



Activity 2: Add other Components in the Dashboard

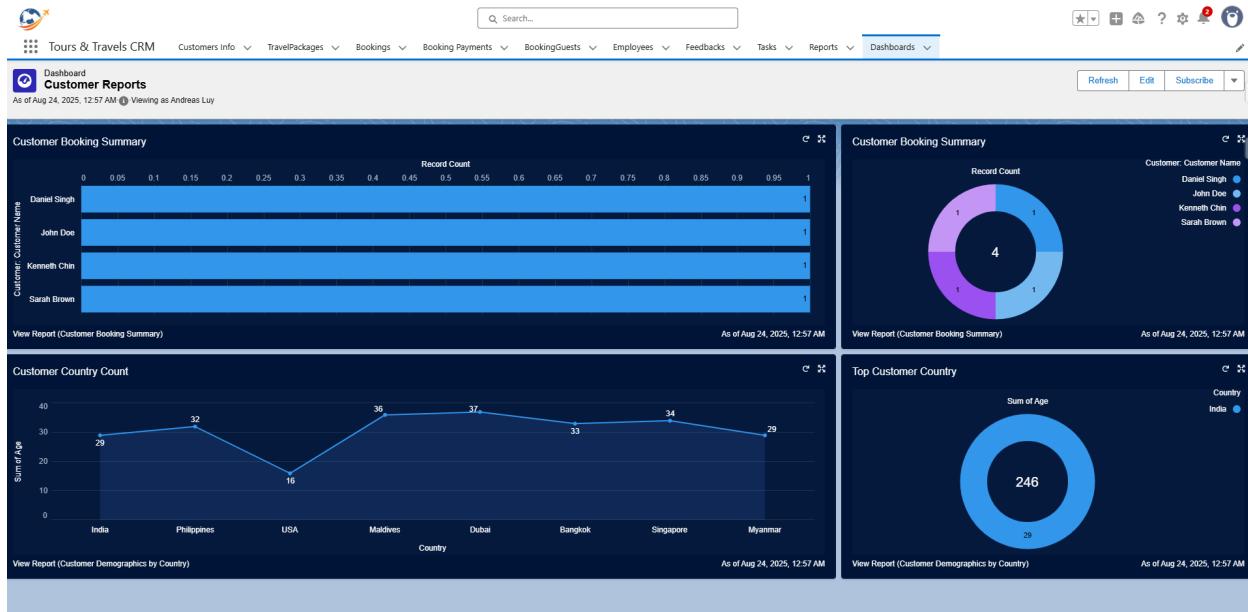


Activity 3: View Dashboard

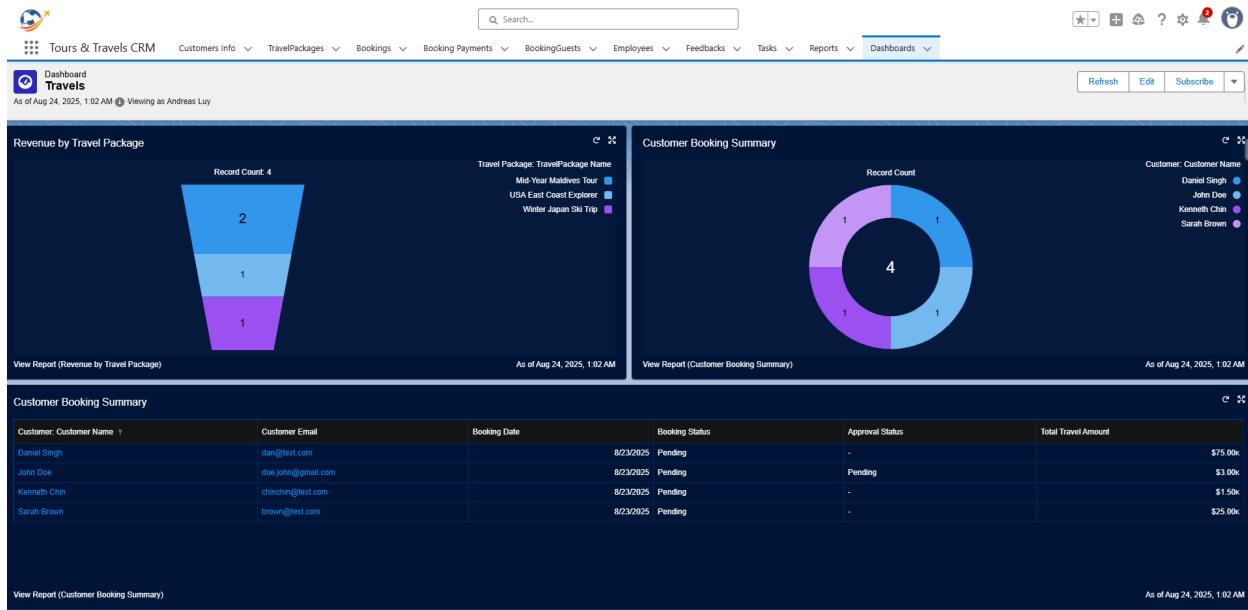


Activity 4: Create another 2 Dashboards

Customer Reports Dashboard



Travels Dashboard



For Milestone 18, I initially struggled with what to put on the dashboard, so I started by asking myself something like "What would a manager need at a glance?" That led me to add widgets for Pending payments, Monthly revenue, Top travel packages, customer booking summary, and an employee role breakdown.

I created the underlying reports (with custom report types where needed) and summarized the data so that it's easier to read. So that when I finally add them to the dashboard, it would be easier and would look organized.

Overall, the dashboard I created gives a clear at-a-glance of what needs to be seen and will keep me up-to-date.

Milestone 19: Lightning Web Component (LWC) Creation

Activity 1: Create Lightning Web Component Creation

The screenshot shows the Lightning Studio interface. On the left, there is a code editor window titled "TravelPackageController.apxc" containing the following Apex code:

```
1 * public with sharing class TravelPackageController {
2
3     @AuraEnabled(cacheable=true)
4
5     *     public static List<TravelPackage__c> getPackagesByCountry(String country) {
6
7         return [SELECT Id, Name, Package_Type__c, Duration_in_Days__c , Guide_Included__c, Membership__c, Region__c, Transportation_Modes__c, Availability_Status__c, Average_Rating__c, Places_Covered__c FROM TravelPackage__c
8 WHERE Country__c = :country];
9
10    }
11
12}
13 }
```

Below the code editor are sections for "Component Name" (set to "TravelPackageSelector"), "Additional Configurations" (Master Label set to "Master Label" and API Version set to "63.0"), and "Target Configuration" (checkboxes checked for "lightning_AppPage", "lightning_HomePage", and "lightning_SitePage").

On the right, the "LIGHTNING WEB COMPONENT" section shows the newly created component "travelPackageSelector". It lists four files: "travelPackageSelector.html" (HTML), "travelPackageSelector.js" (JavaScript), "travelPackageSelector.css" (CSS), and "travelPackageSelector.js-m..." (Metadata). The "travelPackageSelector.html" file is highlighted.

Inserting the source codes:

travelPackageSelector.html

The screenshot shows the Lightning Studio interface with the travelPackageSelector.html component selected. The code editor displays the following HTML template:

```

<template>
  <lightning-card title="Travel Packages">
    <div class="slds-p-around_medium">
      <lightning-combocheckbox name="country"
        label="Select Country"
        value={selectedCountry}
        placeholder="Choose a country"
        options={countryOptions}
        onchange={handleCountryChange}>
    </lightning-combocheckbox>

    <template if:true={packages}>
      <template for:each=[packages] for:item="pkg">
        <div key={pkg.Id} class="slds-box slds-m-top_small">
          <p><b>Package Type Name:</b> {pkg.Name}</p>
          <p><b>Package Type:</b> {pkg.Package_Type__c}</p>
          <p><b>Duration in Days:</b> {pkg.Duration_in_Days__c}</p>
          <p><b>Guide Facility:</b> {pkg.Guide_Included__c}</p>
          <p><b>Places Covered:</b> {pkg.Places_Covered__c}</p>
          <p><b>Membership:</b> {pkg.Membership__c}</p>
          <p><b>Region:</b> {pkg.Region__c}</p>
          <p><b>Transportation mode:</b> {pkg.Transportation_Modes__c}</p>
        </div>
      </template>
    </template>
  </lightning-card>
</template>

```

travelPackageSelector.js

The screenshot shows the Lightning Studio interface with the travelPackageSelector.js component selected. The code editor displays the following JavaScript code:

```

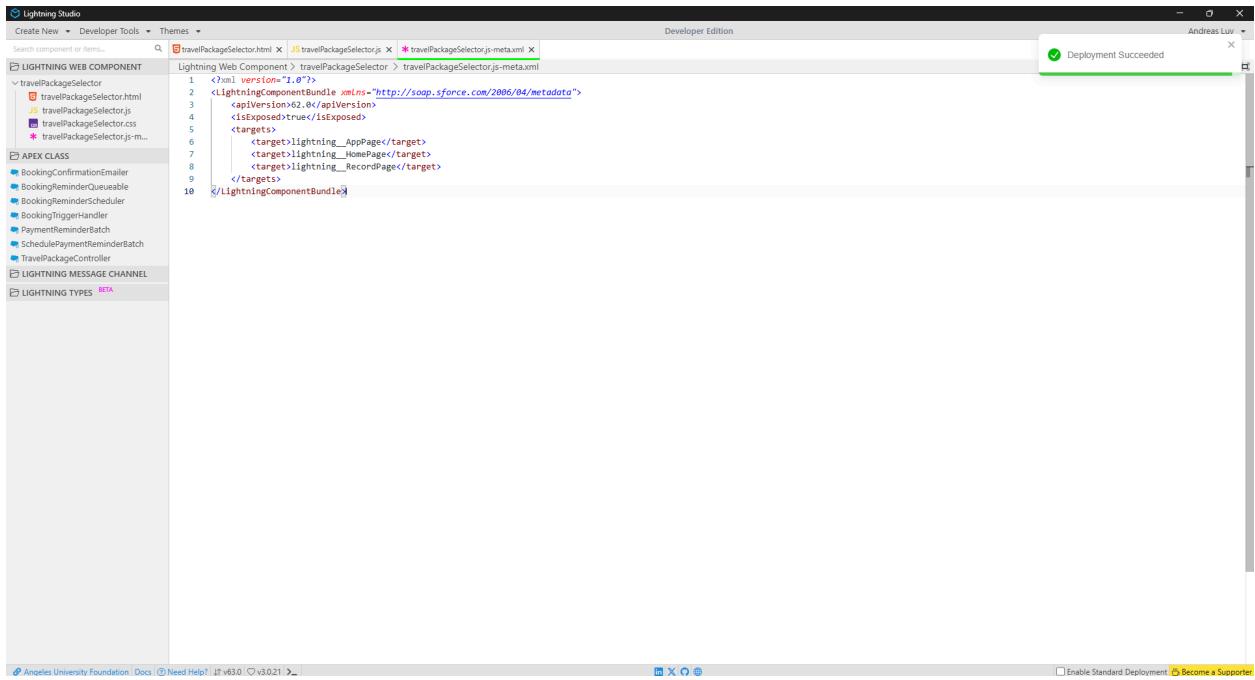
export default class TravelPackageSelector extends LightningElement {
  get countryOptions() {
    return [
      { label: 'Singapore', value: 'Singapore' },
      { label: 'Myanmar', value: 'Myanmar' },
      { label: 'Nepal', value: 'Nepal' },
    ];
  }

  handleCountryChange(event) {
    this.selectedCountry = event.detail.value;
    this.packages = [];
    this.error = undefined;

    getPackagesByCountry({ country: this.selectedCountry })
      .then(result => {
        this.packages = result;
      })
      .catch(error => {
        this.error = error.body.message;
      });
  }
}

```

travelPackageSelector.js-meta.xml



The screenshot shows the Lightning Studio interface. In the top right corner, there is a green success message 'Deployment Succeeded'. The main area displays a code editor with the file 'travelPackageSelector.js-meta.xml'. The code is as follows:

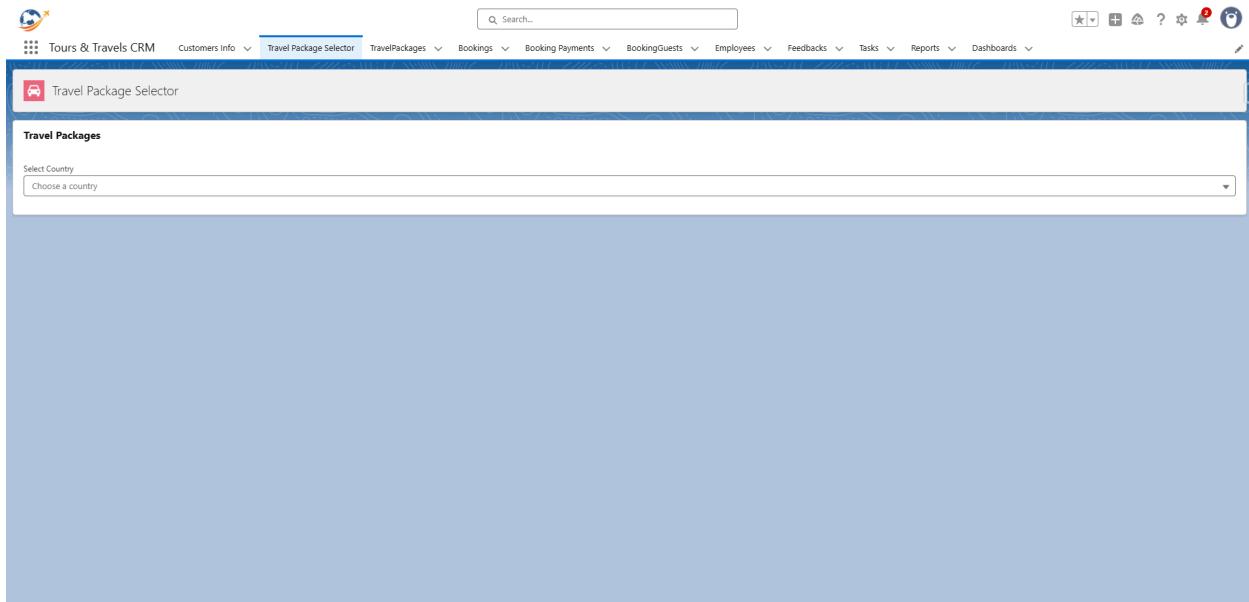
```
<!xm&lt;version="1.0">
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
<apiVersion>42.0</apiVersion>
<isExposed>true</isExposed>
<targets>
<target>lightning__AppPage</target>
<target>lightning__HomePage</target>
<target>lightning__RecordPage</target>
</targets>
</LightningComponentBundle>
```

During this Milestone, I built a lightning web component with an Apex Controller in order to display the travel packages dynamically based on the selected country. At first, I found the setup a bit challenging since I haven't completely read the requirements. But after a while, it made me feel comfortable enough to explore by myself. I found it easier to navigate between files by using the extension provided.

This activity helped me understand the flow process much better especially when working with LWCs and even gave me hands-on experience. Overall, it feels like a rewarding milestone especially that I learned a new extension that can help me whenever I need to insert some source codes.

Milestone 20: Lightning App Page Creation

Activity 1: Adding the Lightning Component to the App page



For this Milestone, I was able to set up a Lightning App Page to showcase the Travel Package Selector as a single focused workspace. I created a new page called Travel Package Selector inside the Tours & Travels CRM.

The guide helped me by helping me click the right button and then when I launched it, the component loaded smoothly.

I also visited the newly created tab directly on the Tours & Travels, and it's a good thing that the Select a country option is displaying without any issues.

Overall, this was a simple change with a big Users experience impact

Phase 4: Data Migration, Testing & Security

Milestone 21: Field History Tracking

Activity 1

Booking

Enable Booking History

[Deselect all fields](#)

Track old and new values

- Owner
- Booking Name
- Customer
- Guide Assigned
- Travelling Start Date
- Travel Package
- Customer Email
- Preferred Accommodation
- Preferred Guide Language
- Booking Status
- Membership Chosen
- Trip Type
- Number of Travelers
- Approval Status
- Include Travel Insurance
- Require Visa Assistance
- Require Tour Guide
- Booking Date
- Cancellation Date
- Cancel Confirmation
- Cancellation Reason

[Cancel](#)

[Save](#)

TravelPackage

Enable TravelPackage History

[Deselect all fields](#)

Track old and new values

- Owner
- TravelPackage Name
- Duration in Days
- Price Per Person
- Meals Included
- Availability Status
- Country
- Region
- Places Covered
- Visa Assistance
- Insurance Included
- Guide Included
- Preferred Guide Language
- Maximum Group Size
- Additional Notes
- Average Rating

Track changes only

- Package Type
- Transportation Modes
- Membership
- Guide LanguagesPlaces Covered

[Cancel](#) [Save](#)

For this milestone, I implemented Field History Tracking. From setup, I was able to find the Field History Tracking and was able to enable the required fields to be tracked for the objects: Booking and Travel Package. This activity helped me understand how I can include an audit trail for my app- Tours & Travels CRM.

I also learned that tracking begins after it's enabled and supports up to 20 fields per object; however, in a business context we should track only the fields that matter to avoid redundancy.

Milestone 22: Duplicate and Matching Rules

Activity 1: Create a Custom Matching Rule

The screenshot shows the 'Matching Rules' page in the Salesforce setup. A new rule named 'Unique Email and Phone Number Combination' has been created for the 'Customer Info' object. The matching criteria are defined as '(Customer Info: Email EXACT MatchBlank = TRUE) AND (Customer Info: Phone EXACT MatchBlank = TRUE)'. The rule is active and was created by Andreas Luy on 8/24/2025 at 2:09 AM.

Activity 2: Create a Duplicate Rule

The screenshot shows the 'Duplicate Rules' page in the Salesforce setup. A new rule named 'unique Email and Phone' has been created for the 'Customer Info' object. The matching rule is set to 'Unique Email and Phone Number Combination' and is mapped. The matching criteria are '(Customer Info: Email EXACT MatchBlank = TRUE) AND (Customer Info: Phone EXACT MatchBlank = TRUE)'. The rule is active and was created by Andreas Luy on 8/24/2025 at 2:12 AM.

Creating the rules for the Customer Info using email and phone combinations gave me a practical way to ensure the data integrity. It's very important to note how easy it is for the duplicated data to creep in. But with these rules in place, the app can flag it before they became a problem.

Milestone 23: Profiles

Activity 1: Travel Agent Profile Creation

The screenshot shows the 'Travel Agent Profile' setup page. At the top, there's a header with a user icon, 'SETUP', and 'Profiles'. Below the header, the profile name 'Travel Agent Profile' is displayed. A note states: 'Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.' It also mentions: 'If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.' A horizontal bar at the top contains various access links like 'Login IP Ranges', 'Enabled Apex Class Access', etc.

Profile Detail

Name: Travel Agent Profile	Custom Profile: ✓
User License: Salesforce Platform	Created By: Andreas.Luy 8/23/2025, 6:37 PM
Modified By: Andreas.Luy 8/23/2025, 6:39 PM	

Page Layouts

Standard Object Layouts	Alternative Payment Method Layout	Object Milestone Layout
Global [View Assignment]	[View Assignment]	[View Assignment]
Email Application Not Assigned [View Assignment]	[View Assignment]	[View Assignment]
Home Page Layout Home Page Default [View Assignment]	[View Assignment]	[View Assignment]
Account Account Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Alternative Payment Method Layout [View Assignment]	[View Assignment]	[View Assignment]
Appointment Invitation Appointment Invitation Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Asset Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Relationship Asset Relationship Layout [View Assignment]	[View Assignment]	[View Assignment]
Assigned Resource Assigned Resource Layout [View Assignment]	[View Assignment]	[View Assignment]
Associated Location Associated Location Layout [View Assignment]	[View Assignment]	[View Assignment]
Asynchronous Operation Log Asynchronous Operation Log Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Authorization Form Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Consent Authorization Form Consent Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Data Use Authorization Form Data Use Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Text Authorization Form Text Layout [View Assignment]	[View Assignment]	[View Assignment]
Lead Lead Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Location Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Group Location Group Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Group Assignment Location Group Assignment Layout [View Assignment]	[View Assignment]	[View Assignment]
Object Milestone Object Milestone Layout [View Assignment]	[View Assignment]	[View Assignment]
Operating Hours Operating Hours Layout [View Assignment]	[View Assignment]	[View Assignment]
Order Order Layout [View Assignment]	[View Assignment]	[View Assignment]
Order Product Order Product Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Payment Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Authorization Payment Authorization Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Authorization Adjustment Payment Authorization Adjustment Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Gateway Payment Gateway Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Gateway Log Payment Gateway Log Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Group Payment Group Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Line Invoice Payment Line Invoice Layout [View Assignment]	[View Assignment]	[View Assignment]

Activity 2: Tour Guide Profile Creation

The screenshot shows the 'Tour Guide' setup page. The layout is identical to the Travel Agent profile page, featuring a header with 'SETUP', 'Profiles', and the profile name 'Tour Guide'. A note about permissions and record types is present. A horizontal bar at the top contains various access links like 'Login IP Ranges', 'Enabled Apex Class Access', etc.

Profile Detail

Name: Tour Guide	Custom Profile: ✓
User License: Salesforce Platform	Created By: Andreas.Luy 8/23/2025, 6:41 PM
Modified By: Andreas.Luy 8/23/2025, 6:42 PM	

Page Layouts

Standard Object Layouts	Alternative Payment Method Layout	Object Milestone Layout
Global [View Assignment]	[View Assignment]	[View Assignment]
Email Application Not Assigned [View Assignment]	[View Assignment]	[View Assignment]
Home Page Layout Home Page Default [View Assignment]	[View Assignment]	[View Assignment]
Account Account Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Alternative Payment Method Layout [View Assignment]	[View Assignment]	[View Assignment]
Appointment Invitation Appointment Invitation Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Asset Layout [View Assignment]	[View Assignment]	[View Assignment]
Asset Relationship Asset Relationship Layout [View Assignment]	[View Assignment]	[View Assignment]
Assigned Resource Assigned Resource Layout [View Assignment]	[View Assignment]	[View Assignment]
Associated Location Associated Location Layout [View Assignment]	[View Assignment]	[View Assignment]
Asynchronous Operation Log Asynchronous Operation Log Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Authorization Form Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Consent Authorization Form Consent Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Data Use Authorization Form Data Use Layout [View Assignment]	[View Assignment]	[View Assignment]
Authorization Form Text Authorization Form Text Layout [View Assignment]	[View Assignment]	[View Assignment]
Business Brand Business Brand Layout [View Assignment]	[View Assignment]	[View Assignment]
Lead Lead Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Location Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Group Location Group Layout [View Assignment]	[View Assignment]	[View Assignment]
Location Group Assignment Location Group Assignment Layout [View Assignment]	[View Assignment]	[View Assignment]
Object Milestone Object Milestone Layout [View Assignment]	[View Assignment]	[View Assignment]
Operating Hours Operating Hours Layout [View Assignment]	[View Assignment]	[View Assignment]
Order Order Layout [View Assignment]	[View Assignment]	[View Assignment]
Order Product Order Product Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Payment Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Authorization Payment Authorization Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Authorization Adjustment Payment Authorization Adjustment Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Gateway Payment Gateway Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Gateway Log Payment Gateway Log Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Group Payment Group Layout [View Assignment]	[View Assignment]	[View Assignment]
Payment Line Invoice Payment Line Invoice Layout [View Assignment]	[View Assignment]	[View Assignment]
Price Book Price Book Layout [View Assignment]	[View Assignment]	[View Assignment]

Activity 3: Finance Officer Profile Creation

Profile:
Finance Officer
[Back to List: Profiles](#)

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.

[Login Page Access](#) | [Enabled Apex Class Access](#) | [Enabled Visualforce Page Access](#) | [Enabled External Data Source Access](#) | [Enabled Named Credential Access](#) | [Enabled External Credential Principal Access](#) | [Enabled Custom Metadata Type Access](#) | [Enabled Custom Setting Definitions Access](#) | [Enabled Flow Access](#) | [Enabled Service Presence Status Access](#) | [Enabled Custom Permissions](#)

Profile Detail		Edit Clone Delete View Users	
Name	Finance Officer	Custom Profile	<input checked="" type="checkbox"/>
User License	Salesforce Platform	Modified By	Andreas Luy 8/23/2025, 6:45 PM
Description			
Created By	Andreas Luy 8/23/2025, 6:45 PM		
Page Layouts			
Standard Object Layouts			
Global	Global Layout [View Assignment]	Lead	Lead Layout [View Assignment]
Email Application	Not Assigned [View Assignment]	Location	Location Layout [View Assignment]
Home Page Layout	Home Page Default [View Assignment]	Location Group	Location Group Layout [View Assignment]
Account	Account Layout [View Assignment]	Location Group Assignment	Location Group Assignment Layout [View Assignment]
Alternative Payment Method	Alternative Payment Method Layout [View Assignment]	Object Milestone	Object Milestone Layout [View Assignment]
Appointment Invitation	Appointment Invitation Layout [View Assignment]	Operating Hours	Operating Hours Layout [View Assignment]
Asset	Asset Layout [View Assignment]	Order	Order Layout [View Assignment]
Asset Relationship	Asset Relationship Layout [View Assignment]	Order Product	Order Product Layout [View Assignment]
Assigned Resource	Assigned Resource Layout [View Assignment]	Payment	Payment Layout [View Assignment]
Associated Location	Associated Location Layout [View Assignment]	Payment Authorization	Payment Authorization Layout [View Assignment]
Async Operation Log	Async Operation Log Layout [View Assignment]	Payment Authorization Adjustment	Payment Authorization Adjustment Layout [View Assignment]
Authorization Form	Authorization Form Layout [View Assignment]	Payment Gateway	Payment Gateway Layout [View Assignment]
Authorization Form Consent	Authorization Form Consent Layout [View Assignment]	Payment Gateway Log	Payment Gateway Log Layout [View Assignment]
Authorization Form Data Use	Authorization Form Data Use Layout [View Assignment]	Payment Group	Payment Group Layout [View Assignment]
Authorization Form Text	Authorization Form Text Layout [View Assignment]	Payment Line Invoice	Payment Line Invoice Layout [View Assignment]
Business Brand	Business Brand Layout [View Assignment]	Price Book	Price Book Layout [View Assignment]

Activity 4: Marketing Executive Profile Creation

Profile:
Marketing Executive
[Back to List: Profiles](#)

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.

[Login Page Access](#) | [Enabled Apex Class Access](#) | [Enabled Visualforce Page Access](#) | [Enabled External Data Source Access](#) | [Enabled Named Credential Access](#) | [Enabled External Credential Principal Access](#) | [Enabled Custom Metadata Type Access](#) | [Enabled Custom Setting Definitions Access](#) | [Enabled Flow Access](#) | [Enabled Service Presence Status Access](#) | [Enabled Custom Permissions](#)

Profile Detail		Edit Clone Delete View Users	
Name	Marketing Executive	Custom Profile	<input checked="" type="checkbox"/>
User License	Salesforce Platform	Modified By	Andreas Luy 8/23/2025, 6:50 PM
Description			
Created By	Andreas Luy 8/23/2025, 6:50 PM		
Page Layouts			
Standard Object Layouts			
Global	Global Layout [View Assignment]	Lead	Lead Layout [View Assignment]
Email Application	Not Assigned [View Assignment]	Location	Location Layout [View Assignment]
Home Page Layout	Home Page Default [View Assignment]	Location Group	Location Group Layout [View Assignment]
Account	Account Layout [View Assignment]	Location Group Assignment	Location Group Assignment Layout [View Assignment]
Alternative Payment Method	Alternative Payment Method Layout [View Assignment]	Object Milestone	Object Milestone Layout [View Assignment]
Appointment Invitation	Appointment Invitation Layout [View Assignment]	Operating Hours	Operating Hours Layout [View Assignment]
Asset	Asset Layout [View Assignment]	Order	Order Layout [View Assignment]
Asset Relationship	Asset Relationship Layout [View Assignment]	Order Product	Order Product Layout [View Assignment]
Assigned Resource	Assigned Resource Layout [View Assignment]	Payment	Payment Layout [View Assignment]
Associated Location	Associated Location Layout [View Assignment]	Payment Authorization	Payment Authorization Layout [View Assignment]
Async Operation Log	Async Operation Log Layout [View Assignment]	Payment Authorization Adjustment	Payment Authorization Adjustment Layout [View Assignment]
Authorization Form	Authorization Form Layout [View Assignment]	Payment Gateway	Payment Gateway Layout [View Assignment]
Authorization Form Consent	Authorization Form Consent Layout [View Assignment]	Payment Gateway Log	Payment Gateway Log Layout [View Assignment]
Authorization Form Data Use	Authorization Form Data Use Layout [View Assignment]	Payment Group	Payment Group Layout [View Assignment]
Authorization Form Text	Authorization Form Text Layout [View Assignment]	Payment Line Invoice	Payment Line Invoice Layout [View Assignment]
Business Brand	Business Brand Layout [View Assignment]	Price Book	Price Book Layout [View Assignment]

Activity 5: Customer Service Rep Profile Creation

The screenshot shows the Salesforce Setup Profiles page. A new profile named "Customer Service Rep" is being created. The profile includes the following details:

- Name:** Customer Service Rep
- User License:** Salesforce Platform
- Description:** (empty)
- Created By:** Andreas Luy (8/23/2025, 6:51 PM)
- Modified By:** Andreas Luy (8/23/2025, 6:51 PM)

The "Custom Profile" checkbox is selected.

Page Layouts: This section lists various standard object layouts assigned to the profile, such as Global Layout, Lead Layout, Location Layout, and various Account, Opportunity, and Case layouts.

Object Milestones: This section lists various object milestones assigned to the profile, such as Order Layout, Order Product Layout, Payment Layout, and Payment Authorization Layout.

Operating Hours: This section lists various operating hours assigned to the profile, such as Standard Operating Hours Layout and Custom Operating Hours Layout.

Order: This section lists various order-related layouts assigned to the profile, such as Order Layout and Order Product Layout.

Order Product: This section lists various order product-related layouts assigned to the profile, such as Order Product Layout and Order Line Item Layout.

Payment: This section lists various payment-related layouts assigned to the profile, such as Payment Layout and Payment Authorization Layout.

Payment Authorization: This section lists various payment authorization-related layouts assigned to the profile, such as Payment Authorization Layout and Payment Authorization Adjustment Layout.

Payment Gateway: This section lists various payment gateway-related layouts assigned to the profile, such as Payment Gateway Layout and Payment Gateway Log Layout.

Payment Gateway Log: This section lists various payment gateway log-related layouts assigned to the profile, such as Payment Gateway Log Layout and Payment Gateway Log Layout.

Payment Group: This section lists various payment group-related layouts assigned to the profile, such as Payment Group Layout and Payment Group Log Layout.

Payment Line Invoice: This section lists various payment line invoice-related layouts assigned to the profile, such as Payment Line Invoice Layout and Payment Line Invoice Log Layout.

Price Book: This section lists various price book-related layouts assigned to the profile, such as Price Book Layout and Price Book Log Layout.

Creating the custom profiles such as the Travel Agent, Tour Guide, Finance Officer, Marketing Executive, and Customer Service Rep helped me design the role-based access for the application I am developing which is the Tours & Travels CRM.

Each profile was given permissions that matched the real-world responsibilities. This milestone helped me understand the importance of security and access control, making the system safe.

Milestone 24: Roles & Role Hierarchy

Activity 1: Creation of Travel Agent Manager Role

Role
Travel Agent Manager

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Travel Agent Manager
Siblings: SVP_Sales & Marketing, SVP_Customer_Service & Support, COO, SVP_Human_Resources, COO

[Users in Travel Agent Manager Role \(0\)](#)

Role Detail

Label	Travel Agent Manager	Edit Delete	
This role reports to	CEO	Role Name	Travel_Agent_Manager
Modified By	Andreas.Luy	Role Name as displayed on reports	
Modified On	8/23/2025, 11:03 AM	Sharing Groups	Role, Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities		
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases		

[Assign Users to Role](#) [New User](#)

[Users in Travel Agent Manager Role Help](#)

No records to display

Activity 2: Creating other Roles Under Travel Agent Manager Role

Role
Travel Agent

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Travel Agent Manager > Travel Agent

[Users in Travel Agent Role \(0\)](#)

Role Detail

Label	Travel Agent	Edit Delete	
This role reports to	Travel Agent Manager	Role Name	Travel_Agent
Modified By	Andreas.Luy	Role Name as displayed on reports	
Modified On	8/23/2025, 11:04 AM	Sharing Groups	Role, Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities		
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases		

[Assign Users to Role](#) [New User](#)

[Users in Travel Agent Role Help](#)

No records to display

Role
Travel Tour Guide

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Travel Agent Manager > Travel Agent > Travel Tour Guide

[Users in Travel Tour Guide Role \(0\)](#)

Role Detail

Label	Travel Tour Guide	Edit Delete	
This role reports to	Travel Agent	Role Name	Travel_Tour_Guide
Modified By	Andreas.Luy	Role Name as displayed on reports	
Modified On	8/23/2025, 11:05 AM	Sharing Groups	Role, Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities		
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases		

[Assign Users to Role](#) [New User](#)

[Users in Travel Tour Guide Role Help](#)

No records to display

Activity 3: Creating other Three Roles Under CEO Role

Role
Finance Officer Role

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Finance Officer Role
Siblings: SVP_Sales & Marketing, SVP_Customer_Service & Support, COO, SVP_Human_Resources, COO, Travel Agent Manager, Marketing Executive Role, Customer Service Rep Role

[Users in Finance Officer Role \(0\)](#)

Role Detail

Label	Finance Officer Role	Edit Delete	
This role reports to	CEO	Role Name	Finance_Officer_Role
Modified By	Andreas.Luy	Role Name as displayed on reports	
Modified On	8/23/2025, 11:06 AM	Sharing Groups	Role, Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities		
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases		

[Assign Users to Role](#) [New User](#)

[Users in Finance Officer Role Help](#)

No records to display

Role Marketing Executive Role

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Marketing Executive Role
Siblings: SVP_Sales & Marketing, SVP_Customer_Service & Support, CFO, SVP_Human_Resources, COO, Travel Agent Manager, Finance Officer Role, Customer Service Rep Role

[Users in Marketing Executive Role Role \[i\]](#)

Role Detail

Label	Marketing Executive Role	Role Name as displayed on reports
This role reports to	CEO	Role
Modified By	Andreas Luy 8/23/2025, 11:06 AM	Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities	
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases	

Users in Marketing Executive Role Role

[Assign Users to Role](#) [New User](#)

No records to display

Role Customer Service Rep Role

Below is the list of users assigned to this role. Click Edit to modify the role name. Click Assign Users to Role to assign existing users to this role. Click New User to create a user for this role.

Hierarchy: Angeles University Foundation > CEO > Customer Service Rep Role
Siblings: SVP_Sales & Marketing, SVP_Customer_Service & Support, CFO, SVP_Human_Resources, COO, Travel Agent Manager, Finance Officer Role, Marketing Executive Role

[Users in Customer Service Rep Role Role \[i\]](#)

Role Detail

Label	Customer Service Rep Role	Role Name as displayed on reports
This role reports to	CEO	Role
Modified By	Andreas Luy 8/23/2025, 11:06 AM	Role and Internal Subordinates
Opportunity Access	Users in this role can edit all opportunities associated with accounts that they own, regardless of who owns the opportunities	
Case Access	Users in this role can edit all cases associated with accounts that they own, regardless of who owns the cases	

Users in Customer Service Rep Role Role

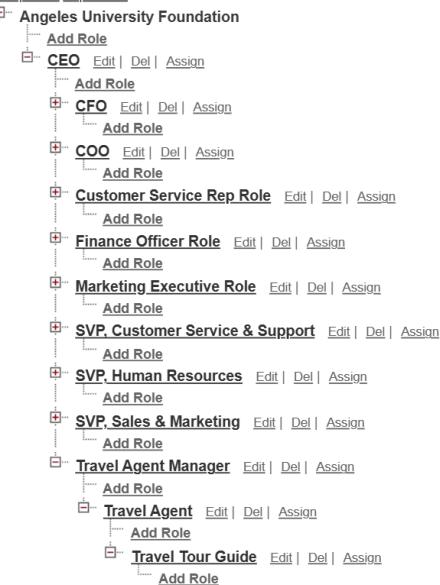
[Assign Users to Role](#) [New User](#)

No records to display

Role Hierarchy

Your Organization's Role Hierarchy

[Collapse All](#) [Expand All](#)



Building the Role Hierarchy for the Tours & Travels CRM made it clear for me how Salesforce supports structured data visibility. This milestone helped me appreciate how each role complements the profiles while the profiles control what the Users can do, and how roles control what data they can see.

Milestone 25: Permission Sets

Activity 1: Creating New Permission Set

The screenshot shows the 'Permission Sets' page in the Salesforce setup. A new permission set named 'Extra Permission For Travel Agent Manager' is being created. The 'Permission Set Overview' section displays basic details: API Name 'Extra_Permission_For_Travel_Agent_Manager', Namespace Prefix 'TravelAgentManager', and the user 'Andreas Luy' as the creator and last modifier. The 'Description' field is empty. Under the 'Apps' tab, various app-related permissions are listed, such as 'Assigned Apps', 'Assigned Connected Apps', 'Object Settings', 'App Permissions', 'Apex Class Access', 'Visualforce Page Access', 'External Data Source Access', 'Flow Access', 'Named Credential Access', 'External Credential Principal Access', 'Service Presence Statuses Access', and 'Custom Permissions'. A note at the bottom of the list states: 'Settings that apply to Salesforce apps, such as Sales, and custom apps built on the Lightning Platform' with a 'Learn More' link.

The screenshot shows the 'Object Settings' for the 'TravelPackages' object within the 'Extra Permission For Travel Agent Manager' permission set. The 'Tab Settings' section shows the 'Available' tab selected, with the 'Visible' tab highlighted. The 'Object Permissions' section lists permissions for the 'TravelPackages' object, including Read, Create, Edit, Delete, View All Records, Modify All Records, and View All Fields, all of which are currently enabled (indicated by checked checkboxes).

Permission Name	Enabled
Read	<input checked="" type="checkbox"/>
Create	<input checked="" type="checkbox"/>
Edit	<input checked="" type="checkbox"/>
Delete	<input checked="" type="checkbox"/>
View All Records	<input type="checkbox"/>
Modify All Records	<input type="checkbox"/>
View All Fields	<input type="checkbox"/>

Current Assignments						
<input type="checkbox"/> Full Name ↑	Active	Role	Profile	User License	Expires On	Add Assignment
<input type="checkbox"/> Michael Jackson	✓	Travel Agent Manager	Travel Agent Profile	Salesforce Platform		

Creating a permission set for the Travel Agent Manager showed me how Salesforce can make access management more adaptable. Instead of creating multiple profiles, I could simply assign an extra permission if needed!

Milestone 26: Sharing Settings

Activity 1: Creating Sharing Rule

The screenshot shows two related pages from the Salesforce setup interface.

Sharing Settings: This page lists sharing rules for various objects. A red box highlights the row for "Customer Info".

Object	Default Access	Access Level	Action
Work Step Template	Private	Private	✓
Work Type	Private	Private	✓
Work Type Group	Public Read/Write	Private	✓
Booking	Public Read/Write	Private	✓
BookingGuest	Controlled by Parent	Controlled by Parent	
Booking Payment	Public Read/Write	Private	✓
Customer Info	Private	Private	✓
Employee	Public Read/Write	Private	✓
Feedback	Public Read/Write	Private	✓
TravelPackage	Public Read/Write	Private	✓

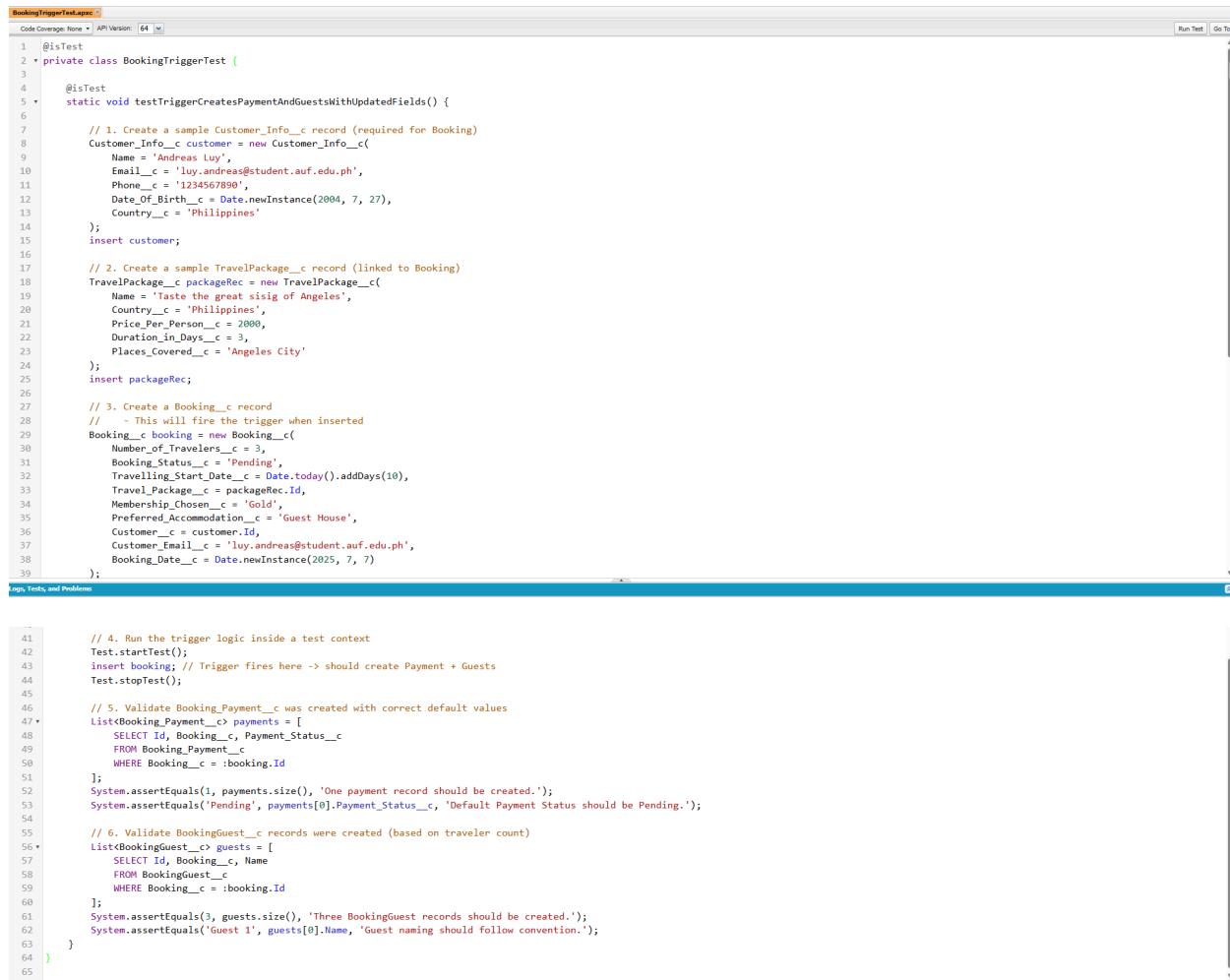
Customer Info Sharing Rules: This page shows a single rule for the "Customer Info" object.

Action	Criteria	Shared With	Access Level
Edit	Owner in Role_Travel Agent	Role_Travel Tour Guide	Read Only

Configuring the sharing rules for the Customer Info helped me see how Salesforce balances the security with usability. By setting the default permissions and access to private and then selectively sharing the customer records with the Tour Guide role, I ensured data privacy while still enabling collaboration. This milestone gave me a better understanding of Salesforce's layered security model.

Milestone 27: Test Classes

Activity 1: Creation of a Test Class



```
BookingTriggerTest.apc
Code Coverage: None • API Version: 64
1  @isTest
2  private class BookingTriggerTest {
3
4      @isTest
5      static void testTriggerCreatesPaymentAndGuestsWithUpdatedFields() {
6
7          // 1. Create a sample Customer__c record (required for Booking)
8          Customer__c customer = new Customer__c(
9              Name = 'Andreas Luy',
10             Email__c = 'luy.andreas@student.auf.edu.ph',
11             Phone__c = '1234567890',
12             Date_of_Birth__c = Date.newInstance(2004, 7, 27),
13             Country__c = 'Philippines'
14         );
15         insert customer;
16
17         // 2. Create a sample TravelPackage__c record (linked to Booking)
18         TravelPackage__c packageRec = new TravelPackage__c(
19             Name = 'Taste the great sisig of Angeles',
20             Country__c = 'Philippines',
21             Price_Per_Person__c = 2000,
22             Duration_In_Days__c = 3,
23             Places_Covered__c = 'Angeles City'
24         );
25         insert packageRec;
26
27         // 3. Create a Booking__c record
28         // - This will fire the trigger when inserted
29         Booking__c booking = new Booking__c(
30             Number_of_Travelers__c = 3,
31             Booking_Status__c = 'Pending',
32             Travelling_Start_Date__c = Date.today().addDays(10),
33             Travel_Package__c = packageRec.Id,
34             Membership_Choice__c = 'Gold',
35             Preferred_Accommodation__c = 'Guest House',
36             Customer__c = customer.Id,
37             Customer_Email__c = 'luy.andreas@student.auf.edu.ph',
38             Booking_Date__c = Date.newInstance(2025, 7, 7)
39         );
40
41         // 4. Run the trigger logic inside a test context
42         Test.startTest();
43         insert booking; // Trigger fires here -> should create Payment + Guests
44         Test.stopTest();
45
46         // 5. Validate Booking_Payment__c was created with correct default values
47         List<Booking_Payment__c> payments = [
48             SELECT Id, Booking__c, Payment_Status__c
49             FROM Booking_Payment__c
50             WHERE Booking__c = :booking.Id
51         ];
52         System.assertEquals(1, payments.size(), 'One payment record should be created.');
53         System.assertEquals('Pending', payments[0].Payment_Status__c, 'Default Payment Status should be Pending.');
54
55         // 6. Validate BookingGuest__c records were created (based on traveler count)
56         List<BookingGuest__c> guests = [
57             SELECT Id, Booking__c, Name
58             FROM BookingGuest__c
59             WHERE Booking__c = :booking.Id
60         ];
61         System.assertEquals(3, guests.size(), 'Three BookingGuest records should be created.');
62         System.assertEquals('Guest 1', guests[0].Name, 'Guest naming should follow convention.');
63     }
64 }
65 
```

Logs and Tests

User	Application	Operation	Time	Status	Read	Size	
Andreas Luy	Unknown	ApeX Test Handler	8/24/2025, 6:50:38 PM	Success	Unread	36.96 KB	
Logs						Tests	
Status	Test Run						
	Enqueued Time	Duration		Failures	Total	Overall Code Coverage	
✗ 707tg000000CetM	Sun Aug 24 2025 17:02:15 GM.			1	1	Class	
✗ 707tg000000CvzQ	Sun Aug 24 2025 18:01:01 GM.			1	1	Percent	
✗ 707tg000000CwP	Sun Aug 24 2025 18:01:04 GM.			1	1	Lines	
✗ 707tg000000CwU	Sun Aug 24 2025 18:01:04 GM.			1	1	Overall	32%
✗ 707tg000000CwV	Sun Aug 24 2025 18:01:04 GM.			1	1	BookingConfirmationEmailer	0%
✗ 707tg000000CwW	Sun Aug 24 2025 18:01:04 GM.			1	1	BookingReminderQuotaable	0%
✗ 707tg000000CwX	Sun Aug 24 2025 18:01:04 GM.			1	1	BookingReminderScheduler	0%
✗ 707tg000000CwY	Sun Aug 24 2025 18:01:04 GM.			1	1	BookingTriggerHandler	95%
✗ 707tg000000CwZ	Sun Aug 24 2025 18:01:04 GM.			1	1	PaymentReminderBatch	0%
✗ 707tg000000CwU	Sun Aug 24 2025 18:01:04 GM.			1	1	SchedulePaymentReminderBatch	0%
✗ 707tg000000CwJ	Sun Aug 24 2025 18:01:04 GM.			1	1	TravelPackageController	0%
✗ 707tg000000CwI	Sun Aug 24 2025 18:01:04 GM.			1	1		
✗ 707tg000000CwR	Sun Aug 24 2025 19:30:40 GM.			1	1		
✗ 707tg000000CwS	Sun Aug 24 2025 19:30:40 GM.			1	1		
✗ 707tg000000CfCN	Sun Aug 24 2025 18:33:22 GM.			1	1		
✓ 707tg000000CfDQ	Sun Aug 24 2025 18:34:49 GM.			0	1		
✓ 707tg000000CfE2	Sun Aug 24 2025 18:36:53 GM.			0	1		
✓ 707tg000000CfFQ	Sun Aug 24 2025 18:46:09 GM.			0	1		
✓ 707tg000000CfH4	Sun Aug 24 2025 18:50:34 GM.			0	1		

Writing the test classes for the Booking Trigger and BookingTriggerHandler gave me a hard time but my key takeaway for this is that in real-world situations, I was able to verify that the booking payment and guest records were being created as intended. This milestone made me understand the importance of testing.

Milestone 28: Preparing Test Cases & Fixing Defects

Activity 1: Test Case 1

Customer Creation

Test Case	Verify that a new Customer can be created successfully with all mandatory fields.
Environment	Includes <ul style="list-style-type: none">• OS: Windows 11• Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	Test Steps: <ul style="list-style-type: none">• Navigate to the Customer object page.• Fill in required fields.• Click Save
Expected Result	<ul style="list-style-type: none">• The Customer record should be saved successfully.• The record should appear on the Customers List page.
Status	PASSED

Test Evidences

The image displays three screenshots of a CRM application interface, likely Tours & Travels CRM, illustrating the process of creating and viewing a customer record.

Screenshot 1: New Customer Info
This screenshot shows the "New Customer Info" dialog box. The "Information" section contains fields for Customer ID (set to "Spongebob Squarepants"), Date Of Birth (set to "1/1/2000"), Customer Name ("Spongebob Squarepants"), Email ("squarepants@bikinibottom.com"), Phone ("0922 475-8391"), Country ("Philippines"), and City ("Angeles City"). The "Owner" field is populated with "Andreas Luy". Buttons at the bottom include "Cancel", "Save & New", and "Save".

Screenshot 2: Customer Info - Spongebob Squarepants
This screenshot shows the details of the newly created customer record for "Spongebob Squarepants".

Customer ID	Date Of Birth	1/1/2000
Customer Name	Age	25
Email	Owner	Andreas Luy
Phone		(0922) 475-8391
Country		Philippines
City		Angeles City
Created By	Last Modified By	Andreas Luy / 8/24/2025, 4:50 AM

The "Activity" section indicates "No activities to show." and provides instructions to "Get started by sending an email, scheduling a task, and more." A note states "No past activity. Past meetings and tasks marked as done show up here."

Screenshot 3: Recently Viewed
This screenshot shows the "Recently Viewed" list, which includes "Spongebob Squarepants" at the top, followed by other customer names: Kenneth Chin, Daniel Singh, Emily Clark, Michael Wong, Sarah Brown, David Kim, Maria Lopez, Alex Johnson, Jane Smith, and John Doe.

Activity 2: Test Case 2

Booking Creation

Test Case	Verify that a new booking can be created successfully with all mandatory fields.
Environment	Includes <ul style="list-style-type: none">• OS: Windows 11• Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	Test Steps: <ul style="list-style-type: none">• Navigate to the Booking object page.• Fill in required fields.• Click Save.
Expected Result	<ul style="list-style-type: none">• The Booking record should be saved successfully.• The record should appear on the Booking List page.• A Record related to Booking must be created in the Booking Payments object with Payment Status pending.• BookingGuests records must be created and maintained to exactly match the Number of Travellers value specified in the related Booking record..
Status	PASSED

Test Evidences

New Booking

Information

Booking Number	Number of Travelers
*Booking Name	1
A trip to bikini bottom	Booking Status
Spongebob Squarepants	Pending
*Customer	Travel Cost Per Person
spongepants@bikini.bottom.com	\$0.00
*Customer Email	This field is calculated upon save
Customer	Total Travel Amount
	\$0.00
*Travel Package	Accommodation Amount per Person per Day
USA East Coast Explorer	\$0.00
This field is calculated upon save	Total Accommodation Amount
Booking Date	\$0.00
8/22/2025	This field is calculated upon save
Traveling Start Date	Cancellation Reason
9/1/2025	Total Accommodation Amount
Traveling End Date	\$0.00
This field is calculated upon save	This field is calculated upon save
Trip Type	Total Billing Amount
Solo	\$0.00
This field is calculated upon save	This field is calculated upon save
*Membership Chosen	Membership Chosen
VIP	Villa
View all dependencies	Preferred Accommodation
Include Travel Insurance	Villa
Require Visa Assistance	View all dependencies
Require Tour Guide	Include Travel Insurance
Guide Assigned	Require Visa Assistance
Mr. Krabs	Require Tour Guide
Preferred Guide Language	Guide Assigned
English	Mr. Krabs
No of Booking Guests Info Available	Preferred Guide Language
0	English
This field is calculated upon save	

Cancel Save & New Save

Booking Details

A trip to bikini bottom

Booking "A trip to bikini bottom" was created.

Related Details

Information

Booking Number	Number of Travelers
11	1
Booking Name	Booking Status
A trip to bikini bottom	Pending
Customer	Travel Cost Per Person
Spongebob Squarepants	\$0.00
Customer Email	Total Travel Amount
spongepants@bikini.bottom.com	\$15,000.00
Travel Package	Accommodation Amount per Person per Day
USA East Coast Explorer	\$6,000.00
Booking Date	Cancellation Reason
8/22/2025	Total Accommodation Amount
Traveling Start Date	\$6,000.00
9/1/2025	Total Billing Amount
Traveling End Date	\$21,000.00
9/21/2025	Owner
Trip Type	Andreas Luz
Solo	Owner
Membership Chosen	Andreas Luz
VIP	Owner
Preferred Accommodation	Villa
Villa	Preferred Accommodation
Include Travel Insurance	Villa
Require Visa Assistance	Include Travel Insurance

Activity

Upcoming & Overdue

No activities to show.

Get started by sending an email, scheduling a task, and more.

No part activity. Past meetings and tasks marked as done show up here.

Filters: All time • All activities • All types

Refresh • Expand All • View All

Booking Details

A trip to bikini bottom

Related **Details**

- BookingGuests (1)**
 - Guest 1
- Booking Payments (1)**
 - Booking Payment Name: a04g0000009pf0
- Feedbacks (0)**
- Approval History (0)**

Activity

No activities to show.

Upcoming & Overage

No past activity. Past meetings and tasks marked as done show up here.

Booking Guest Details

Guest 1

Related **Details**

Booking **A trip to bikini bottom** **Booking Guest Name** **Guest 1** **Country** Maldives **City**

Age 18 **Passport Number**

Gender **Mrs Required**

Relation with Customer

Age Category Adults **Special Needs**

Created by **Last Modified by**

Booking Payment Details

a04g1000009Lpf0

Related **Details**

Payment Number 12 **Payment Method**

Customer Name **Payment Reference Number**

SquareTrade Squarepoints **Payment Mode Details**

Booking **A trip to bikini bottom** **Payment Receipt Sent**

Payment Date 8/22/2025 **Owner** **Andreas Luy**

Total Bill Amount \$21,000.00 **Last Modified By** **Andreas Luy**

Payment Status Pending **Notes**

Booking Payment Name a04g0000009pf0

Activity

No activities to show.

Upcoming & Overage

No past activity. Past meetings and tasks marked as done show up here.

Activity 3: Test Case 3

Payment Status Update and Email Notification

Test Case	Verify whether the Booking Status is Confirmed in Booking Object when Payment Status field is updated to completed in Booking Payment Object. And also verify whether the customer received the mail regarding Booking confirmation and payment completed
Environment	Includes <ul style="list-style-type: none">• OS: Windows 11• Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	Test Steps: <ul style="list-style-type: none">• Navigate to the Booking Payment object page.• Update Payment Status field to Completed.• Click Save
Expected Result	<ul style="list-style-type: none">• The Booking Payment record should be Updated successfully.• The Booking record should be Updated successfully and Booking Status must be Confirmed.• The customer must receive the mail regarding Booking Confirmation and Payment completion
Status	PASSED

Test Evidences

Booking
A trip to bikini bottom

Related **Details**

Information

Booking Number 11	Number of Travelers 1
Booking Name A trip to bikini bottom	Booking Status Pending
Customer Spongebob Squarepants	Travel Cost Per Person \$15,000.00
Customer Email squarepants@bikini.bottom.com	Total Travel Amount \$15,000.00
Travel Package USA East Coast Explorer	Accommodation Amount per Person per Day \$6,000.00
Booking Date 8/22/2025	Cancellation Reason
Travelling Start Date 9/1/2025	Total Accommodation Amount \$6,000.00
Travelling End Date 9/21/2025	Total Billing Amount \$21,000.00
Trip Type Solo	Owner  Andreas Luy
Membership Chosen VIP	
Preferred Accommodation Villa	
Include Travel Insurance <input checked="" type="checkbox"/>	
Require Visa Assistance	

Booking Guests (1) [New](#)

Guest Name Guest_1

View All

Booking Payments (1) [New](#)

Booking Payment Name a04gl0000009Lpf3
--

View All

Feedbacks (0) [New](#)

Approval History (0)

Booking Payment
a04gL000009Lpf3

Details	
Payment Number 12	Payment Method
Customer Name Spongebob Squarepants	Payment Reference Number
Booking A trip to bikini bottom	Payment Mode Details
Payment Date 8/22/2025	Payment Receipt Sent
Total Bill Amount \$21,000.00	Owner
Payment Status Pending	Andreas Luy
Notes	
Booking Payment Name a04gL000009Lpf3	
Last Modified By Andreas Luy, 8/24/2025, 4:58 AM	

Booking Payment
a04gL000009Lpf3

Details	
Payment Number 12	Payment Method
Customer Name Spongebob Squarepants	Payment Reference Number
Booking A trip to bikini bottom	Payment Mode Details
Payment Date 8/22/2025	Payment Receipt Sent
Total Bill Amount \$21,000.00	Owner
Payment Status Completed	Andreas Luy
Notes	
Booking Payment Name a04gL000009Lpf3	
Last Modified By Andreas Luy, 8/24/2025, 5:04 AM	

Email Received

Booking Confirmed: A trip to the bottom of Europe ➤ [Inbox](#)

 **Andreas Luy** via [vblabdy5gwevngfd44sp4f8m7frt4qhv3k58d4tzgl-8vdpduam-car98.lnc.salesforce.com](#)
to me
Dear Customer,

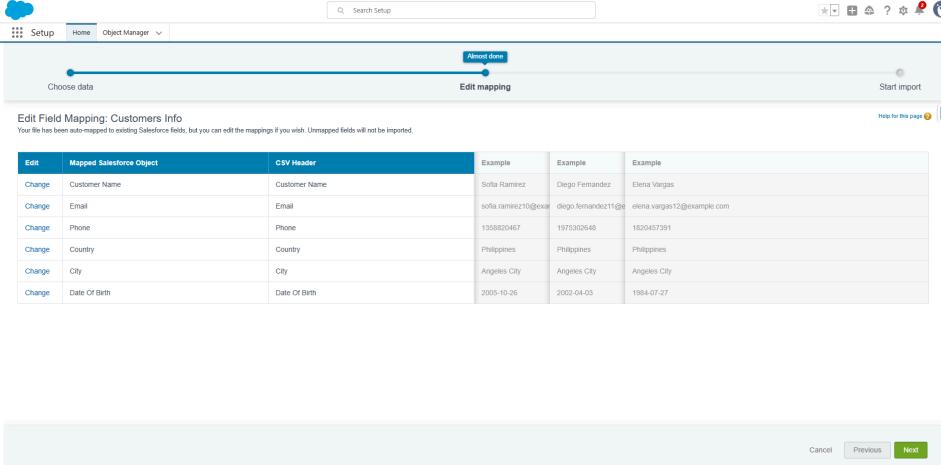
Your booking has been confirmed. Please find the details below.
 Booking ID: A trip to the bottom of Europe
 Total Bill Amount Paid: \$22500.00

Thank you for booking with us!

[Reply](#) [Forward](#)

8:10PM (1 minute ago)   

Activity 4: Test Case 4: Other Test Cases

Test Case	Import Customers via Data Import Wizard																																										
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit) 																																										
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Setup → Data Import Wizard • Choose Object: Customer Info • Upload CSV • Map fields • Start Import → Review results 																																										
Expected Result	<ul style="list-style-type: none"> • 10 records inserted with no invalid rows 																																										
Status	PASSED																																										
Test Evidences	 <table border="1"> <thead> <tr> <th>Edit</th> <th>Mapped Salesforce Object</th> <th>CSV Header</th> <th>Example</th> <th>Example</th> <th>Example</th> </tr> </thead> <tbody> <tr> <td>Change</td> <td>Customer Name</td> <td>Customer Name</td> <td>Sofia Ramirez</td> <td>Diego Fernandez</td> <td>Elena Vargas</td> </tr> <tr> <td>Change</td> <td>Email</td> <td>Email</td> <td>sofia.ramirez210@oscar</td> <td>diego.fernandez211@oscar</td> <td>elena.vargas12@example.com</td> </tr> <tr> <td>Change</td> <td>Phone</td> <td>Phone</td> <td>13008820467</td> <td>1975302648</td> <td>1820457391</td> </tr> <tr> <td>Change</td> <td>Country</td> <td>Country</td> <td>Philippines</td> <td>Philippines</td> <td>Philippines</td> </tr> <tr> <td>Change</td> <td>City</td> <td>City</td> <td>Angeles City</td> <td>Angeles City</td> <td>Angeles City</td> </tr> <tr> <td>Change</td> <td>Date Of Birth</td> <td>Date Of Birth</td> <td>2005-10-26</td> <td>2002-04-03</td> <td>1984-07-27</td> </tr> </tbody> </table>	Edit	Mapped Salesforce Object	CSV Header	Example	Example	Example	Change	Customer Name	Customer Name	Sofia Ramirez	Diego Fernandez	Elena Vargas	Change	Email	Email	sofia.ramirez210@oscar	diego.fernandez211@oscar	elena.vargas12@example.com	Change	Phone	Phone	13008820467	1975302648	1820457391	Change	Country	Country	Philippines	Philippines	Philippines	Change	City	City	Angeles City	Angeles City	Angeles City	Change	Date Of Birth	Date Of Birth	2005-10-26	2002-04-03	1984-07-27
Edit	Mapped Salesforce Object	CSV Header	Example	Example	Example																																						
Change	Customer Name	Customer Name	Sofia Ramirez	Diego Fernandez	Elena Vargas																																						
Change	Email	Email	sofia.ramirez210@oscar	diego.fernandez211@oscar	elena.vargas12@example.com																																						
Change	Phone	Phone	13008820467	1975302648	1820457391																																						
Change	Country	Country	Philippines	Philippines	Philippines																																						
Change	City	City	Angeles City	Angeles City	Angeles City																																						
Change	Date Of Birth	Date Of Birth	2005-10-26	2002-04-03	1984-07-27																																						

The screenshot shows the 'Bull Data Load Jobs' setup interface. At the top, there's a header with a gear icon labeled 'SETUP' and the title 'Bull Data Load Jobs'. Below the header, it says 'Bulk Data Load Job' and '750gL00000Bkfccx'. A message says 'View the details of a bulk data load job.' with a link 'Back to List - Bulk Data Load Jobs'. On the right, there's a 'Help for this Page' button.

The main area is titled 'Bulk Data Load Job Detail' with a 'Reload' button. It displays various job parameters:

- Job ID:** 750gL00000Bkfccx
- Submitted By:** Andreas Lutz
- Start Time:** 8/24/2025, 7:39 PM PST
- End Time:** 8/24/2025, 7:39 PM PST
- Time to Complete (In [min:ss]):** 00:01
- Object:** Customer Info
- External ID Field:**
- Content Type:** CSV
- Concurrency Mode:** Parallel
- API Version:** 64.0

On the right side of the detail card, there's a summary table:

Status	Closed
Total Processing Time (ms)	128
API Active Processing Time (ms)	72
Avg Processing Time (ms)	1
Jobs Processing	1
Processing Time (ms)	1

Below the detail card is a section titled 'Batches' with a 'Reload' button. It lists one batch entry:

View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Avg Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Message	Status
View Request	View Result	751gl00000gDev	8/24/2025, 7:39 PM	8/24/2025, 7:39 PM	128	72	1	10	0	0	0	Completed

Test Case	Create Employee Record
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Go to Employees tab → Click New • Enter details: Name, Role = Guide, Availability Status = Available • Save
Expected Result	<ul style="list-style-type: none"> • The employee record is created successfully and appears in the Employees list view.
Status	PASSED

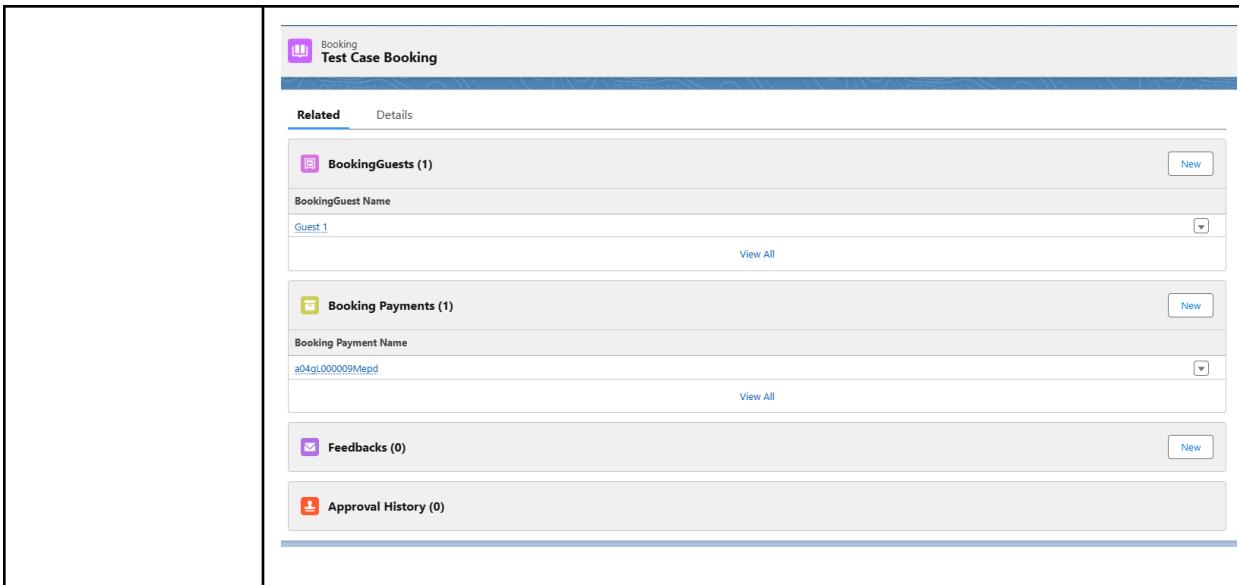
Test Evidences

The image consists of three vertically stacked screenshots from a Salesforce application, likely for a travel agency.

- Screenshot 1: New Employee Creation**
A modal window titled "New Employee" is open. It shows a "Profile Picture" placeholder with a cartoon character. Below it are input fields for "Employee Name" (Michael Reeves), "Email" (reeves.michael@gmail.com), "Phone" ((194) 234-5678), and "Role". A "Save" button is at the bottom right.
- Screenshot 2: Employee Record Detail View**
A detailed view of Michael Reeves' record. Fields include Employee Name, Email, Phone, Department (Travel Operations), Employment Type (Full-Time), Salary (\$50000.00), and more. The "Address Details" section is collapsed. The "Last Modified By" field shows "Andreas Luy" with a timestamp of 8/24/2025, 8:59 PM.
- Screenshot 3: Recently Viewed Employee List**
A list titled "Recently Viewed" showing 10 employees. Michael Reeves is the first item in the list, indicated by a checked checkbox next to his name.

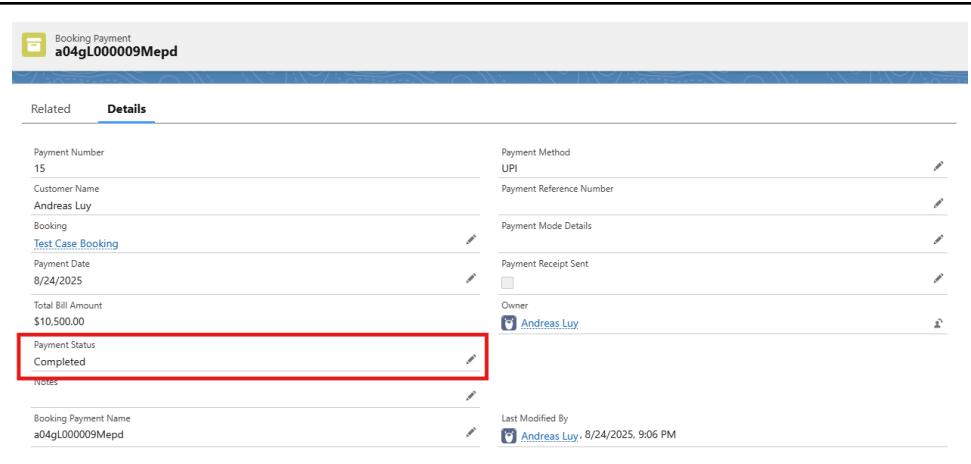
Test Case	Create Booking & Auto-Generate Related Records
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)

Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> Go to Bookings tab → Click New Select a Customer, Travel Package, Guide Assigned (Employee), enter # of Travelers Save
Expected Result	<ul style="list-style-type: none"> Booking record is created. Related Booking Payment record auto-created with status "Pending". Booking Guests auto-created based on # of Travelers.
Status	PASSED
Test Evidences	



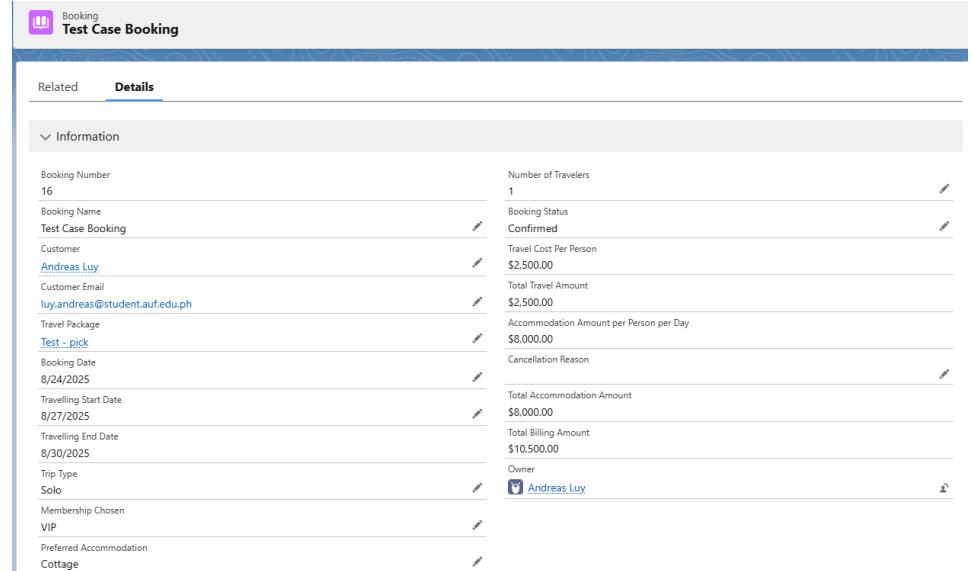
Test Case	Update Payment Status → Email Notification
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Open a Booking Payment record • Change status from "Pending" → "Completed" • Save
Expected Result	<ul style="list-style-type: none"> • Payment record updated. • Email notification sent to the Customer's email address
Status	PASSED

Test Evidences



Booking Payment
a04gL000009Mepd

Related	Details
Payment Number	15
Customer Name	Andreas Luy
Booking	Test Case Booking
Payment Date	8/24/2025
Total Bill Amount	\$10,500.00
Payment Status	Completed
Notes	
Booking Payment Name	a04gL000009Mepd
Last Modified By	Andreas Luy , 8/24/2025, 9:06 PM



Booking
Test Case Booking

Related	Details
Information	
Booking Number	16
Booking Name	Test Case Booking
Customer	Andreas Luy
Customer Email	luy.andreas@student.auf.edu.ph
Travel Package	Test..pick
Booking Date	8/24/2025
Travelling Start Date	8/27/2025
Travelling End Date	8/30/2025
Trip Type	Solo
Membership Chosen	VIP
Preferred Accommodation	Cottage
Include Travel Insurance	<input checked="" type="checkbox"/>
Number of Travelers	1
Booking Status	Confirmed
Travel Cost Per Person	\$2,500.00
Total Travel Amount	\$2,500.00
Accommodation Amount per Person per Day	\$8,000.00
Cancellation Reason	
Total Accommodation Amount	\$8,000.00
Total Billing Amount	\$10,500.00
Owner	Andreas Luy

Booking Confirmed: Test Case Booking [Inbox](#)

 Andreas Luy [a04gL000009Mepd](#) (noreply@bncregistration.com) to me 12:07PM (0 minutes ago)

Dear Customer,

Your booking has been confirmed. Please find the details below.

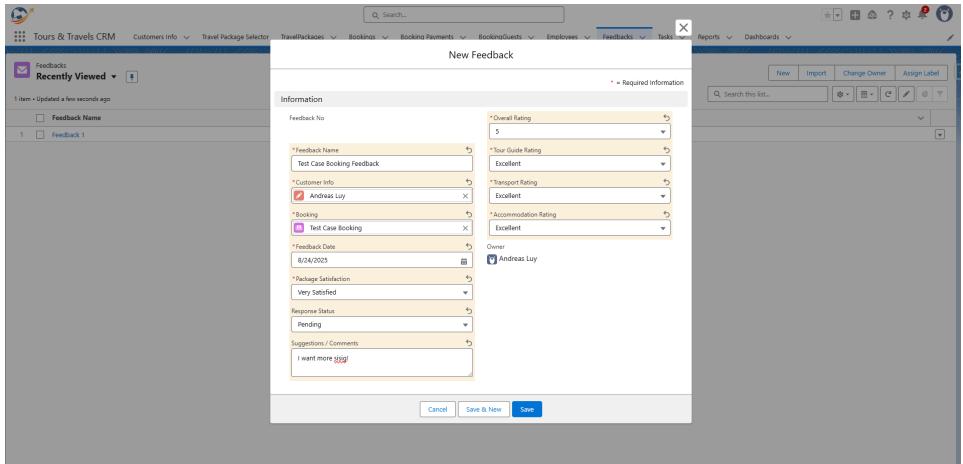
Booking ID: Test Case Booking
Total Bill Amount Paid: \$10500.00

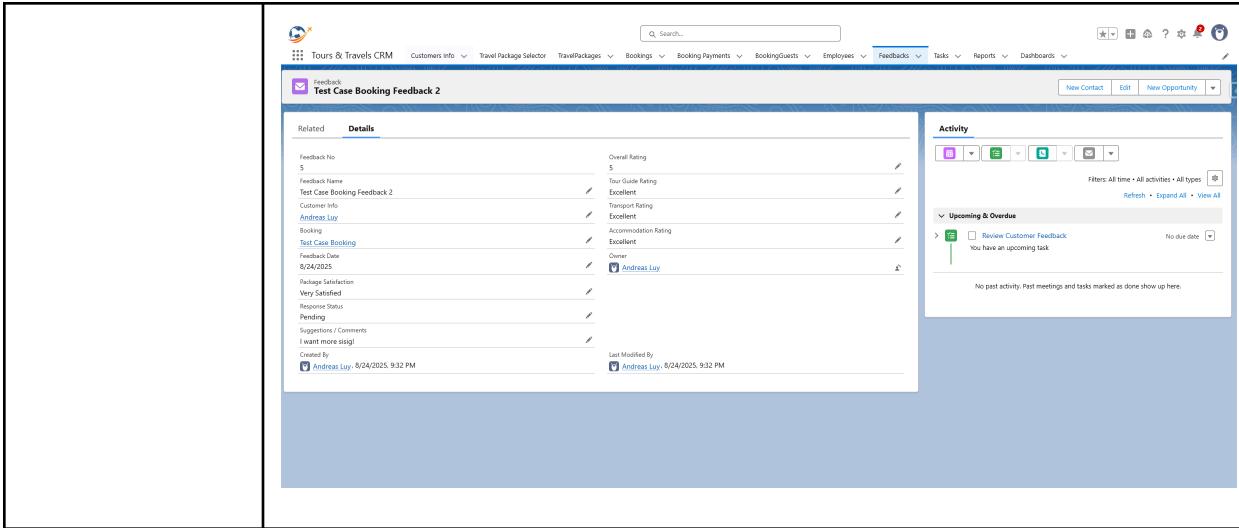
Thank you for booking with us!

[Reply](#) [Forward](#)

Test Case

Submit Feedback

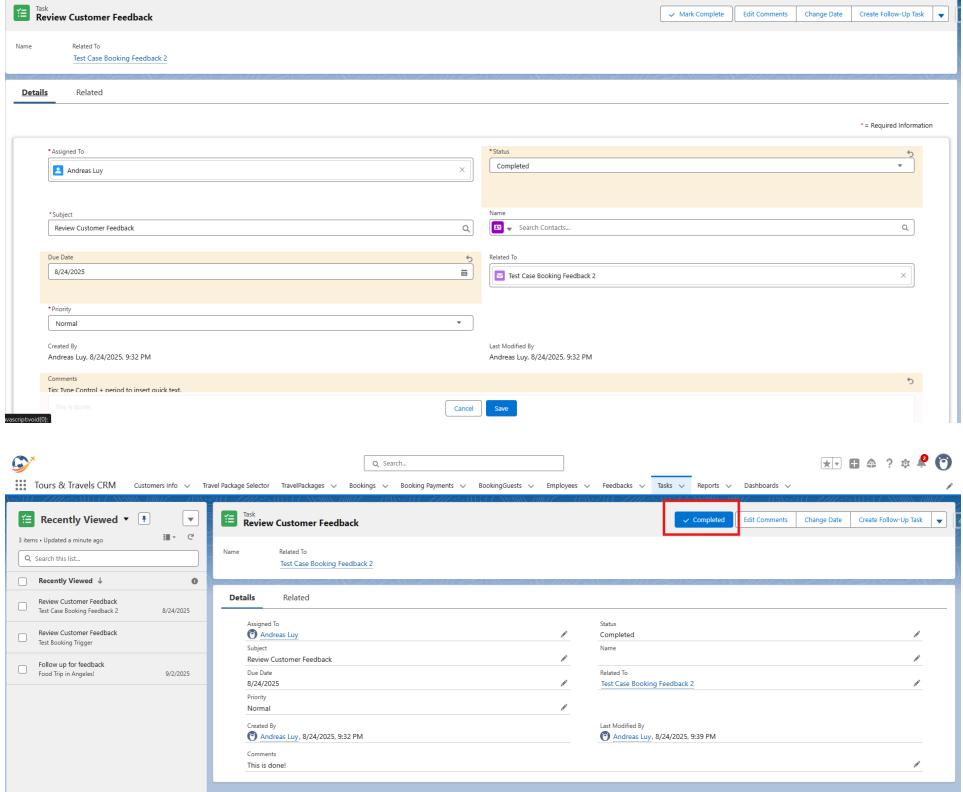
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Go to Feedbacks tab → Click New Feedback • Select Customer, Booking, enter Feedback details + rating • Save
Expected Result	<ul style="list-style-type: none"> • Feedback record created and linked to both Customer Info and Booking.
Status	PASSED
Test Evidences	



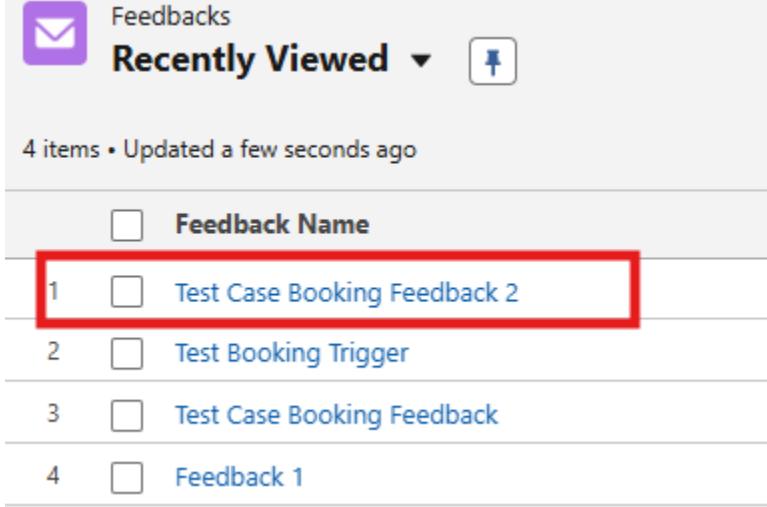
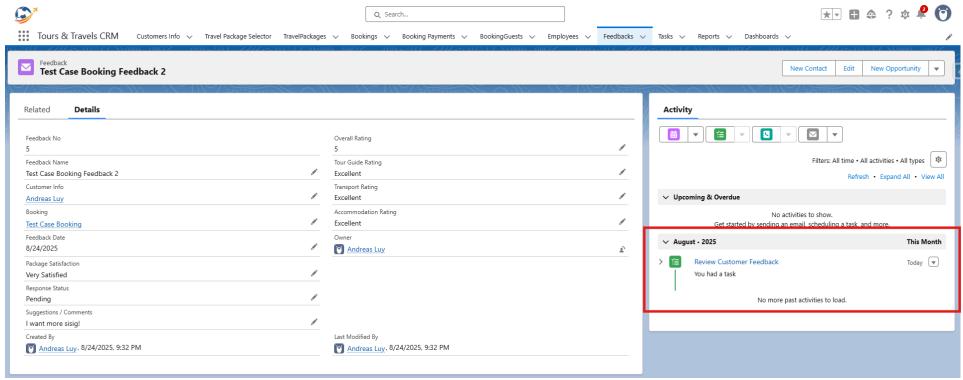
Test Case	Auto-Generate Task after new Feedback record
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Go to Feedbacks tab → Click the recently created Feedback • Verify that there's an upcoming task on the right column
Expected Result	<ul style="list-style-type: none"> • An upcoming task is created after the feedback a new Feedback record is submitted.
Status	PASSED

Test Evidences

Test Case	Verify Task is completed
Environment	<p>Includes</p> <ul style="list-style-type: none"> • OS: Windows 11 • Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> • Go to the Tasks tab, click the latest task record. • Click edit and fill in the necessary details • Click Save • Check if the Task is marked as completed

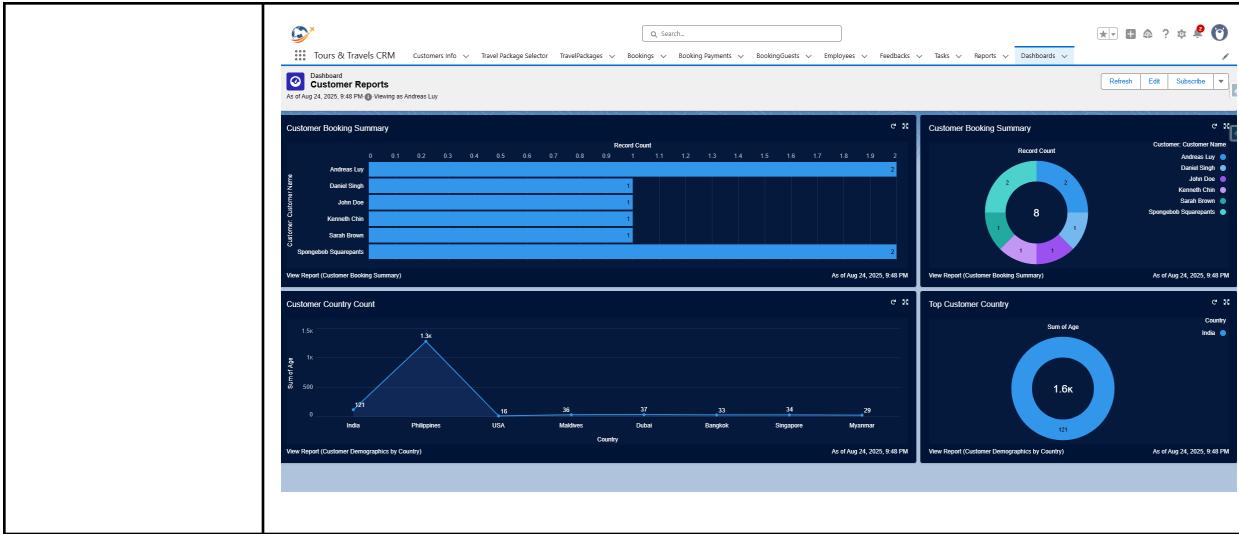
Expected Result	<ul style="list-style-type: none"> The upcoming task created will be marked as Completed
Status	PASSED
Test Evidences	

Test Case	Verify Task is completed
Environment	<p>Includes</p> <ul style="list-style-type: none"> OS: Windows 11 Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> Go to the Feedbacks tab

	<ul style="list-style-type: none"> Click the Feedback record related to it - "Test Case Booking Feedback 2"
Expected Result	<ul style="list-style-type: none"> The upcoming task's status is updated and completed.
Status	PASSED
Test Evidences	 

Test Case	Rechecking the Dashboard
Environment	Includes <ul style="list-style-type: none"> OS: Windows 11

	<ul style="list-style-type: none"> Browser: Google Chrome Version 139.0.7258.129 (Official Build) (64-bit)
Steps to Reproduce	<p>Test Steps:</p> <ul style="list-style-type: none"> Go to the Dashboards Tab Click the created Dashboards from the recent milestones Refresh the charts to see if the data from the reports are up to date.
Expected Result	<ul style="list-style-type: none"> The charts show the up-to-date data from the created reports
Status	PASSED
Test Evidences	<p>Before</p> <p>The screenshot displays the CRM interface with the following data:</p> <ul style="list-style-type: none"> Customer Booking Summary: A horizontal bar chart showing record counts for five customers: Daniel Singh (0.95), John Doe (1), Kenneth Chan (1), and Sarah Brown (1). Customer Booking Summary (Donut Chart): A donut chart showing the distribution of record counts across four categories, with a total count of 4. Customer Country Count: A line chart showing the sum of age across different countries. The data points are: India (29), Philippines (32), USA (19), Maldives (36), Dubai (37), Bangkok (33), Singapore (34), and Myanmar (29). Top Customer Country: A donut chart showing the sum of age for the top customer country, which is India at 246. <p>After Refreshing</p>



For this activity, I created and executed multiple test cases to validate the end-to-end functionality of the Tours & Travels CRM application. The tests covered both record creation across different modules (Customer Info, Travel Packages, Employees, Bookings, Booking Payments, and Feedback) as well as system behaviors such as email notifications, validation rules, automation, and dashboard refreshes.

The results confirmed that key workflows function as expected:

- When a customer makes a booking, the system automatically generates related payment and guest records.
- Updating a payment status to Completed triggers an email confirmation to the customer and updates the booking status.
- Submitting customer feedback links the record to both the booking and the customer, while also auto-generating a follow-up Task for employees.
- Dashboards and reports accurately refresh to display the latest operational data.

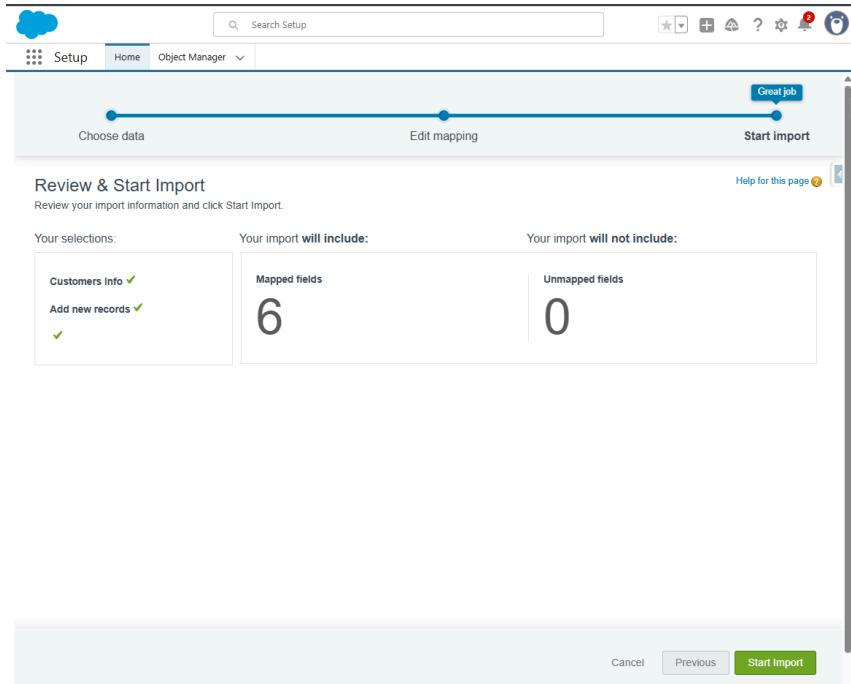
In a real-world scenario, the test cases I covered will ensure that core business processes — from booking to payment to customer feedback — are handled consistently and reliably. This can reduce human error, enforces validation rules, prevents incomplete data entry, ensuring that the employees are notified of follow-up actions through the use of automation.

Overall, this milestone emphasized how structured testing can strengthen the system's reliability, build Users confidence, and support the smooth handling of day-to-day operations like in a travel and tourism business.

Milestone 29: Data Import Wizard

Activity 1: Importing Records into Customer Info, TravelPackage and Employee Objects

Customer Info



BULK DATA LOAD JOBS

Bulk Data Load Job: **750gL00000BiazA** [Help for this Page](#)

View the details of a bulk data load job.

[« Back to List: Bulk Data Load Jobs](#)

Bulk Data Load Job Detail		Reload			
Job ID	750gL00000BiazA	Job Type	Bulk V1	Status	Closed
Submitted By	Andreas Luy	Operation	Insert	Total Processing Time (ms)	96
Start Time	8/24/2025, 8:54 AM PST	Queued Batches	0	API Active Processing Time (ms)	45
End Time	8/24/2025, 8:54 AM PST	In Progress Batches	0	Apex Processing Time (ms)	1
Time to Complete ([hh]:mm:ss)	00:01	Completed Batches	1		
Object	Customer Info	Failed Batches	0		
External ID Field		Progress	100%		
Content Type	CSV	Records Processed	20		
Concurrency Mode	Parallel	Records Failed	0		
API Version	64.0	Retries	0		

[Reload](#)

Batches											
View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State
View Request	View Result	751gL000009f06x	8/24/2025, 8:54 AM	8/24/2025, 8:54 AM	96	45	1	20	0	0	Completed

Travel Package

Setup [Home](#) [Object Manager](#) [Great job!](#)

[Search Setup](#) [Help for this page](#)

Choose data [Edit mapping](#) [Start import](#)

Review & Start Import
Review your import information and click Start Import.

Your selections: **TravelPackages ✓** Your import will include: **19** Your import will not include: **2**

[Cancel](#) [Previous](#) [Start Import](#)

BULK DATA LOAD JOBS

Bulk Data Load Job
750gL00000Bij4j

View the details of a bulk data load job.

< Back to List: Bulk Data Load Jobs

Bulk Data Load Job Detail		Reload			
Job ID	750gL00000Bij4j	Job Type	Bulk V1	Status	Closed
Submitted By	Andreas Luy	Operation	Insert	Total Processing Time (ms)	138
Start Time	8/24/2025, 8:43 AM PST	Queued Batches	0	API Active Processing Time (ms)	67
End Time	8/24/2025, 8:43 AM PST	In Progress Batches	0	Apex Processing Time (ms)	0
Time to Complete ([hh]:mm:ss)	00:01	Completed Batches	1		
Object	TravelPackage	Failed Batches	0		
External ID Field		Progress	100%		
Content Type	CSV	Records Processed	20		
Concurrency Mode	Parallel	Records Failed	0		
API Version	64.0	Retries	0		

Reload

Batches

View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Mes
View Request	View Result	751gL000009eryD	8/24/2025, 8:43 AM	8/24/2025, 8:43 AM	138	67	0	20	0	0	

Employee

Setup Home Object Manager ▾

Search Setup

Great job!

Choose data Edit mapping Start import

Review & Start Import

Review your import information and click Start Import.

Your selections: Your import will include: Your import will not include:

Employees ✓ Add new records ✓	Mapped fields 15	Unmapped fields 0
----------------------------------	----------------------------	-----------------------------

Cancel Previous Start Import

Bulk Data Load Job
750gL00000Bigmq

View the details of a bulk data load job.
« Back to List: Bulk Data Load Jobs

Bulk Data Load Job Detail		Reload			
Job ID	750gL00000Bigmq	Job Type	Bulk V1	Status	Closed
Submitted By	Andreas Luv	Operation	Insert	Total Processing Time (ms)	99
Start Time	8/24/2025, 9:04 AM PST	Queued Batches	0	API Active Processing Time (ms)	63
End Time	8/24/2025, 9:04 AM PST	In Progress Batches	0	Apex Processing Time (ms)	0
Time to Complete (hh:mm:ss)	00:01	Completed Batches	1		
Object	Employee	Failed Batches	0		
External ID Field		Progress	100%		
Content Type	CSV	Records Processed	20		
Concurrency Mode	Parallel	Records Failed	0		
API Version	64.0	Retries	0		

[Reload](#)

Batches											
View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State M
View Request	View Result	751gL000009ewma	8/24/2025, 9:04 AM	8/24/2025, 9:04 AM	99	63	0	20	0	0	More

The Data Import Wizard is a simple and Users-friendly tool that allows the import of small to medium data volumes and provides field mapping to ensure data accuracy during the import process.

Importing the records for the three objects was one of the most challenging parts of the project. Despite preparing the CSV files carefully, rechecking the fields correctly, I still ran into multiple issues where the data failed to record successfully. This milestone reminded me how important and strict data formatting and accuracy is on Salesforce. It tested my patience, but it was very rewarding once I saw the 20 records successfully inserted.

Phase 5: Deployment, Documentation & Maintenance

Milestone 30: Maintenance, Monitoring & Troubleshooting

UI Validation of Features

All the developed features were thoroughly validated using the Salesforce Users interface. Each tab - including the Customer Info, Employees, Travel Packages, Bookings, Payments, and Feedback were tested to ensure that the forms, required fields, lookups, and related lists functioned as expected. This confirmed that the Users can perform the core operations directly from the User Interface without encountering any navigation issues.

Functional Testing of Scenarios

The critical automation processes such as the Flows, Approval Processes, Workflow Rules, Process Builder actions, and Apex triggers were tested with realistic dummy data (to prevent data privacy concerns) inputs. For example, updating the Booking payment status to Completed will automatically trigger an email notification to the customer and update the related Booking status to Confirmed. These tests have validated that all backend logic were executed as intended.

End-to-End Business Flow Verification

The complete business flow was executed to verify smooth data transitions:

1. Customer creation (via manual entry or Data Import Wizard).
2. Booking creation linked to Customer, Travel Package, and Employee Guide.
3. Booking Payment auto-generation with status Pending.
4. Booking Guests auto-creation based on traveler count.
5. Feedback submission linked to both Booking and Customer.
6. Task auto-generation assigned to the responsible Employee upon Feedback creation.

This confirmed that the Tours & Travels CRM operates as a fully integrated system.

Data Accuracy Checks

Validation focused on the integrity of each relationship between the objects. The Bookings were properly linked to the Customers Info, Travel Packages, and Employees, while the Booking Payment and Booking Guest accurately reflected the details of the parent Booking. The validation rules prevented incomplete or invalid data entry, such as any attempts to save a Booking without specifying the number of travelers

Monitoring Scheduled Jobs and Flows

A review of Scheduled Flows, Apex Jobs, and Queueable Batch processes confirmed that all automated jobs executed successfully without runtime failures. Exception logs were examined to ensure that recurring operations — such as batch updates or scheduled triggers — completed as expected.

Debug Logs Review for Apex & Workflows

Debug logs were generated and analyzed for Apex Triggers and Classes, particularly those handling Booking and Payment logic. The logs confirmed that records were processed without recursion, infinite loops, or governor limit issues. This step also helped identify performance metrics, ensuring optimized code execution.

Error Handling and User Notifications

Both system-generated and custom error messages were tested. Validation errors (e.g., preventing duplicate Customer creation or bookings with zero travelers) were displayed in a clear, Users-friendly format. Additionally, email notifications were verified to alert customers regarding payment completion and booking confirmation. This ensures that Users are guided effectively when issues arise.

Security & Access Testing

Profiles, Permission Sets, and Role Hierarchies were reviewed to ensure compliance with data visibility rules. For instance, Travel Agents had access to Customer and Booking records but not to Finance-only Payment configurations, while Administrators maintained full system access. These checks confirmed that the principle of least privilege was consistently enforced.

Duplicate & Validation Rule Checks

Duplicate Rules were tested by attempting to create Customer Info records with identical email addresses, and the system correctly flagged the duplication. Validation Rules were also validated, such as ensuring that the *Number of Travelers* field must be greater than zero for any Booking. These controls ensure data cleanliness and adherence to business rules.

Report and Dashboard Accuracy

Reports and dashboards were reviewed for accuracy and freshness. For example, the Monthly Revenue Dashboard was refreshed after adding new Bookings and Payments, and the updated data reflected correctly in the charts. This ensures that decision-makers can rely on dashboards for real-time insights.

Dynamic Form Behavior

Dynamic Forms were tested to ensure contextual display of fields. For example, certain fields in the Feedback object only appeared when specific conditions were met (e.g., *Feedback Rating* field displayed only when *Feedback Type* = Survey). This improved the Users experience by reducing clutter and presenting only relevant fields.

Cross-Browser and Device Testing

The Tours & Travels CRM app was tested across different browsers such as Google Chrome, Microsoft Edge, and Brave. The functionality and responsiveness were consistent across different browsers using the Development Tools (F12, CTRL+SHIFT+I, or Right Click and Inspect). This ensured that the Tours & Travels CRM Application was accessible to Users regardless of their preferred device or browser.

Milestone 31: Documentation

Project Demo Video Presentation

 Salesforce Capstone Recording and Documentation

Summary & Conclusion

The Tours & Travels CRM capstone project successfully showcased the design development, testing, and the maintenance of a custom Salesforce application tailored for the Travel and Tourism industry. From the initial creation of custom objects to the integration of automation and dashboards, the project covered the entire CRM lifecycle. Each of the phases on the capstone project contributed to a scalable and user-friendly system capable of simulating real world travel agency operations.

The project integrated the core Salesforce features including the use of custom objects, validation rules, approval processes, roles, reports, dashboards, flows, workflows, process builders, and Apex triggers. The automation ensured that the critical processes such as auto-creating Booking Payments, Booking Guests, and Tasks from Feedback were handled seamlessly. Security and data integrity were maintained through Profiles, permission sets, validation rules, and duplicate rules, which ensured a reliable business operation.

Testing was a key focus of the project. Through UI validation, automation functionality checks, end-to-end business flow verification, and dashboard refresh validation, the Tours & Travels CRM App was proven to be consistent and reliable. Evidence from the test cases showed that the system not only worked correctly but also mirrored real-world workflows, such as booking management, payment tracking, customer feedback collection, and employee assignment.

Practices for maintenance and monitoring were also considered, including debug log reviews, scheduled job monitoring, and cross-browser testing, ensuring that the Tours & Travels CRM could be sustained and scaled in a real production environment.

The system successfully simulated a travel agency ecosystem — connecting customers, employees, bookings, packages, payments, feedback, and tasks into one cohesive workflow. The result was a CRM that is not only functional, but also extensible, adaptable, and aligned with the daily operations of a tourism business.

What I learned

The capstone project for Tours & Travels gave me the opportunity to experience building an enterprise-grade Salesforce application from the ground up.

I learned how to:

- Design data models that can capture real-world relationships between customers, bookings, packages, and payments.
- Use declarative tools (Flows, Process Builder, Validation Rules, Dashboards) to automate processes without code.
- Apply programmatic features like Apex triggers and asynchronous jobs to extend functionality.
- Debug and troubleshoot issues using Salesforce's debug logs, test classes, and error handling tools.
- Ensure data accuracy, system reliability, and user accessibility through rigorous testing and security configurations.
- Align technical design with real-world travel agency operations, reducing manual effort and ensuring smooth workflows.

Overall, the Tours & Travels CRM capstone project reflects not only technical knowledge of Salesforce but also the ability to think like a business stakeholder, focusing on efficiency, usability, and scalability. It stands as both a professional portfolio piece and a learning journey into building end-to-end CRM solutions.

Future Enhancements

While the current Tours & Travels CRM successfully automates the processes for Bookings, Payments, and Customer management, there are opportunities for further improvements. Possible enhancements include:

- Integration with Third Party-APIs: Connecting with popular payment gateways (e.g. Maya, Gcash) and travel platforms (e.g. Amadeus) for real-time package availability and seamless transactions.
- Mobile application support: Building a Salesforce Mobile Application extension so customers and agents can manage bookings and payments directly from their mobile devices.
- AI-powered Recommendations: Leveraging Salesforce Einstein to suggest a personalised travel package to customers based on past bookings, preferences, and seasonal trends.
- Multi-Language & Multi-Currency support: Enhancing the CRM's global usability by supporting multiple languages and currencies for international customers.
- Loyalty & Rewards Program: Implement a pointing-based reward system where frequent travelers can earn discounted price or special offers.
- Social Media Integration: Connecting the CRM with popular social media platforms (e.g. Facebook, Instagram) to capture leads from ads and social media inquiries directly into Salesforce.
- Sustainability Tracking: Allowing agencies to showcase eco-friendly travel packages and track carbon footprints for environmentally conscious travelers.

End