

Tours and Travel Salesforce Documentation

The Tours & Travels CRM project was initiated to address critical operational challenges in the travel industry, including inconsistent customer experiences, inefficient booking processes, manual feedback tracking, and a lack of centralized employee coordination. Users, such as customers, agents, guides, and finance officers, required a system that streamlined communication, automated workflows, and provided role-specific access and reporting. Specific needs addressed included real-time booking updates, automated follow-ups, dynamic pricing, and secure data management.

Defining Project Scope and Objectives

Scope:

- Develop a multi-role CRM for global travel agencies.
- Support full booking lifecycle: inquiry to feedback.
- Include object-level data design, workflows, validations, and UI elements.
- Implement security with Profiles, Roles, Sharing Rules, and FLS.

Objectives:

- Enhance operational efficiency by automating workflows.
- Improve customer satisfaction through faster communication.
- Enable real-time monitoring with analytics and dashboards.
- Ensure data accuracy and security through validation and controlled access.

Designing the Data Model and Security Model

Data Model Highlights:

- Custom objects: Customer_info_c, Booking_c, TravelPackage_c, Employee_c, Feedback_c, BookingPayment_c, BookingGuest_c
- Relationships: Lookup, Master-Detail, and formula fields to link entities logically.
- Roll-up summaries to calculate guest count.

Security Model:

Profiles: Admin, Travel Agent, Finance Officer, Tour Guide, Customer Service.

- Admin Profile has full access to all features.
- Travel Agent Profile can create and manage bookings.
- Finance Officer Profile manages payment-related data.
- Tour Guide Profile has read-only access to travel assignments.
- Customer Service Profile can view and manage customer feedback.

Role Hierarchy: CEO > Manager > Agent > Guide.

- The CEO has access to everything.
- Managers can see data from the agents and guides they supervise.
- Agents handle customer bookings directly.
- Guides only see what's related to their assignments.

Permission Sets: These provide additional access when needed without modifying the user's main profile. For instance, a Travel Agent Manager might get extra reporting permissions via a permission set.

Sharing Rules: These automatically share specific records. For example, customer details are auto-shared with guides assigned to those customers, but only for viewing, not editing.

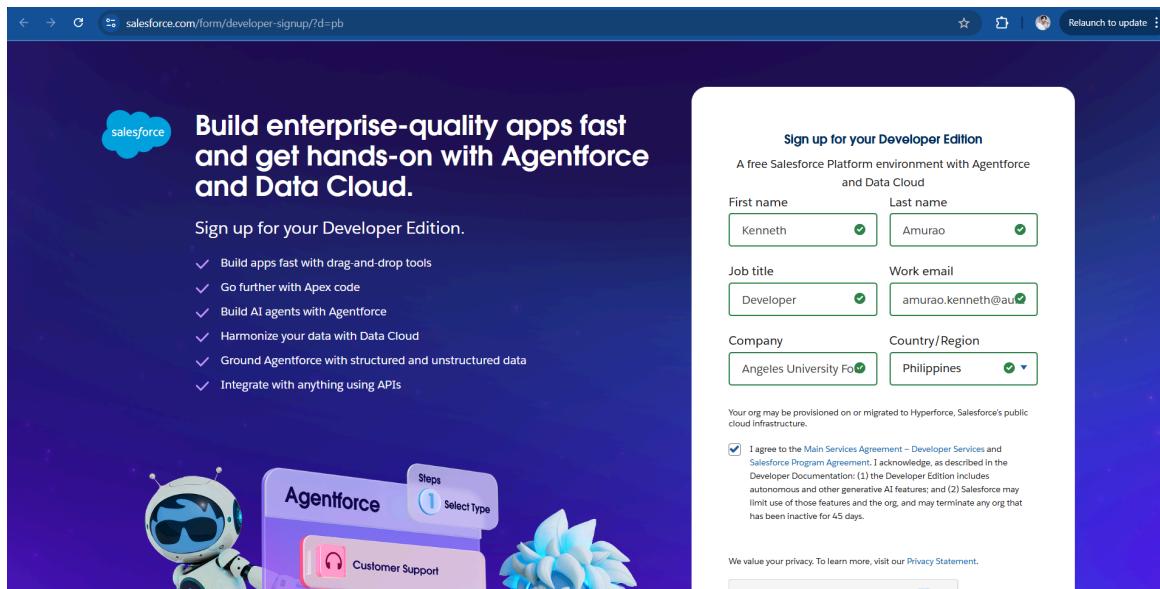
Project Roadmap and Milestones

- **Milestone 1–6:** Object, Field, and UI Setup
- **Milestone 7–10:** Automation (Approval Process, Flows, Workflow, Process Builder)
 - **Note:** Before starting Milestone 7, start the following pre-requisite Milestone activity:
 - Start with **Milestone 13 (Creating Lightning App "Tours & Travels CRM")**
 - Then proceed to **Milestone 24 (Roles & Role Hierarchy)** before Profile to prevent conflict for further errors,
 - After that, I now proceed to **Milestone 23 (Profile)**
 - Next to profile **Milestone 16 (User)**
 - Once all is complete, that's the time I start **Milestone 7**
 -
- **Milestone 11–12:** Apex Development (Trigger, Future, Queueable, Batch Apex)
- **Milestone 13–20:** App Manager, UI Enhancement, LWC, Dynamic Forms
- **Milestone 21–26:** Field History, Matching Rules, Profiles, Roles, Permissions
- **Milestone 27–29:** Test Classes, Test Cases, Data Import

Phase 2: Salesforce Development - Backend & Configurations

Setup Environment & DevOps Workflow

- Developer Org was created via Salesforce Developer Signup.
- DevOps followed iterative testing and validations using Developer Console and Change Sets for deployment

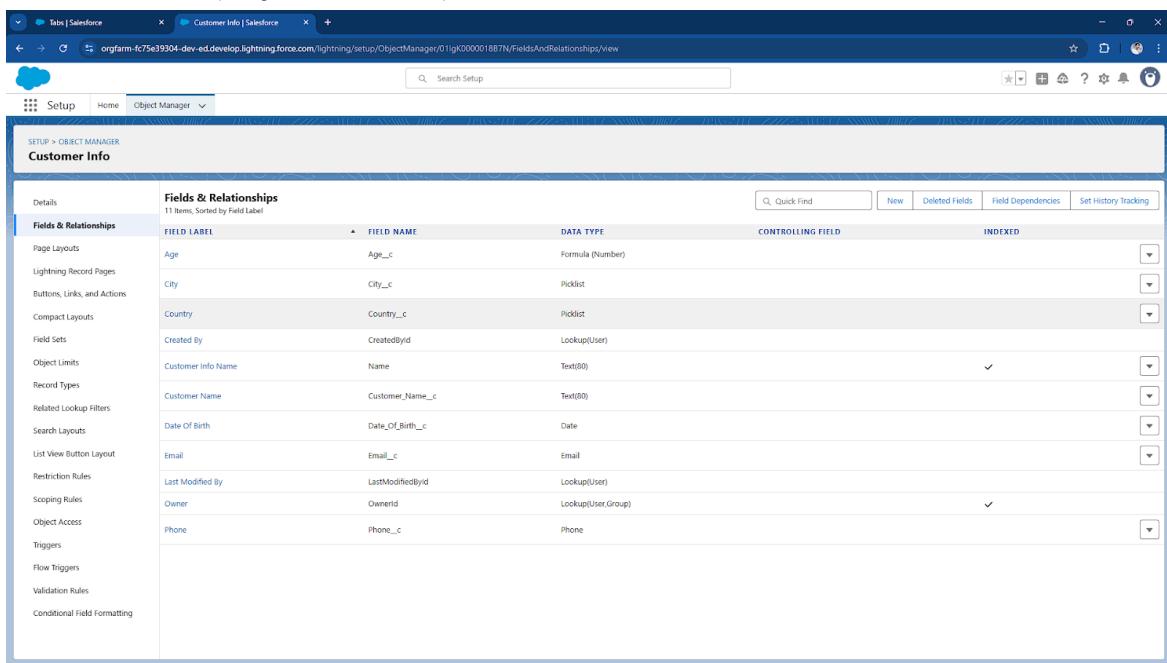


A screenshot of the Salesforce Setup Home page. The URL is orgfarm-fc75e39304-dev-ed.develop.lightning.force.com/lightning/setup/SetupOneHome/home. The page has a sidebar with links like Setup Home, Salesforce Go, Service Setup Assistant, etc. The main content area shows three cards: "Data Cloud" (Watch Video, Let's Go), "Get Started with Einstein Bots" (Get Started), and "Mobile Publish" (Learn More). On the right, there's a "Setup Menu" dropdown with options like Setup, Data Cloud Setup, Service Setup, and Salesforce Go. Below the menu, there's a "Most Recently Used" section with a table showing items like Customer Info Layout, Kenneth Amurao, and SFDC_DevConsole. The table has columns for NAME, TYPE, and OBJECT.

Customizations Implemented

- **Seven Custom Objects & Fields:** Customer__c, Booking__c, BookingGuest__c, TravelPackage__c, BookingPayment__c, Employee__c, and Feedback__c. Over 60 fields were configured across these objects, including lookups, picklists, formula fields, and roll-up summaries.

Customer info (Object & Fields)



The screenshot shows the Salesforce Object Manager interface. The top navigation bar includes tabs for 'Tab | Salesforce' and 'Customer Info | Salesforce'. The URL is 'orgfarm-fc75e9304-dev-ed.lightning.force.com/lightning/setup/ObjectManager/C1gK00001887N/FieldsAndRelationships/view'. The main content area is titled 'Customer Info' under 'SETUP > OBJECT MANAGER'. On the left, a sidebar lists various configuration categories: 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 'Fields & Relationships' section is currently selected and displays a table with 11 items, sorted by Field Label. The table 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 Info Name	Name	Text(60)		✓
Customer Name	Customer_Name__c	Text(60)		
Date Of Birth	Date_of_Birth__c	Date		
Email	Email__c	Email		
Last Modified By	LastModifiedById	Lookup(User)		
Owner	OwnerId	Lookup(User/Group)		✓
Phone	Phone__c	Phone		

BookingGuest (Object & Fields)

The screenshot shows the Salesforce Object Manager interface for the 'BookingGuest' object. The left sidebar lists various setup options like Details, Page Layouts, Lightning Record Pages, etc. The main area is titled 'Fields & Relationships' and displays 14 items, sorted by Field Label. The table columns are FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Age	Age_c	Number(18, 0)		
Age Category	Age_Category__c	Formula (Text)		
BookingGuest Name	BookingGuest_Name__c	Text(80)		
BookingGuest Name	Name	Text(80)		✓
City	City__c	Picklist		
Country	Country__c	Picklist		
Created By	CreatedById	Lookup(User)		
Gender	Gender__c	Picklist		
Last Modified By	LastModifiedById	Lookup(User)		

This screenshot is identical to the one above, showing the 'Fields & Relationships' section for the 'BookingGuest' object. It displays the same 14 items and table structure as the first screenshot.

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Age	Age_c	Number(18, 0)		
Age Category	Age_Category__c	Formula (Text)		
BookingGuest Name	BookingGuest_Name__c	Text(80)		
BookingGuest Name	Name	Text(80)		✓
City	City__c	Picklist		
Country	Country__c	Picklist		
Created By	CreatedById	Lookup(User)		
Gender	Gender__c	Picklist		
Last Modified By	LastModifiedById	Lookup(User)		

Booking (Object & Fields)

Screenshot of the Salesforce Object Manager interface for the Booking object.

Fields & Relationships (30 items, Sorted by Field Label)

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Accommodation Amount per Person per Day	Accommodation_Amount_per_Person_per_Day__c	Formula (Currency)		
Approval Status	Approval_Status__c	Picklist		
Booking Date	Booking_Date__c	Date		
Booking Name	Name	Auto Number		
Booking Number	Booking_Number__c	Auto Number		
Booking Status	Booking_Status__c	Picklist		
Cancel Confirmation	Cancel_Confirmation__c	Checkbox		
Cancellation Date	Cancellation_Date__c	Date		
Cancellation Reason	Cancellation_Reason__c	Text Area(255)		
Created By	CreatedById	Lookup(User)		
Customer	Customer__c	Lookup(Customer)		
Customer Email	Customer_Email__c	Email		
Include Travel Insurance	Include_Travel_Insurance__c	Checkbox		
Last Modified By	LastModifiedById	Lookup(User)		
Membership Chosen Reg	Membership_Chosen_Reg__c	Picklist		

Screenshot of the Salesforce Object Manager interface for the Booking object.

Fields & Relationships (30 items, Sorted by Field Label)

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
No of Booking Guests Info Available	No_of_Booking_Guests_Info_Available__c	Roll-Up Summary (COUNT BookingGuest)		
Number of Travelers	Number_of_Travelers__c	Number(18, 0)		
Owner	OwnerId	Lookup(User/Group)		
Preferred Accommodation	Preferred_Accommodation__c	Picklist		
Preferred Guide Language	Preferred_Guide_Language__c	Picklist		
Require Tour Guide	Require_Tour_Guide__c	Checkbox		
Require Visa Assistance	Require_Visa_Assistance__c	Checkbox		
Total Accommodation Amount	Total_Accommodation_Amount__c	Formula (Currency)		
Total Billing Amount	Total_Billing_Amount__c	Formula (Currency)		
Total Travel Amount	Total_Travel_Amount__c	Formula (Currency)		
Travel Cost Per Person	Travel_Cost_Per_Person__c	Formula (Currency)		
Travelling End Date	Travelling_End_Date__c	Formula (Date)		
Travelling Start Date	Travelling_Start_Date__c	Date		
TravelPackage	TravelPackage__c	Lookup(TravelPackage)		
Trip Type	Trip_Type__c	Picklist		

Travel Package Object (Object & Fields)

Screenshot of the Salesforce Object Manager showing the fields and relationships for the TravelPackage object.

Fields & Relationships (23 items, Sorted by Field Label)

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Additional Notes	Additional_Notes_c	Text Area(255)		
Availability Status	Availability_Status_c	Picklist		
Average Rating	Average_Rating_c	Number(18, 0)		
Country	Country_c	Picklist		
Created By	CreatedById	Lookup(User)		
Duration in Days	Duration_in_Days_c	Number(18, 0)		
Guide Included	Guide_Included_c	Checkbox		
Guide LanguagesPlaces Covered	Guide_LanguagesPlaces_Covered_c	Picklist (Multi-Select)		
Insurance Included	Insurance_Included_c	Checkbox		
Last Modified By	LastModifiedById	Lookup(User)		
Maximum Group Size	Maximum_Group_Size_c	Number(18, 0)		
Meals Included	Meals_Included_c	Picklist		
Membership	Membership_c	Picklist (Multi-Select)		
Owner	OwnerId	Lookup(User/Group)		
Package Type	Package_Type_c	Picklist (Multi-Select)		

Screenshot of the Salesforce Object Manager showing the fields and relationships for the TravelPackage object.

Fields & Relationships (23 items, Sorted by Field Label)

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Insurance Included	Insurance_Included_c	Checkbox		
Last Modified By	LastModifiedById	Lookup(User)		
Maximum Group Size	Maximum_Group_Size_c	Number(18, 0)		
Meals Included	Meals_Included_c	Picklist		
Membership	Membership_c	Picklist (Multi-Select)		
Owner	OwnerId	Lookup(User/Group)		
Package Type	Package_Type_c	Picklist (Multi-Select)		
Places Covered	Places_Covered_c	Text Area(255)		
Preferred Guide Language	Preferred_Guide_Language_c	Picklist		
Price Per Person	Price_Per_Person_c	Currency(18, 0)		
Region	Region_c	Picklist		
Transportation Modes	Transportation_Modes_c	Picklist (Multi-Select)		
Travel Package Name	Travel_Package_Name_c	Text(80)		
TravelPackage Name	Name	Text(80)		
Visa Assistance	Visa_Assistance_c	Checkbox		

Booking Payment (Object & Fields)

The screenshot shows the Salesforce Object Manager interface for the 'Booking Payment' object. The left sidebar lists various setup categories like Page Layouts, Lightning Record Pages, etc. The main content area displays a table of fields and relationships:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Booking	Booking__c	Lookup(Booking)		✓
Booking Payment Name		Name	Auto Number	✓
Created By	CreatedById	Lookup(User)		✓
Customer Name	Customer_Name__c	Formula (Text)		✓
Last Modified By	LastModifiedById	Lookup(User)		✓
Notes	Notes__c	Text Area(255)		✓
Owner	OwnerId	Lookup(User,Group)		✓
Payment Date	Payment_Date__c	Date		✓
Payment Method	Payment_Method__c	Picklist		✓
Payment Mode Details	Payment_Mode_Details__c	Text Area(255)		✓
Payment Number	Payment_Number__c	Auto Number		✓
Payment Receipt Sent	Payment_Receipt_Sent__c	Checkbox		✓
Payment Reference Number	Payment_Reference_Number__c	Text(50)		✓
Payment Status	Payment_Status__c	Picklist		✓
Total Bill Amount	Total_Bill_Amount__c	Formula (Currency)		✓

Feedback Object

The screenshot shows the Salesforce Object Manager interface for the 'Feedback' object. The left sidebar lists various setup categories like Page Layouts, Lightning Record Pages, etc. The main content area displays a table of fields and relationships:

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Accommodation Rating	Accommodation_Rating__c	Picklist		✓
Booking	Booking__c	Lookup(Booking)		✓
Created By	CreatedById	Lookup(User)		✓
Customer	Customer__c	Lookup(Customer)		✓
Feedback Date	Feedback_Date__c	Date		✓
Feedback Name		Name	Auto Number	✓
Feedback No	Feedback_No__c	Auto Number		✓
Last Modified By	LastModifiedById	Lookup(User)		✓
Overall Rating	Overall_Rating__c	Picklist		✓
Owner	OwnerId	Lookup(User,Group)		✓
Package Satisfaction	Package_Satisfaction__c	Picklist		✓
Response Status	Response_Status__c	Picklist		✓
Suggestions / Comments	Suggestions_Comments__c	Long Text Area(32768)		✓
Tour Guide Rating	Tour_Guide_Rating__c	Picklist		✓
Transport Rating	Transport_Rating__c	Picklist		✓

- **Validation Rules:** Ensured data accuracy (e.g., 10-digit phone, email regex).
 - Phone field must be 10 digits.(Customer Info, BookingGuest)

The screenshot shows the 'Customer Info Validation Rule' detail page in the Salesforce Setup interface. The left sidebar lists various object settings like Details, Fields & Relationships, Page Layouts, etc. The main content area is titled 'Customer Info Validation Rule' and contains a 'Validation Rule Detail' section. The rule is named 'Phone_Number_Must_10_Digits'. The 'Error Condition Formula' is set to 'LEN(Phone__c) <> 10'. The 'Error Message' is 'Phone Number Must Be 10 Digits'. The 'Active' checkbox is checked. The 'Error Location' is set to 'Phone'. The 'Created By' field shows 'Kenneth Amurao' with a timestamp of '7/9/2025, 7:08 AM'. The 'Modified By' field also shows 'Kenneth Amurao' with the same timestamp.

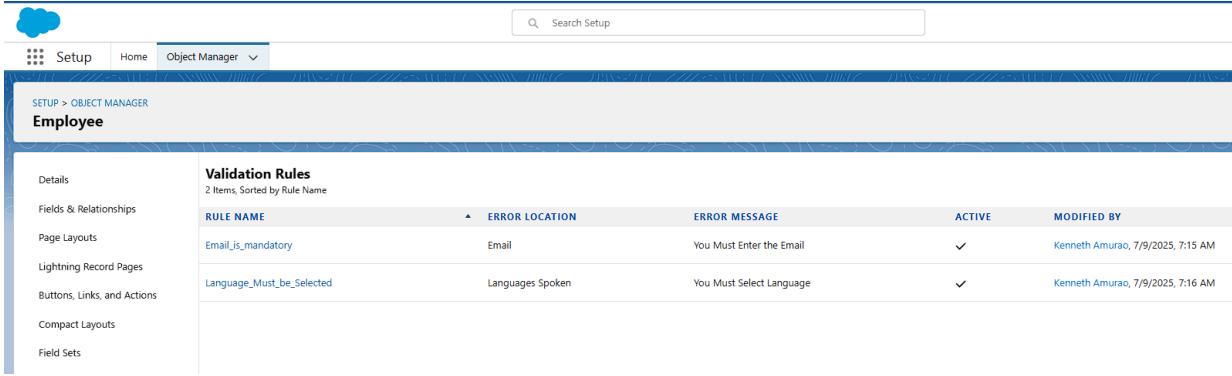
- The email must match the standard email format.

This screenshot shows another instance of the 'Customer Info Validation Rule' detail page. The rule is named 'Email_Valid_Address'. The 'Error Condition Formula' is 'NOT(REGEX>Email__c, \"^([a-zA-Z0-9_.]+@[a-zA-Z0-9.][2,]\$)\")'. The 'Error Message' is 'Please Enter Valid Email Address'. The 'Active' checkbox is checked. The 'Error Location' is set to 'Email'. The 'Created By' field shows 'Kenneth Amurao' with a timestamp of '7/9/2025, 7:10 AM'. The 'Modified By' field also shows 'Kenneth Amurao' with the same timestamp.

- Date of Birth must not be in the future. (Customer Info)

The final screenshot shows the 'Customer Info Validation Rule' detail page again. This rule is named 'Prevent_Future_DOB'. The 'Error Condition Formula' is 'DateOfBirth__c > TODAY()'. The 'Error Message' is 'Date of Birth cannot be in the future'. The 'Active' checkbox is checked. The 'Error Location' is set to 'Date Of Birth'. The 'Created By' field shows 'Kenneth Amurao' with a timestamp of '7/9/2025, 7:11 AM'. The 'Modified By' field also shows 'Kenneth Amurao' with the same timestamp.

- Mandatory Email and Language Spoken(Employee)

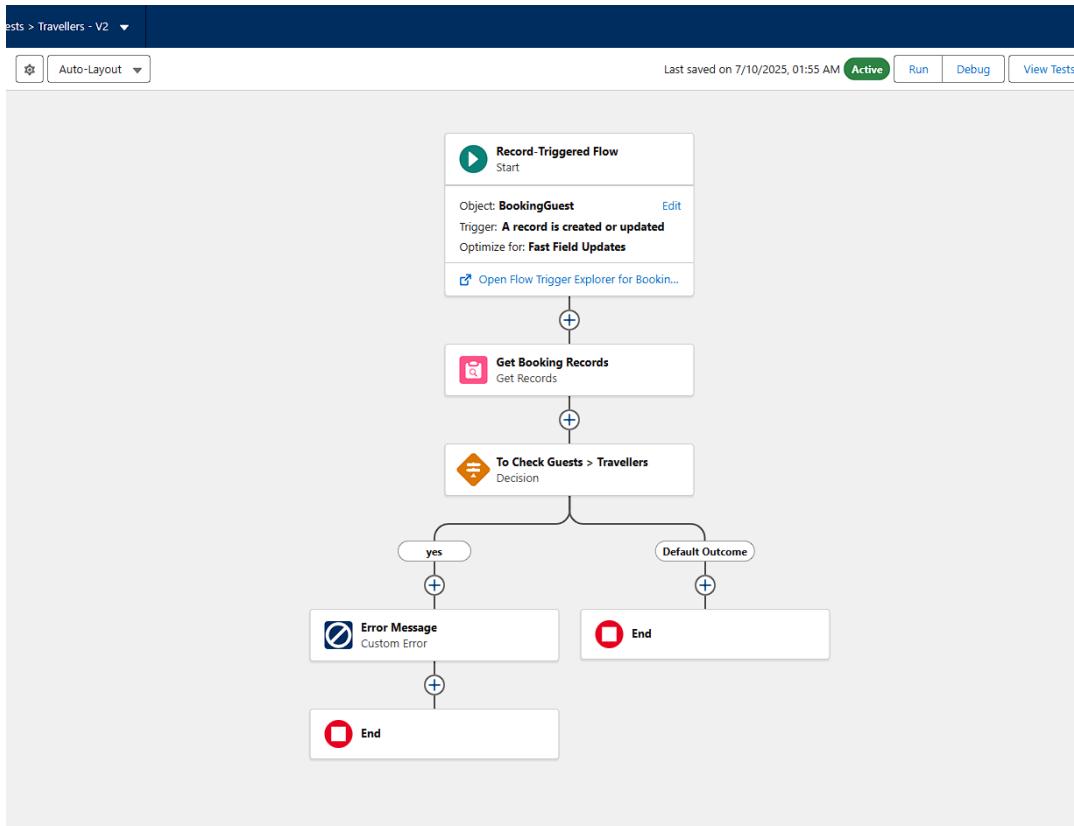


The screenshot shows the Salesforce Object Manager interface for the 'Employee' object. The left sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, and Field Sets. The main content area is titled 'Validation Rules' and shows two items: 'Email_is_mandatory' (Email location, message: 'You Must Enter the Email') and 'Language_Must_be_Selected' (Languages Spoken location, message: 'You Must Select Language'). Both rules are active and were modified by Kenneth Amurao on 7/9/2025.

Validation Rules				
RULE NAME	ERROR LOCATION	ERROR MESSAGE	ACTIVE	MODIFIED BY
Email_is_mandatory	Email	You Must Enter the Email	✓	Kenneth Amurao, 7/9/2025, 7:15 AM
Language_Must_be_Selected	Languages Spoken	You Must Select Language	✓	Kenneth Amurao, 7/9/2025, 7:16 AM

● Automation Tools:

- Flows: A flow was built to prevent adding more guests than travelers in a booking. Another flow handled error messages for violations.



- **Workflow Rule:** Automatically assigned a follow-up Task to the Travel Agent three days after the trip's end.

The screenshot shows the 'Workflow Rule Detail' page for a rule named 'Follow-up Task After Booking Completion'. The rule is active and evaluates the 'Booking' object when a record is created or updated. It was created by Kenneth Amurao on 7/9/2025 at 10:59 AM and modified by him on 7/9/2025 at 11:02 AM. The 'Workflow Actions' section is empty.

- **Process Builder:** Automatically updated Booking Status to "Confirmed" when payment status became "Completed".

The screenshot shows a Process Builder screen with two main sections: a process flow diagram and a configuration panel.

Process Flow Diagram:

```

graph TD
    START([START]) --> Payment[Booking Payment]
    Payment --> Decision{Payment Completed}
    Decision -- TRUE --> Actions1[IMMEDIATE ACTIONS]
    Actions1 --> STOP1([STOP])
    Decision -- FALSE --> STOP2([STOP])
  
```

Configuration Panel:

- Define Criteria for this Action Group:**
 - Conditions are met: Formula evaluates to true. No criteria—just execute the actions!
- Set Conditions:**

Field *	Operator *	Type *	Value *
[Booking_Payme...]	Equals	Picklist	Completed
- Conditions ***
 - All of the conditions are met (AND)
 - Any of the conditions are met (OR)
 - Customize the logic
- Action Name ***: Update Booking Status
- Record ***: [Booking_Payment_c].Booking
- Criteria for Updating Records ***
 - Updated records meet all conditions
 - No criteria—just update the records!
- Set new field values for the records you update**

Field *	Type *	Value *
Booking Status	Picklist	Confirmed

- **Approval Process:** Canceled bookings required manager approval, triggering approval emails and status updates based on the outcome.

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 Submit
Description	Process to approve booking cancellations		
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, Book Manager		
Created By	Kenneth Amuroo, 7/9/2025, 9:59 AM	Modified By	Kenneth Amuroo, 7/9/2025,

Initial Submission Actions [1] [Add Existing](#) [Add New ▾](#)

Action	Type	Description
Record Lock		Lock the record from being edited
Edit Remove	Field Update	Booking status

Approval Steps [1]

Action	Step Number	Name	Description
Show Actions	1	Travel Agent Manager Approval	

Final Approval Actions [1] [Add Existing](#) [Add New ▾](#)

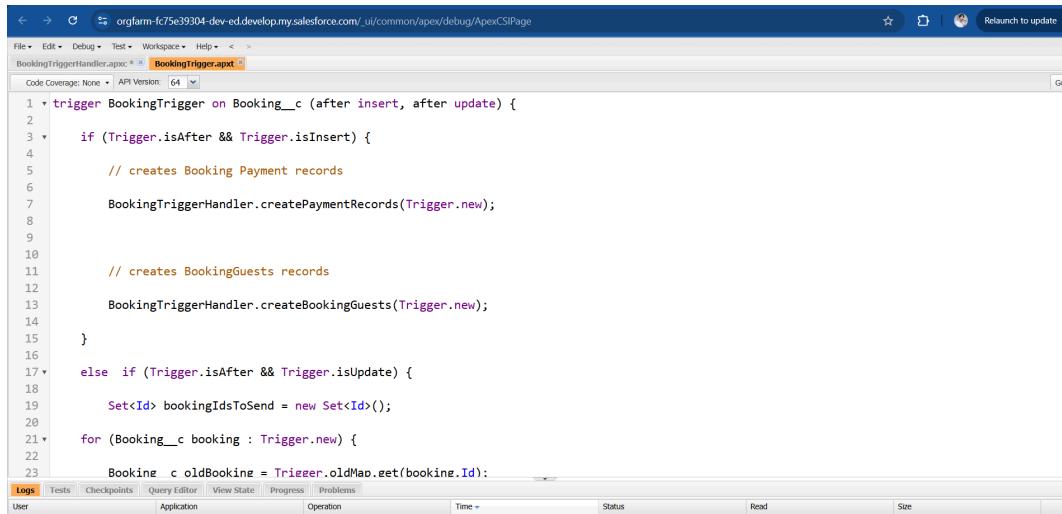
Action	Type	Description
Edit	Record Lock	Lock the record from being edited
Edit Remove	Field Update	Approval Status Update
Edit Remove	Email Alert	Send Email to Booking Customer

Final Rejection Actions [1] [Add Existing](#) [Add New ▾](#)

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

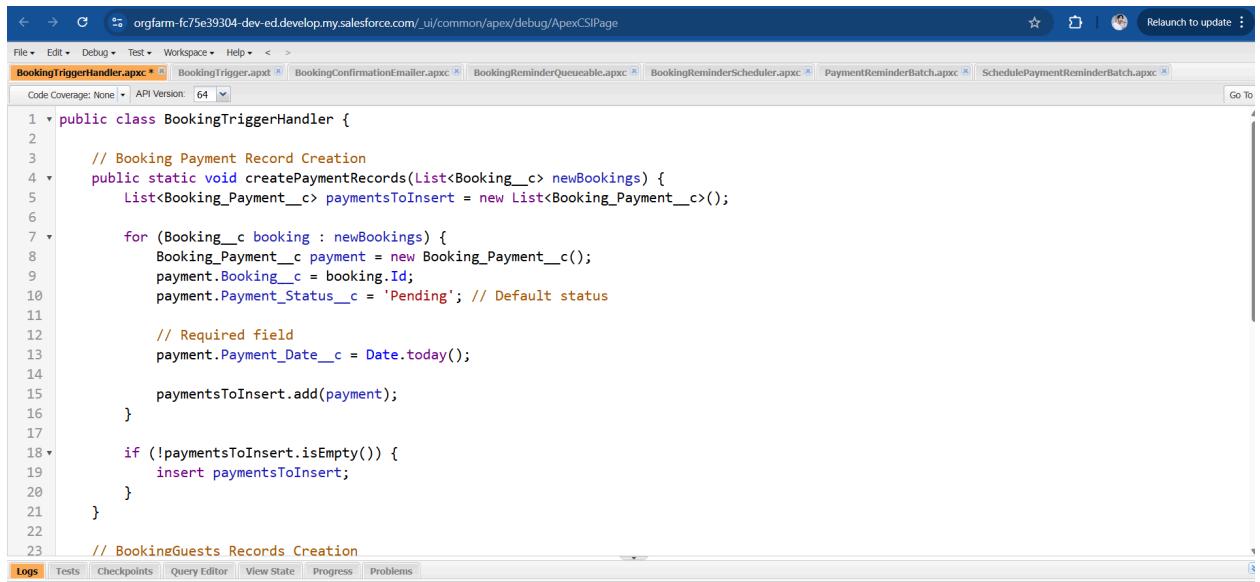
Apex Development

- **Trigger:** A trigger on the Booking__c object created BookingPayment__c and BookingGuest__c records automatically when a booking was inserted.



```
trigger BookingTrigger on Booking__c (after insert, after update) {
    if (Trigger.isAfter && Trigger.isInsert) {
        // creates Booking Payment records
        BookingTriggerHandler.createPaymentRecords(Trigger.new);
    }
    else if (Trigger.isAfter && Trigger.isUpdate) {
        Set<Id> bookingIdsToSend = new Set<Id>();
        for (Booking__c booking : Trigger.new) {
            Booking_c oldBooking = Trigger.oldMap.get(booking.Id);
            Booking_c newBooking = Trigger.newMap.get(booking.Id);
            if (oldBooking != null && oldBooking.Status__c != newBooking.Status__c) {
                bookingIdsToSend.add(booking.Id);
            }
        }
        if (!bookingIdsToSend.isEmpty()) {
            BookingTriggerHandler.createBookingGuests(bookingIdsToSend);
        }
    }
}
```

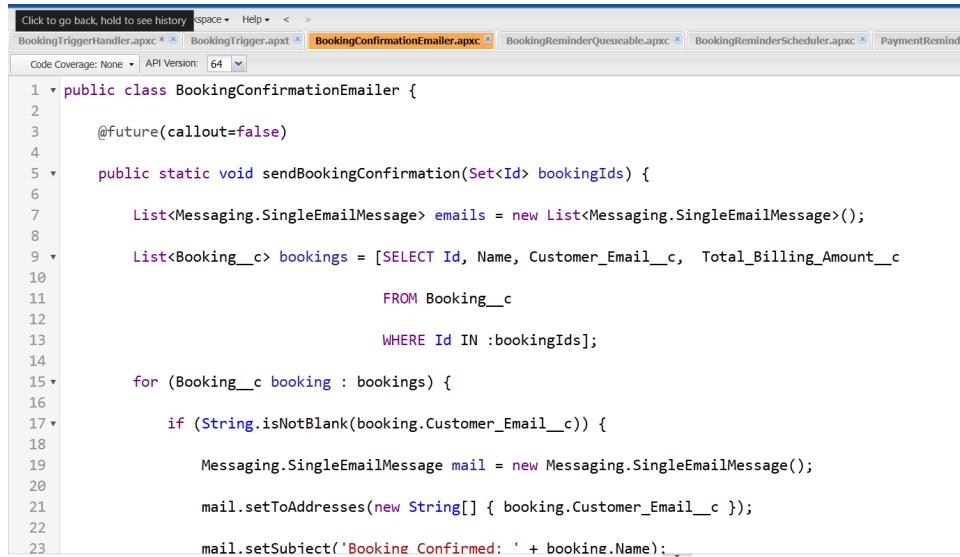
- **Apex Class:** The logic was abstracted into BookingTriggerHandler class for modularity and reusability.



```
public class BookingTriggerHandler {
    // Booking Payment Record Creation
    public static void createPaymentRecords(List<Booking__c> newBookings) {
        List<Booking_Payment__c> paymentsToInsert = new List<Booking_Payment__c>();
        for (Booking__c booking : newBookings) {
            Booking_Payment__c payment = new Booking_Payment__c();
            payment.Booking__c = booking.Id;
            payment.Payment_Status__c = 'Pending'; // Default status
            payment.Payment_Date__c = Date.today();
            paymentsToInsert.add(payment);
        }
        if (!paymentsToInsert.isEmpty()) {
            insert paymentsToInsert;
        }
    }
    // BookingGuests Records Creation
}
```

- **Asynchronous Apex:**

- **Future Method:** This method was used to send booking confirmation emails after the booking status was updated to "Confirmed."

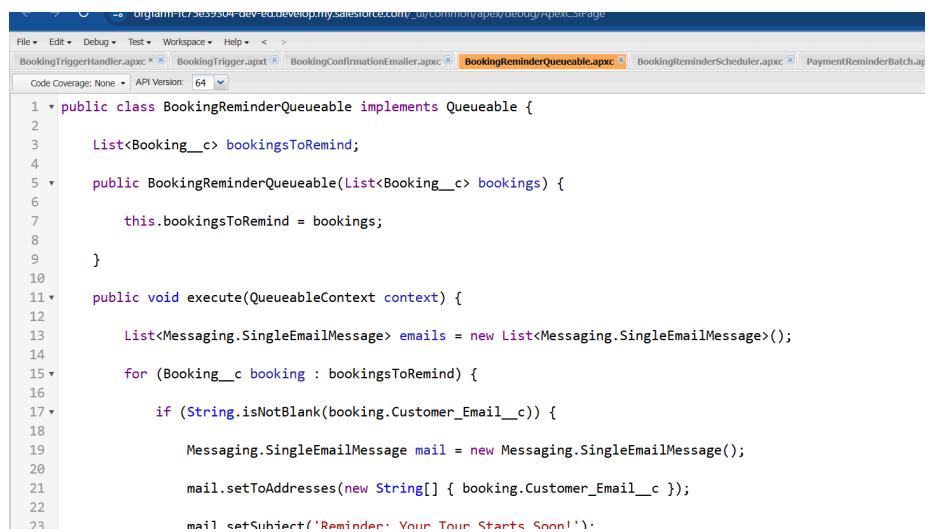


```

1 public class BookingConfirmationEmailer {
2
3     @future(callout=false)
4
5     public static void sendBookingConfirmation(Set<Id> bookingIds) {
6
7         List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
8
9         List<Booking__c> bookings = [SELECT Id, Name, Customer_Email__c, Total_Billing_Amount__c
10
11             FROM Booking__c
12
13             WHERE Id IN :bookingIds];
14
15         for (Booking__c booking : bookings) {
16
17             if (String.isNotBlank(booking.Customer_Email__c)) {
18
19                 Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
20
21                 mail.setToAddresses(new String[] { booking.Customer_Email__c });
22
23                 mail.setSubject('Booking Confirmed: ' + booking.Name);
24
25             }
26
27         }
28
29     }
30
31 }

```

- **Queueable Class:** A queueable class was created to handle the logic of sending personalized reminder emails to customers exactly 3 days before their tour starts.

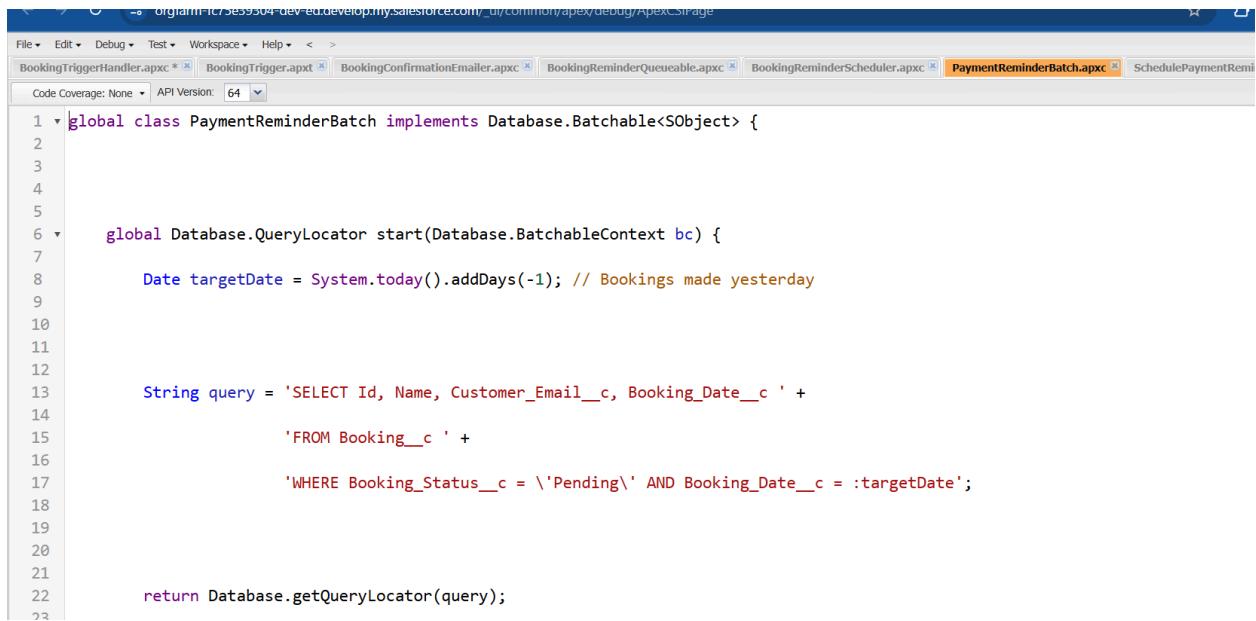


```

1 public class BookingReminderQueueable implements Queueable {
2
3     List<Booking__c> bookingsToRemind;
4
5     public BookingReminderQueueable(List<Booking__c> bookings) {
6
7         this.bookingsToRemind = bookings;
8
9     }
10
11     public void execute(QueueableContext context) {
12
13         List<Messaging.SingleEmailMessage> emails = new List<Messaging.SingleEmailMessage>();
14
15         for (Booking__c booking : bookingsToRemind) {
16
17             if (String.isNotBlank(booking.Customer_Email__c)) {
18
19                 Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
20
21                 mail.setToAddresses(new String[] { booking.Customer_Email__c });
22
23                 mail.setSubject('Reminder: Your Tour Starts Soon!');
24
25             }
26
27         }
28
29     }
30
31 }

```

- **Batch Class:** Daily payment reminder emails. A batch job was implemented to send payment reminders on a daily basis. It queried bookings with a 'Pending' payment status and emailed the respective customers.



```

1 global class PaymentReminderBatch implements Database.Batchable<SObject> {
2
3
4
5
6     global Database.QueryLocator start(Database.BatchableContext bc) {
7
8         Date targetDate = System.today().addDays(-1); // Bookings made yesterday
9
10
11
12         String query = 'SELECT Id, Name, Customer_Email__c, Booking_Date__c '
13             'FROM Booking__c '
14             'WHERE Booking_Status__c = \'Pending\' AND Booking_Date__c = :targetDate';
15
16
17         return Database.getQueryLocator(query);
18
19
20
21
22
23

```

- **Schedulable Apex:** Scheduled jobs for automation.



```

1 public class SchedulePaymentReminderBatch implements Schedulable {
2
3     public void execute(SchedulableContext sc) {
4
5         PaymentReminderBatch batch = new PaymentReminderBatch();
6
7         Database.executeBatch(batch, 200);
8
9     }
10
11 }

```

Phase 3: UI/UX Development & Customization

Lightning App Setup

A custom app named "**Tours & Travels CRM**" was built using the **App Manager** in Salesforce Lightning. This app centralized access to all CRM components such as Customer Info, Bookings, Payments, Employees, and more.

- Tabs were organized for ease of navigation: Customer, Booking, TravelPackage, Employee, Feedback, Payment, Task, Reports, and Dashboard.

The screenshot shows the Salesforce Lightning interface for the 'Tours & Travels CRM' app. The top navigation bar is highlighted with a red box and contains tabs for various CRM components. On the left, there's a sidebar with a 'Recently Viewed' section showing a list of customer names. An arrow points from this sidebar to the 'Customers Info' tab in the navigation bar.

UI Elements

Page Layouts

Customized layouts were created for each object to improve data visibility and user productivity. Key fields were rearranged logically to streamline user data entry and reduce cognitive load. (Customer Info, BookingGuest, TravelGuest, Employee, Booking, Booking Payment, Feedback)

Customer Info

The screenshot shows the 'Customer Info' object in the 'Object Manager' under 'Setup'. The 'Page Layouts' tab is selected in the sidebar. The main area displays the layout configuration for the 'Customer Info' object, showing fields like Name, Email, Phone, and Date of Birth. The layout is designed to improve data visibility and user productivity.

BookingGuest

SETUP > OBJECT MANAGER
BookingGuest

Fields

Field Name	Type	Description
Booking	Section	Booking
Booking Guest Name	Text	Country
Created By	Text	Passport Number
Age	Text	Relation with Cus...
Booking Guest Name	Text	Gender
Age Category	Text	Special Needs
City	Text	Last Modified By
Visa Required	Text	

BookingGuest Detail

Information (Header visible on edit only)

Field	Type	Value
Booking	Text	Sample Text
Booking Guest Name	Text	Sample Text
Age	Text	29.646
Gender	Text	Sample Text
Relation with Customer	Text	Sample Text
Age Category	Text	Sample Text
Special Needs	Text	Sample Text

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

TravelGuest

SETUP > OBJECT MANAGER
TravelPackage

Fields

Field Name	Type	Description
Section	Section	Average Rating
Country	Text	Guide Languages
Additional Notes	Text	Meals Included
Created By	Text	Places Covered
Availability Status	Text	Preferred Guide
Duration in Days	Text	Transportation Modes
Package Type	Text	
Transportation Modes	Text	
Availability Status	Text	
Average Rating	Text	
Meals Included	Text	

TravelPackage Detail

Information (Header visible on edit only)

Field	Type	Value
TravelPackage Name	Text	Sample Text
Region	Text	Sample Text
Duration in Days	Text	49.223
Package Type	Text	Sample Text
Transportation Modes	Text	Sample Text
Availability Status	Text	Sample Text
Average Rating	Text	99.957
Meals Included	Text	Sample Text

Buttons: Edit, Delete, Clone, Change Owner, Change Record Type, Printable View, Sharing, Sharing Hierar...

Employee

SETUP > OBJECT MANAGER
Employee

Fields

Field Name	Type	Description
Section	Section	Availability Status
Blank Space	Text	Department
Address	Text	Employee Name
Assigned Region	Text	Last Modified By
Created By	Text	Role

Employee Detail

Information (Header visible on edit only)

Field	Type	Value
Employee ID	Text	GEN-2004-001234
Employee Name	Text	GEN-2004-001234
Employee Name	Text	Sample Text
Email	Text	sarah.sample@company.com
Phone	Text	1.415-555-1212
Role	Text	Sample Text
Department	Text	Sample Text
Employment Type	Text	Sample Text
Salary	Text	\$123.45
Joining Date	Text	7/10/2025
Availability Status	Text	Sample Text
Language Spoken	Text	Sample Text
Assigned Region	Text	Sample Text

Buttons: Save, Quick Save, Preview As..., Cancel, Undo, Redo, Layout Properties, Profile Picture, Sample Text, Owner, Sample Text

Booking

SETUP > OBJECT MANAGER
Booking

Page Layouts

Field	Type	Description	Value
Booking Number	Text	Booking Number	GEN-2004-001234
Customer	Text	Customer Email	sarah.sample@company.com
Travel Package	Text	Travel Package	Sample Text
Booking Date	Date	Booking Date	7/10/2025
Travelling Start Date	Date	Travelling Start Date	7/10/2025
Travelling End Date	Date	Travelling End Date	7/10/2025
Trip Type	Text	Trip Type	Sample Text
Membership Chosen (Req)	Text	Membership Chosen	Sample Text
Preferred Accommodation	Text	Preferred Accommodation	Sample Text
Include Travel Insurance	Text	Include Travel Insurance	✓
Require Visa Assistance	Text	Require Visa Assistance	✓
Number of Travelers	Text	Number of Travelers	60,525
Booking Status	Text	Booking Status	Sample Text
Approval Status	Text	Approval Status	Sample Text
Cancellation Date	Date	Cancellation Date	7/10/2025
Cancellation Reason	Text	Cancellation Reason	✓
Travel Cost Per Person	Text	Travel Cost Per Person	\$123.45
Total Travel Amount	Text	Total Travel Amount	\$123.45
Accommodation Amount per Person per Day	Text	Accommodation Amount per Person per Day	\$123.45
Cancellation Reason	Text	Cancellation Reason	Sample Text
Total Accommodation Amount	Text	Total Accommodation Amount	\$123.45
Total Billing Amount	Text	Total Billing Amount	\$123.45
Owner	Text	Owner	Sample Text

Booking Payment

SETUP > OBJECT MANAGER
Booking Payment

Page Layouts

Field	Type	Description	Value
Created By	Text	Created By	Sample Text
Customer Name	Text	Customer Name	Sample Text
Payment Date	Date	Payment Date	7/10/2025
Payment Method	Text	Payment Method	Payment Reference...
Payment Receipt Sent	Text	Payment Receipt Sent	✓
Payment Status	Text	Payment Status	Sample Text
Notes	Text	Notes	Sample Text

Booking Payment Detail

Field	Type	Description	Value
Payment Number	Text	Payment Number	GEN-2004-001234
Customer Name	Text	Customer Name	Sample Text
Booking	Text	Booking	Sample Text
Payment Date	Date	Payment Date	7/10/2025
Total Bill Amount	Text	Total Bill Amount	\$123.45
Payment Status	Text	Payment Status	Sample Text
Notes	Text	Notes	Sample Text
Payment Method	Text	Payment Method	Sample Text
Payment Reference Number	Text	Payment Reference Number	Sample Text
Payment Mode Details	Text	Payment Mode Details	Sample Text
Payment Receipt Sent	Text	Payment Receipt Sent	✓
Owner	Text	Owner	Sample Text

Feedback

SETUP > OBJECT MANAGER
Feedback

Page Layouts

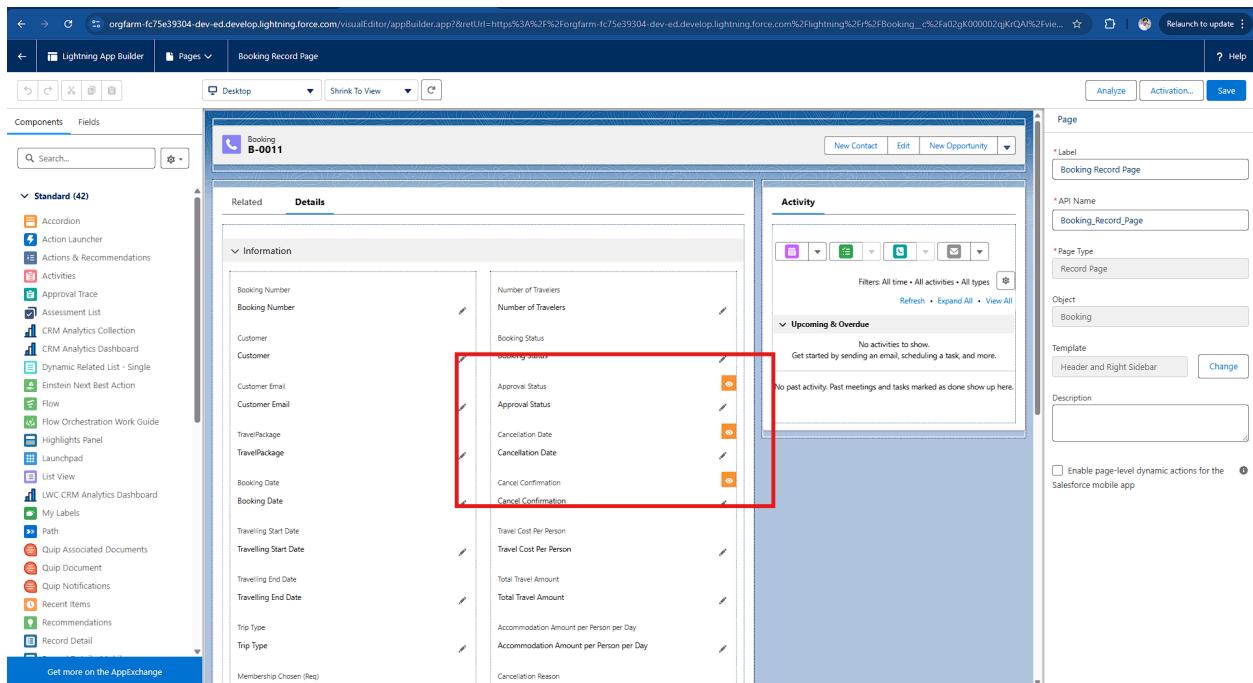
Field	Type	Description	Value
Created By	Text	Created By	Sample Text
Customer	Text	Customer Last Modified By	Response Status
Booking	Text	Booking Last Modified By	Response Status
Feedback Date	Date	Feedback Date	Overall Rating
Feedback Name	Text	Feedback Name	Transport Rating
Feedback No	Text	Feedback No	Sample Text
Customer	Text	Tour Guide Rating	Sample Text
Booking	Text	Transport Rating	Sample Text
Feedback Date	Date	Accommodation Rating	Sample Text
Package Satisfaction	Text	Overall Rating	Sample Text
Response Status	Text	Tour Guide Rating	Sample Text
Suggestions / Comments	Text	Transport Rating	Sample Text
Owner	Text	Accommodation Rating	Sample Text
Owner	Text	Owner	Sample Text

Feedback Detail

Field	Type	Description	Value
Feedback No	Text	Feedback No	GEN-2004-001234
Customer	Text	Customer	Sample Text
Booking	Text	Booking	Sample Text
Feedback Date	Date	Feedback Date	7/10/2025
Package Satisfaction	Text	Package Satisfaction	Sample Text
Response Status	Text	Response Status	Sample Text
Suggestions / Comments	Text	Suggestions / Comments	Sample Text
Overall Rating	Text	Overall Rating	Sample Text
Tour Guide Rating	Text	Tour Guide Rating	Sample Text
Transport Rating	Text	Transport Rating	Sample Text
Accommodation Rating	Text	Accommodation Rating	Sample Text
Owner	Text	Owner	Sample Text

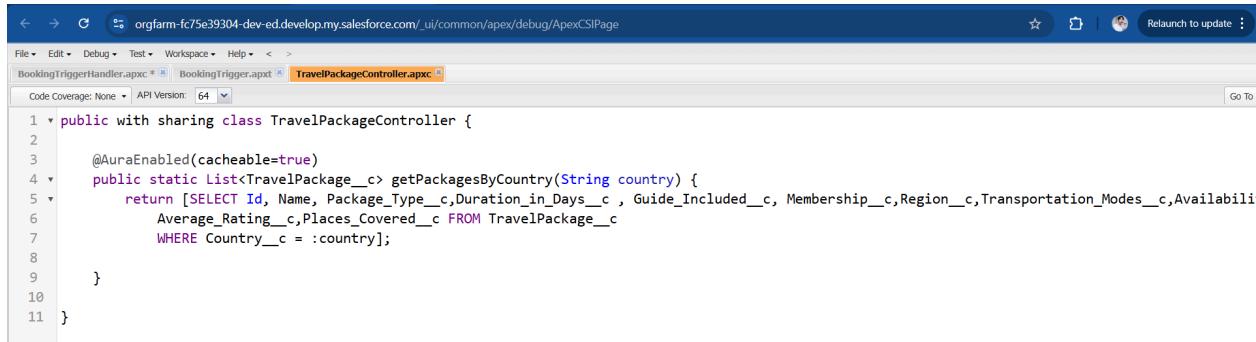
Dynamic Forms

Dynamic Forms were enabled for the Booking object to show or hide fields based on conditions. For instance, cancellation-related fields only appear when the booking status is set to “Cancelled”.

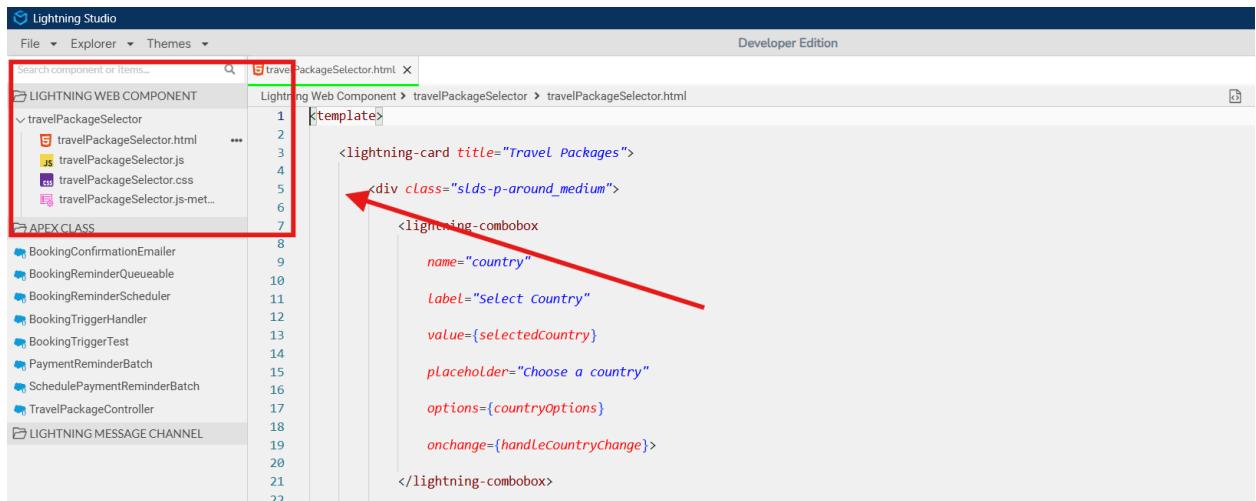


Lightning Pages

Lightning Record Pages were customized to display key information, related lists, charts, and LWCs. This provided users with a single pane view of all essential CRM operations.



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



The screenshot shows the Lightning Studio interface with the search bar set to "travelPackageSelector.html". The left sidebar lists components under "LIGHTNING WEB COMPONENT" and classes under "APEX CLASS". A red box highlights the "travelPackageSelector" component in the sidebar. The main area displays the component's template code:

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

Lightning Web Components (LWC)

Developed a custom LWC named TravelPackageSelector, which filters packages by country dynamically. This component enhances user interactivity by allowing users to select packages without refreshing the page.

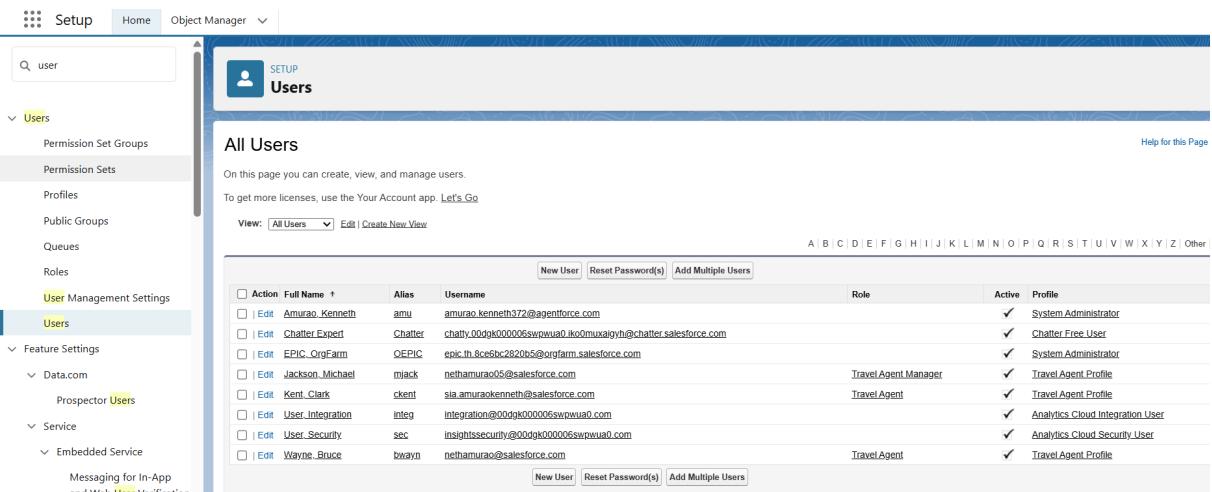
The screenshot shows a Salesforce Lightning Web Component titled "Travel Package Selector". The component displays three travel package options for the Philippines:

- Philippines**
 - Package Type Name: Cebu Tour Package
 - Package Type: Family
 - Duration in Days: 3
 - Guide Facility: false
 - Places Covered: Temple of Leah Magellan's Cross Kawasan Falls
 - Membership: Gold
 - Region:
 - Transportation mode: Flight/Private Vehicle
 - Availability Status:
 - Average Rating:
- Boracay**
 - Package Type Name: Boracay
 - Package Type: Family/Solo/Friends
 - Duration in Days: 3
 - Guide Facility: false
 - Places Covered: Katiklan Boracay Island hopping
 - Membership: Basic/Gold/VIP
 - Region: Philippines
 - Transportation mode: Flight/Private Vehicle
 - Availability Status:
 - Average Rating:
- Island Getaway in Palawan**
 - Package Type Name: Island Getaway in Palawan
 - Package Type: Family/Honeymoon/Solo/Friends
 - Duration in Days: 5
 - Guide Facility: true
 - Places Covered: El Nido, Coron, Puerto Princesa
 - Membership: Basic/Gold/VIP
 - Region:
 - Transportation mode: Flight/Private Vehicle
 - Availability Status:
 - Average Rating:

User Management

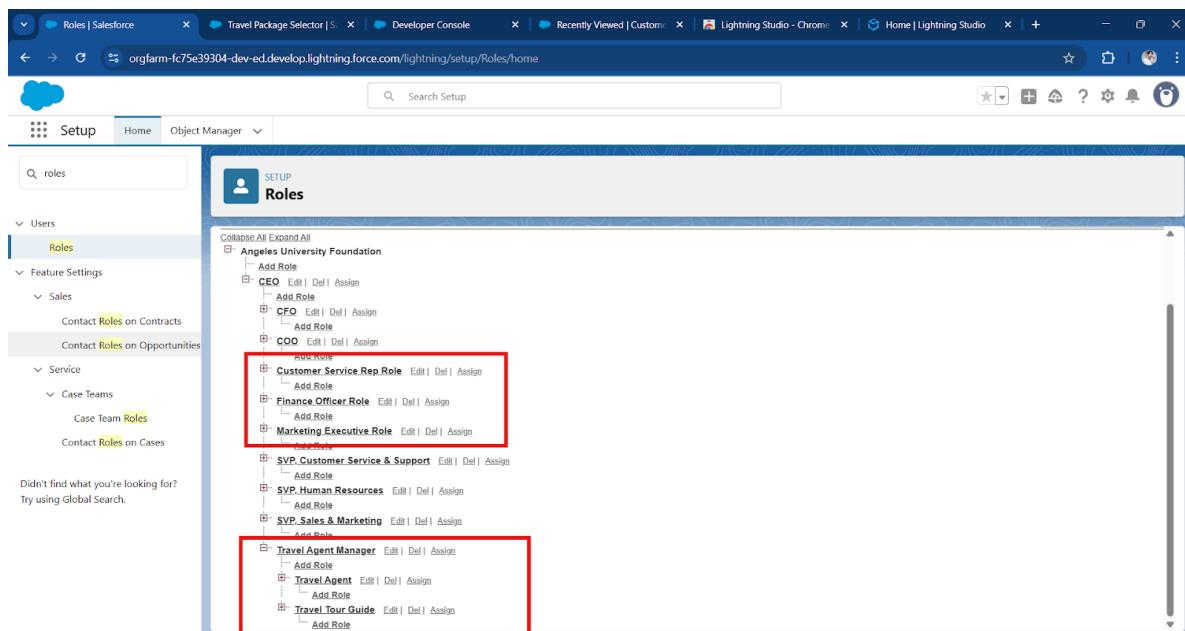
Created users for each role (e.g., Travel Agent, Tour Guide, Finance Officer) and assigned them appropriate Profiles and Roles. This ensured secure, role-specific access throughout the CRM.

- Users created for each role (Travel Agent, Guide, Finance, Marketing Executive Profile & Customer Service Rep) cloning Standard Platform User. I set Custom Object Permission for each Profile.



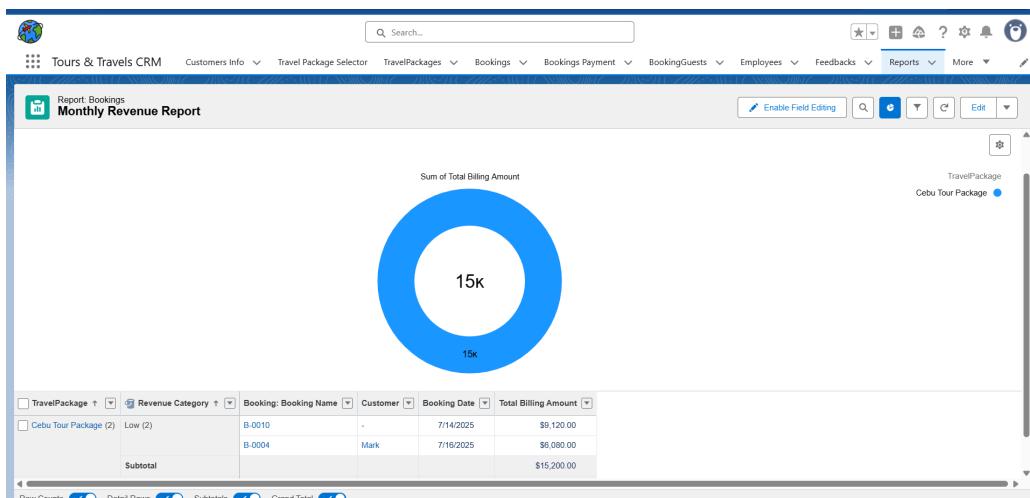
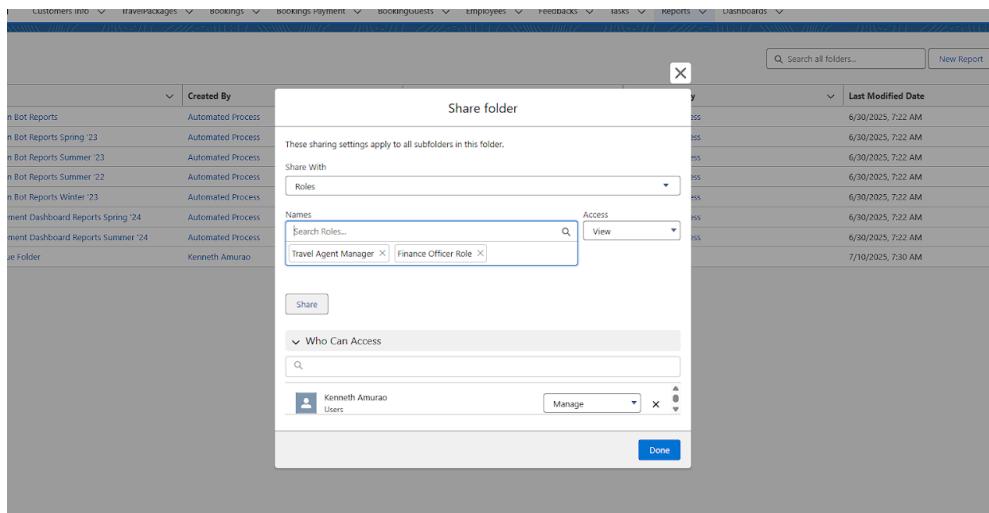
Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit	Amurao, Kenneth	amu	amurao.kenneth372@agentforce.com		✓	System Administrator
<input type="checkbox"/> Edit	Chatter Expert	chatty	00dgk00000swwwua0!ko0mxuqyh@chatter.salesforce.com		✓	Chatter Free User
<input type="checkbox"/> Edit	EPIC_OrgFarm	QEPI	epic.ln.8ce6bc-2820b5@orgfarm.salesforce.com		✓	System Administrator
<input type="checkbox"/> Edit	Jackson, Michael	mjack	nethamrao05@salesforce.com	Travel Agent Manager	✓	Travel Agent Profile
<input type="checkbox"/> Edit	Kent, Clark	ckent	sia.amurao kenneth@salesforce.com	Travel Agent	✓	Travel Agent Profile
<input type="checkbox"/> Edit	User Integration	integ	integration@00dgk00000swwwua0.com		✓	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User Security	sec	insightssecurity@00dgk00000swwwua0.com		✓	Analytics Cloud Security User
<input type="checkbox"/> Edit	Wayne, Bruce	bwayn	nethamrao@salesforce.com	Travel Agent	✓	Travel Agent Profile

- Assigned appropriate profiles and roles to enforce hierarchy.



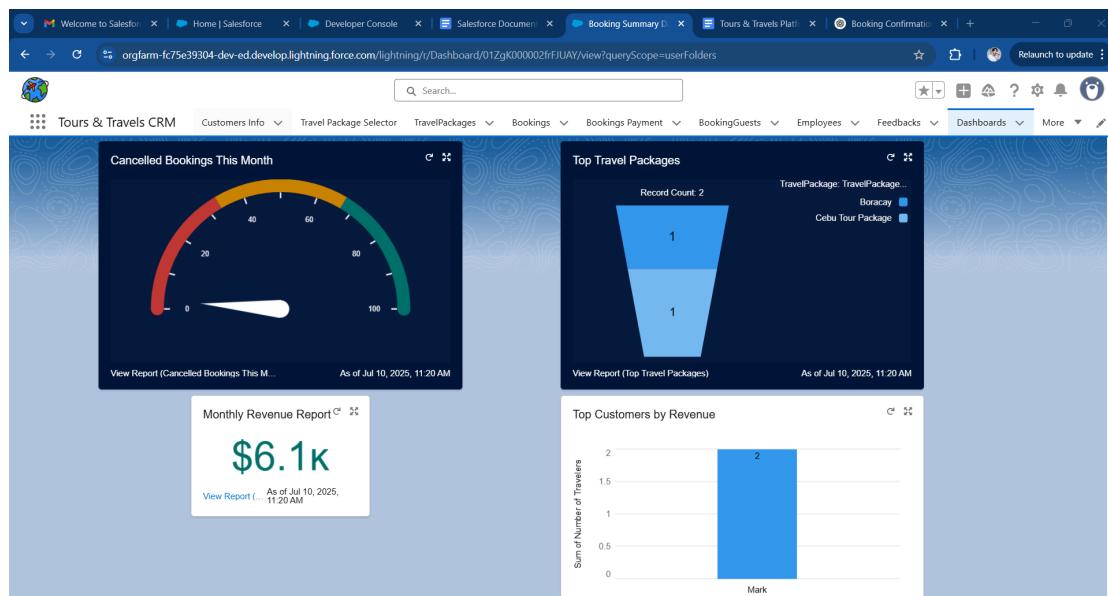
Reports and Dashboards

- Reports: Revenue, Top Packages, Pending Payments, Employee Role Breakdown.



Note: Create a “share folder” first, where you can assign a specific role and store the created report.

- Dashboards: Monthly Revenue, Booking Trends, Package Demand.



Note: If you Edit or Create a Dashboard you can customize your charts through add “widgets” KPI’s based on the created from “Reports”

Phase 4: Data Migration, Testing & Security

Data Loading Process

To bring existing data into the CRM, the **Salesforce Data Import Wizard** was used due to its simplicity and support for standard and custom objects. This tool allowed a user-friendly, guided process for importing:

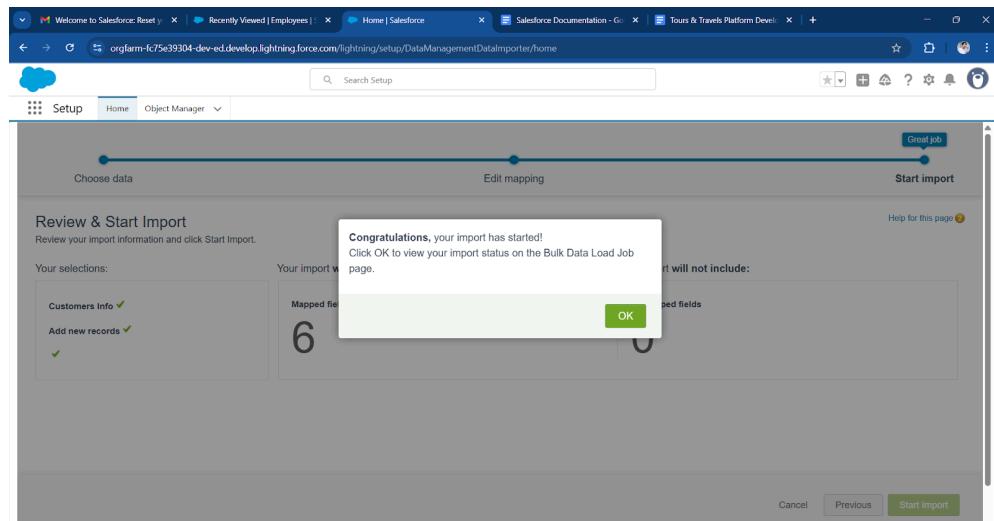
- **Objects Migrated:** Customer__c, TravelPackage__c, and Employee__c.
- Each CSV file contained a minimum of 20 records with clean data.
- Fields were carefully mapped to ensure accurate data alignment with the corresponding Salesforce fields.

Here is the sample Data Loading Process:

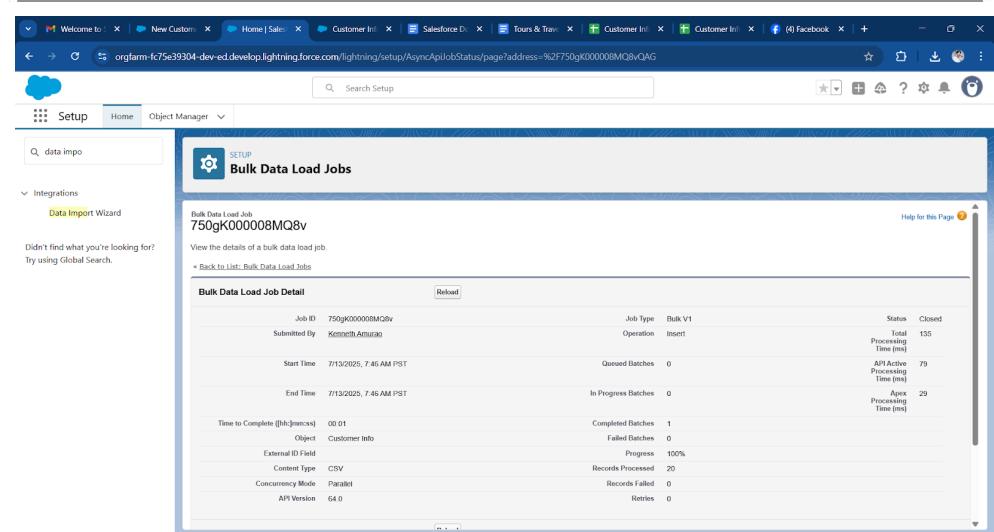
Customer Info CSV import wizard

The screenshot shows the 'Import your Data into Salesforce' step of the Data Import Wizard. The progress bar at the top indicates 'Getting closer...' with segments for 'Choose data', 'Edit mapping', and 'Start import'. The main area is divided into three columns: 'What kind of data are you importing?' (listing Standard objects like BookingGuests, Bookings, Payment, and Customers info), 'What do you want to do?' (selected 'Add new records'), and 'Where is your data located?' (File selected as Customer Info.csv, Character Code ISO-8859-1, Values Separated By Comma). Buttons at the bottom include 'Cancel', 'Previous', and 'Next'.

Import Map Field



The screenshot shows the 'Review & Start Import' step of the Data Import Wizard. It displays a summary of selected objects: 'Customers Info' and 'Add new records'. A progress bar at the top indicates the steps: 'Choose data', 'Edit mapping', and 'Start import'. A modal window titled 'Congratulations' informs the user that the import has started and provides a link to view the status on the Bulk Data Load Job page. Buttons for 'OK' and 'Cancel' are visible.



The screenshot shows the 'Bulk Data Load Jobs' page. It lists a single job entry: '750gK000008MQ8v'. The job details include:

Job ID	750gK000008MQ8v
Submitted By	Kenneth Amurao
Object	Customer Info
External ID Field	
Content Type	CSV
Concurrency Mode	Parallel
API Version	64.0
Start Time	7/13/2025, 7:46 AM PST
End Time	7/13/2025, 7:46 AM PST
Time to Complete (hh:mm:ss)	00:01

The job status is 'Closed' with a total processing time of 135 ms. The API active processing time was 79 ms, and the apex processing time was 29 ms. The job summary table shows:

Batch Status	Count
Queued Batches	0
In Progress Batches	0
Completed Batches	1
Failed Batches	0
Progress	100%
Records Processed	20
Records Failed	0
Retries	0

Milestone proof activity completed minimum 20 records for (Customer_Info__c, TravelPackage__c, and Employee__c.)

Job ID	Submitted By	Start Time	Status	Job Type	Operation	Object	Records Processed	Records Failed	Progress
Z50gK000000MWXX	Amrao, Kenneth	7/13/2025, 9:14 AM	Closed	Bulk V1	Insert	Employee	20	0	00:01
Z50gK0000008MYG	Amrao, Kenneth	7/13/2025, 9:06 AM	Closed	Bulk V1	Insert	Employee	20	0	00:00
Z50gK0000008MSUz	Amrao, Kenneth	7/13/2025, 8:41 AM	Closed	Bulk V1	Insert	Employee	20	19	00:01
Z50gK0000008MWEL	Amrao, Kenneth	7/13/2025, 8:28 AM	Closed	Bulk V1	Insert	Employee	20	17	00:01
Z50gK0000008MTdL	Amrao, Kenneth	7/13/2025, 8:06 AM	Closed	Bulk V1	Insert	TravelPackage	20	0	00:00
Z50gK0000008MCBv	Amrao, Kenneth	7/13/2025, 7:46 AM	Closed	Bulk V1	Insert	Customer Info	20	0	00:01
Z50gK0000008MPKv	Amrao, Kenneth	7/13/2025, 7:34 AM	Closed	Bulk V1	Insert	Customer Info	20	0	00:00

After import, the integrity of records was verified by reviewing the record list views and sample record detail pages.

Note: After clicking “View Record” it will automatically download imported record

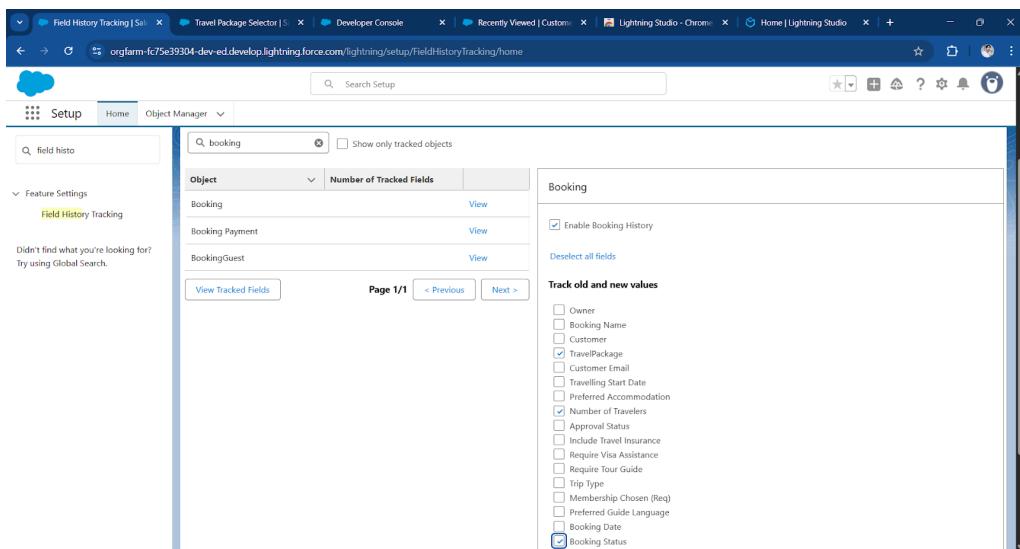
View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Message	Status
View Request	View Result	751gK000006dkLC	7/13/2025, 7:46 AM	7/13/2025, 7:46 AM	135	79	29	20	0	0	Completed	

A	B	C	D	E	F	G	H	I
1 City__c	Country__c	Customer_Name__c	DateOfBirth__c	Email__c	Phone__c			
2 Palayan	Philippines	Paul Ford	29/07/1992	paul.ford@yahoo.com	1953867218			
3 New York	USA	Amy Allen	07/12/1960	amy.allen@gmail.com	1975243167			
4 Palayan	Switzerland	Barbara Frye	31/01/1969	barbara.frye@yahoo.com	1986273561			
5 Bangalore	Thailand	Sarah Lucas	13/12/1978	sarah.lucas@yahoo.com	1973623187			
6 Jeddah	Thailand	Sabrina Cortez	01/06/1966	sabrina.cortez@yahoo.com	1987435612			
7 Quezon City	Thailand	Jason Lee Jr.	25/07/1987	jason.jr@yahoo.com	1983716481			
8 Bangalore	USA	Andrea Downs	30/01/1987	andrea.downs@gmail.com	1915736481			
9 Mumbai	Philippines	Ashley Lynn	15/03/2003	ashley.lynn@yahoo.com	1966381558			
10 Chennai	Thailand	Daniel Lindsey	08/02/2007	daniel.lindsey@yahoo.com	1967361852			
11 Mumbai	Thailand	Beth Collins	22/07/1982	beth.collins@yahoo.com	1975834658			
12 Hyderabad	Switzerland	Edwin Garcia	01/11/1996	edwin.garcia@gmail.com	1967843158			
13 Columbus	Thailand	Alec Medina	03/08/1970	alec.medina@yahoo.com	1987543185			
14 Bangalore	USA	Christopher Simon	29/03/2000	christopher.simon@gmail.com	1967391873			
15 Los Angeles	India	Jerry Lopez	12/10/2004	jerry.lopez@gmail.com	1948613287			
16 Alaminos	Thailand	Lindsay Lewis	16/09/1965	lindsay.lewis@gmail.com	1987328457			
17 Chicago	India	Haley Clayton	19/11/1967	haley.clayton@yahoo.com	1915746132			
18 Delhi	Philippines	Katherine Shaw	08/06/1976	katherine.shaw@gmail.com	1936857286			
19 Medina	Thailand	Chelsea Ashley	13/08/1977	chelsea.ashley@gmail.com	1978653264			
20 Geneva	Switzerland	Gerald Bryant	16/09/1988	gerald.bryant@gmail.com	1986432672			
21 Bern	India	Joshua Hill	01/02/1996	joshua.hill@gmail.com	1978436481			

Security Implementation

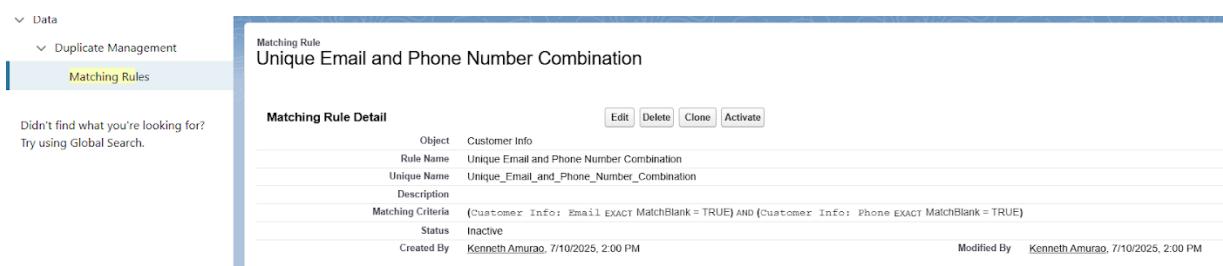
To protect business-sensitive information and ensure proper user access:

- **Field History Tracking:** Enabled for key fields on Booking__c and TravelPackage__c to monitor changes over time. This allows audit trails for tracking who changed what and when.



- **Duplicate Rules & Matching Rules:** Created on the Customer__c object to prevent duplicate entries based on unique combinations of Email and Phone.

Custom Matching Rule



Duplication Rule

Duplicate Rule Detail

Rule Name	unique Email and Phone	Object	Customer Info	Record-Level Security	Enforce sharing rules	Action On Create	Allow	Operations On Create	<input checked="" type="checkbox"/> Alert <input checked="" type="checkbox"/> Report
Description	unique Email and Phone	Alert Text	Email and iPhone must be Unique	Action On Edit	Allow	Matching Rule	<input checked="" type="checkbox"/> Unique Email and Phone Number Combination <input checked="" type="checkbox"/> Mapped	Operations On Edit	<input checked="" type="checkbox"/> Alert <input checked="" type="checkbox"/> Report
Conditions		Active	<input checked="" type="checkbox"/>	Matching Criteria	(Customer Info: Email EXACT MatchBlank = TRUE) AND (Customer Info: Phone EXACT MatchBlank = TRUE)				
Created By	Kenneth Amurao, 7/10/2025, 2:07 PM	Modified By	Kenneth Amurao, 7/10/2025, 2:10 PM						

- Profiles & Roles:** Users were assigned to predefined profiles such as Travel Agent, Finance Officer, and Tour Guide. Roles were used to determine visibility and record-level access.

Profile Detail

Name	Travel Agent Profile	Custom Profile	<input checked="" type="checkbox"/>
User License	Salesforce Platform	Created By	Kenneth Amurao, 7/9/2025, 8:33 AM
Description		Modified By	Kenneth Amurao, 7/9/2025, 8:35 /

Page Layouts

Standard Object Layouts	Global	Global Layout	Lead	Lead Layout
-------------------------	--------	---------------	------	-------------

- Permission Sets:** Used to extend object access for specific users (e.g., Travel Agent Manager needing report access) without changing their base profile.

Permission Set Overview

Description	API Name	Namespace Prefix	
License	Extra_Permission_For_Travel_Agent_Manager		
Session Activation Required	<input type="checkbox"/>	Created By	Kenneth Amurao, 7/10/2025, 2:30 PM
Permission Set Groups Added To	0	Last Modified By	Kenneth Amurao, 7/10/2025, 2:30 PM

- **Sharing Rules:** Configured to share customer records automatically with Tour Guide roles based on their association with bookings.

Action	Criteria	Shared With	Access Level
Edit Del	Owner in Role: Travel Agent	Role: Travel Tour Guide	Read Only

Test Class Creation

- **Apex Test Class:** A test class named BookingTriggerTest was developed to verify the logic inside Booking triggers and handlers. The class inserted mock records and confirmed that associated BookingGuest__c and BookingPayment__c records were automatically created.
- **Challenges and Error:** The biggest challenge is that some of the fields given in Milestone 4 are not aligned with the code provided in BookingTriggerHandler.apxc and BookingTriggerTest.apxc. Some of the errors are in the BookingTriggerTest.apxc file, where certain fields are not properly defined in Customer_Info__c and Membership_Chosen__c. For example, I encountered a "Missing Date_Of_Birth__c" error on BookingTriggerTest.apxc. To fix this, I added the code: Date_of_Birth__c = Date.today().addYears(-30);
- **Coverage:** More than 90% code coverage was achieved across Apex triggers and classes.

Status	Test Run	Enqueued Time	Duratio	Failures	Total
x	BookingTriggerTest			1	1
x	TestRun @ 2:30:14 am			1	1
x	BookingTriggerTest			1	1
x	TestRun @ 2:35:42 am			1	1
x	BookingTriggerTest			1	1
x	testTriggerCreatesPaymentAndGuestsWithUpdatedFields			1	1
x	TestRun @ 2:59:01 am			1	1
x	BookingTriggerTest			1	1
x	testTriggerCreatesPaymentAndGuestsWithUpdatedFields			0:00	0:00
✓	TestRun @ 2:59:49 am			0	1
✓	BookingTriggerTest			0	1
✓	TestRun @ 3:00:01 am			0	1
✓	BookingTriggerTest			0	1
✓	testTriggerCreatesPaymentAndGuestsWithUpdatedFields			0:00	0:00

BookingTriggerTest.apxc

@isTest

```
private class BookingTriggerTest {
```

@isTest

```
        static void testTriggerCreatesPaymentAndGuestsWithUpdatedFields() {
```

```
            // Create test customer
```

```
            Customer_Info__c customer = new Customer_Info__c(
```

```
                Name = 'John Doe',
```

```
                Email__c = 'annapurna@gmail.com', // enter your valid email address here
```

```
                Phone__c = '1234567890',
```

```
                DateOfBirth__c = Date.today().addYears(-30)
```

```
            );
```

```
            insert customer;
```

```
            // Create a Travel Package
```

```
            TravelPackage__c packageRec = new TravelPackage__c(
```

```
                Name = 'European Delight',
```

```
                Country__c = 'India',
```

```
                Price_Per_Person__c = 2000,
```

```
                Duration_in_Days__c = 3,
```

```
                Places_Covered__c = 'Paris, Rome, Barcelona'
```

```
            );
```

```
            insert packageRec;
```

```

// Create a test Booking record

Booking__c booking = new Booking__c

    Number_of_Travelers__c = 3,
    Booking_Status__c = 'Pending',
    Travelling_Start_Date__c = Date.today().addDays(10),
    TravelPackage__c = packageRec.Id,
    Membership_Chosen_Req__c = 'Gold',
    Preferred_Accommodation__c = 'Guest House',
    Customer_Info__c = customer.Id,
    //Customer__c = customer.Id, // fixed field name

    Customer_Email__c ='annapurna@gmail.com', // enter a valid email address here

    Booking_Date__c = Date.today() // added required field

);

Test.startTest();

insert booking; // Trigger will fire here

Test.stopTest();

// Validate Booking_Payment__c creation

List<Booking_Payment__c> payments = [
    SELECT Id, Booking__c, Payment_Status__c
    FROM Booking_Payment__c
    WHERE Booking__c = :booking.Id
];

System.assertEquals(1, payments.size(), 'One payment record should be created.');

```

```
        System.assertEquals('Pending', payments[0].Payment_Status__c, 'Default Payment Status  
should be Pending.');

    // Validate BookingGuest__c creation

    List<BookingGuest__c> guests = [  
  
        SELECT Id, Booking__c, Name  
  
        FROM BookingGuest__c  
  
        WHERE Booking__c = :booking.Id  
  
    ];  
  
    System.assertEquals(3, guests.size(), 'Three BookingGuest records should be created.');

    System.assertEquals('Guest 1', guests[0].Name, 'Guest naming should follow  
convention.');
```

}

Testing Execution

Test Case 1: Ensure that a new Customer record can be added successfully when all required fields are filled.

Test Steps:

- Open the Customer module or object page.
- Enter values for all mandatory fields.
- Click on the **Save** button

Expected Result:

- The Customer record should be created and stored without any errors.
- The newly added record should be visible in the list of existing Customers.

Welcome to Salesforce | Kenneth Amurao | Travel Package Select | Recently Viewed | Developer Console | Facebook | Sales

orgfarm-fc75e39304-dev-ed.develop.lightning.force.com/lightning/r/Customer_Info_c/a00gK00000AWyuYQAT/view

Tours & Travels CRM | Customers Info | Travel Package Selector | Travel Packages | Bookings | Bookings Payment | Booking Guests | Employees

Customer Info Kenneth Amurao

Customer Info "Kenneth Amurao" was created.

Related Details

Customer ID C-0003	Date Of Birth 7/5/2025
Customer Info Name Kenneth Amurao	Age 0
Email nethamurao05@gmail.com	Owner Kenneth Amurao

Activity

3 items • Sorted by Customer Info Name • Updated a few seconds ago

	Customer Info Name ↑
1	<input type="checkbox"/> kenneth
2	<input type="checkbox"/> Kenneth
3	<input type="checkbox"/> Kenneth Amurao

Test Case 2: Confirm that a Booking can be successfully recorded when all required information is provided.

Test Steps:

- Go to the Booking object interface.
- Complete all necessary fields with appropriate values.
- Click **Save** to submit the form.

Expected Result:

- A new Booking entry should be created and saved properly.
- The record should show up on the Booking List view.
- A corresponding entry should automatically be created in the Booking Payments object with the Payment Status set to "Pending".
- A number of BookingGuest entries equal to the "Number of Travellers" specified in the Booking should also be created and linked to the Booking record.

The screenshot displays two views of the Tours & Travels CRM system. The top view shows the 'Booking' details for record B-0010. The 'Details' tab is selected, showing fields for Booking Number (B-0010), Customer (Kenneth Amurao), and Customer Email (nethamurao05@gmail.com). It also shows the Number of Travelers (3), Booking Status (Pending), and Travel Cost Per Person (\$40.00). The bottom view shows the 'Recently Viewed' list, which includes three items: B-0004, B-0007, and B-0010. The item B-0010 is highlighted with a red box and has a red arrow pointing from the 'Booking Name' column towards it, indicating its recent viewing.

Test Case 3: Verify that when the Payment Status of a Booking Payment record is changed to "Completed", the related Booking record updates to "Confirmed", and the customer is notified via email.

Test Steps:

- Navigate to the Booking Payment object page.
- Change the **Payment Status** field value to "Completed".
- Save the changes.

Expected Result:

- The Booking Payment record should be updated successfully.
- The corresponding Booking record should automatically reflect a "Confirmed" status.
- An email should be sent to the customer confirming the Booking and successful payment.

Test Case 4: Booking Creation With Employee TourGuide

Objective:

- Prevent booking with a past travel start date.

Test Steps:

- Navigate to the Booking object.
- Create a new booking.
- Fill other required fields
- Checked the “Required” Tou Guide”.
- Create or add Existing Employee and click Save.

Expected Result:

- Successful Banner Appear then te Guide employee assigned will booking details

Validation Area:

- Lookup Filter on Employee__c field in booking, Role and Availability

The screenshot shows a Salesforce booking creation interface. At the top, there's a header with a phone icon and the text "Booking B-0011". Below the header, there are several input fields and sections:

- Travelling End Date:** 8/2/2025
- Trip Type:** Family
- Membership Chosen (Req):** Gold
- Preferred Accommodation:** Guest House
- Include Travel Insurance:** (checkbox)
- Require Visa Assistance:** (checkbox)
- Require Tour Guide:** (checkbox) - This field is highlighted with a red border and contains a checked checkbox.
- Guide Assigned:** E-0005
- Preferred Guide Language:** Filipino
- No. of Booking Guests Info Available:** 3

On the right side of the screen, there are some summary amounts:

- Total Accommodation Amount:** \$12,000.00
- Total Billing Amount:** \$12,120.00
- Owner:** Kenneth Amurao (with a small profile icon)

Test Case 5: Duplicate Customer Creation

Objective:

- Prevent creation of customers with same phone and email.

Test Steps:

- Create a Customer Info record with Email = Gabby@gmail.com and Phone = 1234567890.
- Try creating another customer with the same email and phone.

Expected Result:

- Error: "Email and Phone must be Unique."
- Validation Area:** Duplicate rule, Matching rule.

The screenshot shows a CRM application interface for creating a new customer. On the left, there's a sidebar with a list of recently viewed customer info items. The main window is titled 'New Customer Info' and has an 'Information' tab selected. In the 'Customer Info Name' field, 'Rose Mary' is entered. The 'Email' field contains 'gabby@gmail.com' and the 'Phone' field contains '9275768408'. A modal dialog box titled 'Similar Records Exist' is displayed, warning the user that the record looks like an existing one and suggesting they check for duplicates before saving. At the bottom of the main form, there are 'Cancel', 'Save & New', and 'Save' buttons.

Test Case 6: Dynamic Field Visibility

Objective:

- Verify dynamic forms hide/show fields.

Test Steps:

- Open a Booking record with Booking Status = Cancelled.
- Check if *Cancellation Date*, *Cancel Confirmation*, *Approval Status* are visible.

Expected Result:

- Fields are visible only when status = Cancelled.

Validation Area:

- Dynamic Forms.

The screenshot shows a 'New Booking' form in the Tours & Travels CRM. The 'Booking Status' field is set to 'Confirmed'. A red error message box appears at the bottom left, stating '∅ We hit a snag.' and listing 'Review the following fields' with a bullet point for 'Booking_Status'. The right side of the screen shows a sidebar with various tabs like Import, Change Owner, and Assign Label.

Test Case 7: Trip Cancellation Approval Workflow

Objective:

- Validate approval process when booking is cancelled.

Test Steps:

- Update an existing Booking record:
- Booking Status = Cancelled
- Cancel Confirmation = True
- Submit the record for approval.

Expected Result:

- Approval request sent to manager.
- Email sent upon approval or rejection.

Validation Area: Approval process, email templates.

The screenshot displays two main sections of a booking management application. On the left, the 'Details' tab of a booking record is shown. The booking number is B-0013, and the customer is Gabriel Aquino. The travel package is Cebu Cultural Tour, and the booking date is 7/17/2025. The travelling start date is 8/1/2025, and the end date is 8/5/2025. The trip type is Group. The membership chosen is required. The booking status is set to 'Cancelled', and the cancel confirmation is checked. The number of travelers is 2. The travel cost per person is \$300.00, and the total travel amount is \$600.00. The accommodation amount per person per day is \$3,000.00. The cancellation reason is listed as 'Cancelling Reason'. On the right, the 'Activity' section shows a header with filters for all time, all activities, and all types, with options to refresh, expand all, or view all. It indicates no upcoming or overdue activities. A note says 'Get started by sending an email, scheduling a task, and more.' Below this, it states 'No past activity. Past meetings and tasks marked as done show up here.'

Phase 5: Deployment, Documentation & Maintenance

Deployment Strategy

The deployment methodology for the Tours & Travels CRM system was executed through the use of **Change Sets**, a widely accepted and secure method in the Salesforce ecosystem. This process entailed:

- **Preparation of Outbound Change Sets:** All essential metadata components—including custom objects, fields, automation flows, Apex classes, analytical reports, dashboards, and user access configurations (profiles and permission sets)—were aggregated and packaged within the source environment.
- **Transmission to Target Environment:** The prepared change set was transmitted to the designated target instance, whether production or staging.
- **Validation and Execution:** Prior to final deployment, a comprehensive validation step was conducted to confirm that all required dependencies were included. Once validated, the components were successfully deployed.
-

Maintenance and Monitoring Plan

To ensure continuous operational integrity and system resilience, the following monitoring and maintenance protocols are established:

- **Audit Log Supervision:** Login histories and setup audit trails are reviewed on a regular basis to detect unauthorized access or administrative changes.
- **Scheduled Job Oversight:** The statuses of automated jobs, including both Schedulable and Batch Apex executions, are monitored through the Apex Jobs interface.
- **Data Quality Assurance:** Duplicate management rules and validation rules are utilized to ensure the integrity and reliability of entered data.
- **Field History Analysis:** Key object fields (such as those in Bookings and Travel Packages) are tracked to audit data changes and facilitate debugging.
- **Access Control Reviews:** Profiles, Permission Sets, and Role assignments are periodically audited to maintain appropriate data visibility and compliance.

Troubleshooting Approach

In the event of unexpected application behavior or systemic errors, a formalized troubleshooting methodology is employed:

1. **User Issue Documentation:** Collect and analyze user-submitted reports or support tickets detailing the observed issues.
2. **Debug Log Activation:** Enable debug logs for relevant users or classes to trace runtime exceptions and performance bottlenecks.
3. **Automation Failure Inspection:** Examine error logs in Flows and Process Builders to identify failed or misconfigured automation.
4. **Embedded Error Messaging:** Utilize predefined error messages within Flows and Apex logic to guide users and expedite issue identification.
5. **Deployment Diagnostics:** Review validation logs in Change Sets to troubleshoot deployment failures and missing dependencies.

Conclusion

The development and implementation of the Tours & Travels CRM platform on Salesforce comprehensively fulfill the strategic and operational needs of modern travel service providers. From efficient customer onboarding and itinerary bookings to automated approval workflows and payment tracking, the system has proven effective in enhancing organizational efficiency.

By integrating declarative tools such as Flows and Reports alongside programmatic assets like Apex Triggers, Classes, and Lightning Web Components, the solution is both robust and extensible. The modular design ensures adaptability to future business requirements and facilitates potential upgrades such as AI-driven recommendations, chatbot support, and mobile accessibility.

The successful culmination of this project reflects a thorough understanding of Salesforce architecture and development best practices, reinforcing the project's viability for real-world deployment.