


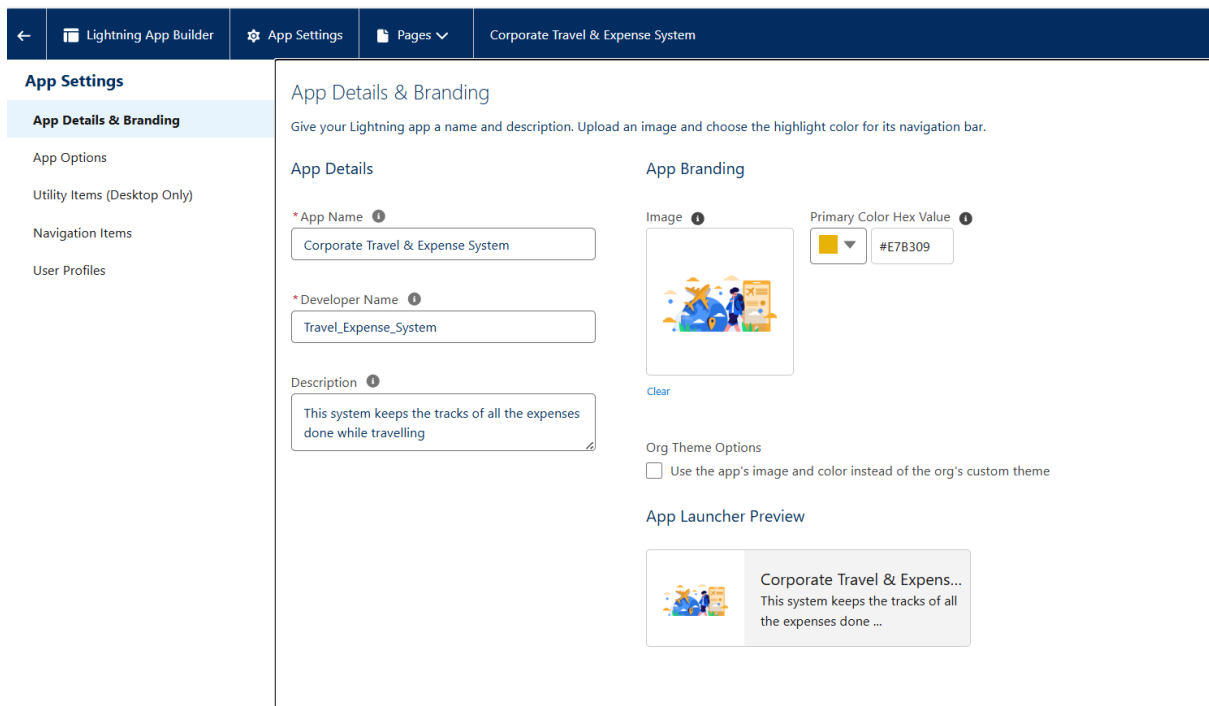
Corporate Travel & Expense Management System Project

Phase 6: User Interface Development

 **Goal:** Make it user-friendly and interactive.

1. Lightning App Builder

- Create a custom Lightning App “**Corporate Travel & Expense Management**” app.
- API metadata: **AppDefinition.DeveloperName**
=Travel_Expense_System.



The screenshot shows the Lightning App Builder interface for an app named "Corporate Travel & Expense System". The left sidebar contains the "App Settings" menu with options: "App Details & Branding" (selected), "App Options", "Utility Items (Desktop Only)", "Navigation Items", and "User Profiles". The main content area is titled "App Details & Branding" and includes the instruction: "Give your Lightning app a name and description. Upload an image and choose the highlight color for its navigation bar." The "App Details" section contains three fields: "App Name" (filled with "Corporate Travel & Expense System"), "Developer Name" (filled with "Travel_Expense_System"), and "Description" (filled with "This system keeps the tracks of all the expenses done while travelling"). The "App Branding" section includes an "Image" field with a travel-themed icon, a "Primary Color Hex Value" field (filled with "#E7B309"), and a "Clear" link. Below this is the "Org Theme Options" section with a checkbox labeled "Use the app's image and color instead of the org's custom theme". At the bottom is the "App Launcher Preview" section, which shows a preview of the app's launcher card with the app's icon and name.

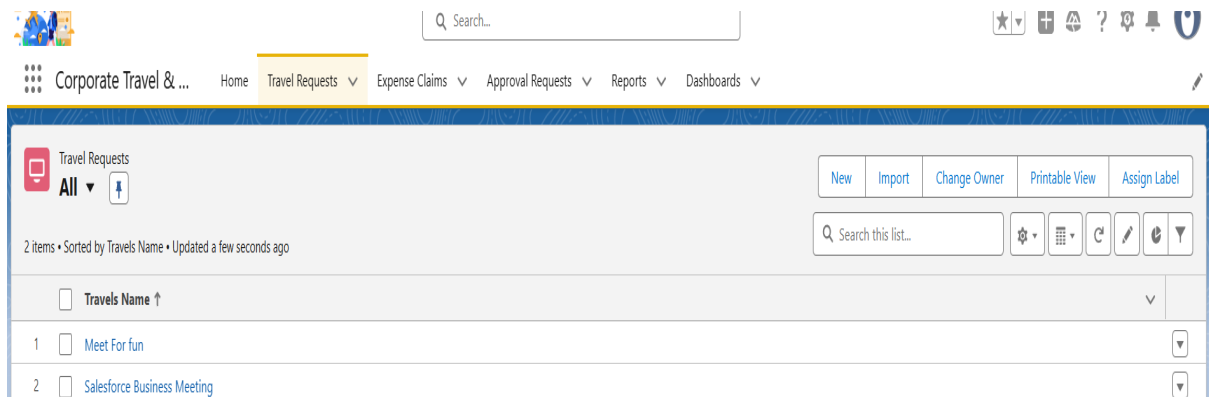
2. Record Pages

- **Travel Request Page (Travel_Request__c)** → show related list of **Expense_Claim__c**.

- **Expense Claim Page (Expense_Claim__c)** → show lookup to **Travel_Request__c** + Files (Receipts).

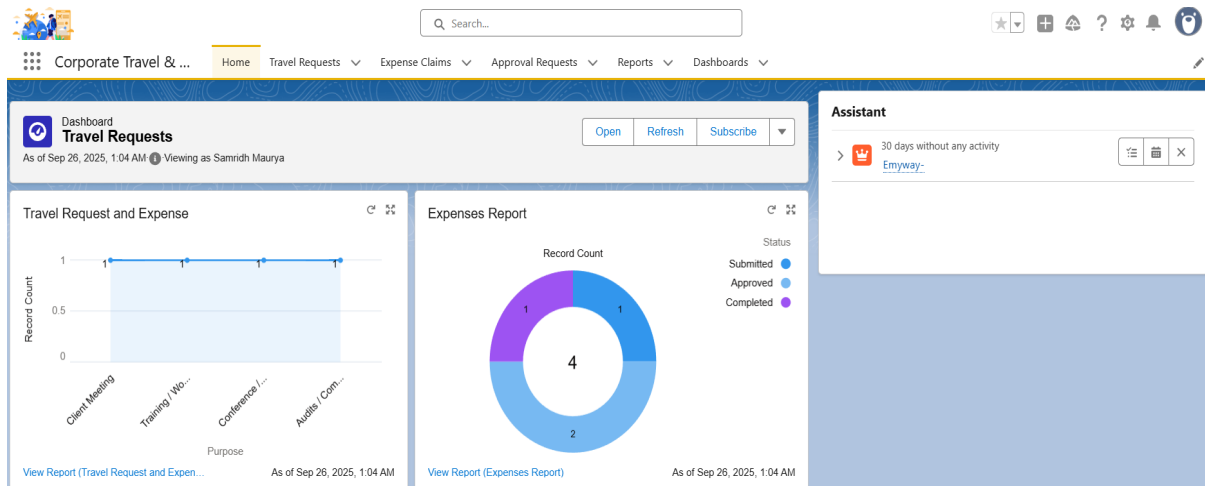
3. Tabs

- Add **Travel Requests Tab** → **Travel_Request__c**.
- Add **Expense Claims Tab** → **Expense_Claim__c**.



4. Home Page Layouts

- Add Dashboard component showing:
 - ✓ Pending Approvals (**Travel_Request__c.Status__c** = "Submitted").
 - ✓ Recent Claims (**Expense_Claim__c.Status__c** = "Submitted").



5. Utility Bar

- Quick Action: **New Travel Request** → launches standard action

Travel_Request__c.New.

- Quick Action: **New Expense Claim** → launches
- Expense_Claim__c.New.**

SETUP > OBJECT MANAGER

Travel Request

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

Travel Request Action

New Event

Predefined Field Values (0)

Action Detail

Label	Value	Object Name	Action Type
Standard Label Type	New [Record]	Travel Request	Create a Record
Name	New_Travel_Request	Relationship Field	Related To
Description			
Target Object	Event	Icon	
Create Feed Item	✓		
Success Message			
Created By	Samrith Maurya, 9/26/2025, 1:14 AM	Modified By	Samrith Maurya, 9/26/2025, 1:14 AM

Predefined Field Values

No records to display

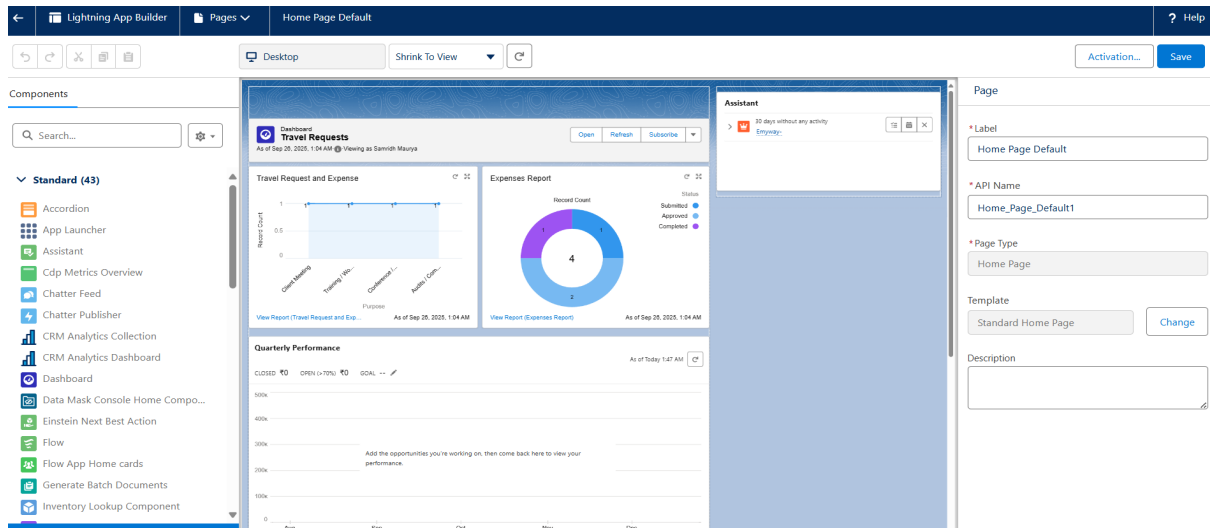
Back To Top

Always show me more records per related list

6. LWC

- Component Name: **searchTravelRequests.**
- Function: Search **Travel_Request__c** records by:
 - Status__c** (picklist).

- **OwnerId** (Employee/Manager).
- Display results in lightning-datatable.



7. Apex with LWC

- **Imperative Apex call** → create Travel Request with related Expense Claims.
- Method:

```
public static Id createTravelRequest(Travel_Request__c req,
List<Expense_Claim__c> claims) {

    insert req;

    for(Expense_Claim__c c : claims){

        c.Travel_Request__c = req.Id;

    }

    insert claims;

    return req.Id;
}
```

}

- Called imperatively from LWC → creates **Travel_Request__c** with related **Expense_Claim__c**.

8. Events in LWC

- **Child LWC** → search form (**searchTravelRequestsForm**).
- **Parent LWC** → results table (**searchTravelRequestsResults**).
- **Event API**: **this.dispatchEvent(new CustomEvent("search"))**.

9. Wire Adapters

- Use **@wire** (getRecords, {objectApiName: 'Travel_Request__c', fields: ['Purpose__c','Start_Date__c','Status__c']}).
- Filters → **Status__c = 'Submitted'** for Manager view.

10. Imperative Apex Calls

- On click “**Submit Claim**” → call
ExpenseClaimController.createClaim(Expense_Claim__c).

11. Navigation Service

- After record creation → use **NavigationMixin.Navigate** →
recordPage with recordId = **Travel_Request__c.Id** or
Expense_Claim__c.Id.
-