🔵 Phase 1: Problem Understanding & Industry Analysis

Project Name: AttendSync – Event Automation CRM
Platform: Salesforce (Admin + Developer)
Phase Owner: Akash Bhujbale.

🎯 Objective

The goal of Phase 1 is to deeply understand the business challenge, analyze the target industry, identify key stakeholders, and define a solution scope that aligns with real-world use cases. This phase lays the strategic foundation for all technical implementation.

🧠 Problem Statement

Event organizers often rely on manual processes to manage ticketing and attendee communication. This leads to:

- Missed reminders and poor attendance
- Lack of visibility into upcoming events
- No centralized dashboard for tracking ticket status
- Inefficient workflows for sending updates and confirmations

These challenges are especially common in educational institutions, cultural organizations, and internal corporate teams. A scalable, automated solution is needed to streamline communication and improve engagement.

🏢 Industry Analysis

AttendSync targets the **event management sector**, with a focus on:

- **Educational Institutions**: Seminars, workshops, guest lectures
- **Cultural Organizations**: Festivals, performances, community events

- **Corporate Teams**: Internal trainings, team-building events, product launches

Common Pain Points:

Challenge	Impact
Manual ticketing	Time-consuming and error-prone
No automated reminders	Missed events and poor turnout
Fragmented communication	Confusion among attendees
No dashboards	Organizers lack visibility

Salesforce is ideal for solving these problems due to its automation tools, data modeling flexibility, and secure communication features.

📋 Requirement Gathering

Functional Requirements:

- Create and manage event tickets
- Associate tickets with attendees
- Send automated email reminders one day before the event
- Track ticket status (Confirmed, Cancelled, Pending)
- View upcoming events and reminder logs

Non-Functional Requirements:

- Scalable flow logic
- Reusable components (email templates, flows)
- Clean UI for organizers and admins

- Secure access via profiles and permission sets
- Debugging and rollback support

👥 Stakeholder Analysis

Stakeholder	Role	Needs
Organizer	Creates events and tickets	Automation, visibility, control
Attendee	Receives ticket and reminders	Timely communication, clarity
Admin	Manages system and flow logic	Monitoring, security, reporting

---

🔁 Business Process Mapping

Core Workflow:

1. Organizer creates a ticket for an event
2. Ticket is linked to an attendee
3. A scheduled flow runs daily to find tickets for tomorrow
4. Flow sends personalized reminder emails
5. Admin monitors flow execution and email delivery

🛠️ AppExchange Exploration

Explored AppExchange for:

- Email template accelerators
- RSVP tracking components

- Event dashboards

While custom development was prioritized, AppExchange tools were considered for future enhancements.

Phase 2

🔵 Phase 2: Org Setup & Configuration

ProjectName: AttendSync – Event Automation CRM
Platform:Salesforce(Admin+Developer)
Phase Owner: Akash Bhujbale

🧩 Configuration Components

1. Salesforce Edition

AttendSync was built using the **Developer Edition**, which provides full access to customization, Apex development, and automation features. This edition is ideal for proof-of-concept and portfolio-grade projects.

---

2. Company Profile Setup

- **Time Zone**: India Standard Time (IST)
- **Locale**: English (India)
- **Currency**: INR
  These settings ensure that scheduled flows run at the correct time and emails reflect regional formatting.

---

3. Business Hours & Holidays

Defined business hours to align with working schedules and prevent flows from triggering outside active hours. Added public holidays to avoid sending reminders on non-working days.

---

4. Fiscal Year Settings

Used **Standard Fiscal Year** for simplicity. AttendSync does not require custom fiscal logic, but this setting ensures compatibility with reporting and dashboards.

---

5. User Setup & Licenses

Created test users to simulate real-world roles:

- **Organizer**: Manages events and tickets
- **Attendee**: Receives ticket and reminders
- **Admin**: Oversees automation and reporting Assigned appropriate licenses and profiles to each user.

---

6. Profiles, Roles & Permission Sets

- Created custom **Profiles** for Organizer and Attendee
- Defined **Role Hierarchy** to control record visibility
- Used **Permission Sets** to grant access to flows, email actions, and custom objects

---

7. Organization-Wide Defaults (OWD) & Sharing Rules

- Set Ticket__c OWD to **Private** to protect attendee data
- Created **Criteria-Based Sharing Rules** to allow organizers to view tickets for their events

---

8. Login Access Policies

Enabled login access for support and debugging during flow testing and Apex deployment. This ensures secure collaboration and troubleshooting.

---

9. Dev Org Setup & Sandbox Usage

AttendSync was built in a **Developer Org**. Sandbox environments were prepared for staging and testing future enhancements, ensuring safe deployment practices.

---

10. Deployment Basics

- Used **Change Sets** for metadata migration
- Configured **VS Code with SFDX** for Apex and LWC deployment
- Verified deployment paths and rollback strategies

Phase 3

🧩 Custom Object Design

AttendSync uses two core custom objects:

1. Ticket__c

Represents an individual event ticket issued to an attendee.

Key Fields:

- **Subject__c (Text):** Name of the event
- **Event_Date__c (Date):** Scheduled date of the event
- **Location__c (Text):** Venue or platform
- **Status__c (Picklist**): Values include Confirmed, Cancelled, Pending
- **Attendee__c (Lookup**): Links to the attendee record

2. Attendee__c

Represents the person attending the event.

Key Fields:

- Name (Standard)
- Email__c (Email): Used for sending reminders
- Phone__c (Phone): Optional contact field
- Object Manager → Ticket__c and Attendee__c detail pages
- Fields & Relationships tab for both objects

🔗 Relationships & Schema

🔹 Lookup Relationship

- Ticket__c has a Lookup to Attendee__c
- This allows each ticket to be associated with one attendee
- Lookup was chosen over Master-Detail to preserve record independence and allow deletion flexibility

🔹 Master-Detail & Hierarchical

- Not used in current version
- Lookup provides better control for automation and UI customization

---

🔹 Schema Builder

Used to visualize object relationships and field dependencies. This helped confirm that Ticket__c and Attendee__c are correctly linked and that all required fields are present.

Benefits:

- Quick overview of data model
- Easy to spot missing relationships
- Helps during flow and Apex design

📃 Page Layouts & Compact Layouts

🔹 Page Layouts

Customized layouts for different roles:

- Organizer: Sees full ticket details including status and attendee

- Attendee: Sees event name, date, and location
- Admin: Sees system fields and flow logs

🔹 Compact Layouts

Used for mobile views and highlights:

- Subject__c
- Event_Date__c
- Status__c

🧱 Record Types

Currently, AttendSync uses a single record type for Ticket__c. Future enhancements may include:

- Seminar Ticket
- Festival Ticket
- Corporate Event Ticket

🔀 Junction Objects

Not used in current version. Could be added later to support many-to-many relationships (e.g., multiple attendees per ticket or multiple events per attendee).

🌐 External Objects

Not used in current version. Could be integrated later for syncing with external event platforms or RSVP systems via Salesforce Connect.

🔵 Phase 4: Process Automation (Admin)

⚙️ Tools & Techniques Used

Tool	Purpose
Flow Builder	Scheduled automation for reminders
Email Alerts	Personalized communication
Assignment Elements	Dynamic data preparation
Debug Logs	Flow testing and validation
Scheduled Jobs	Monitoring execution

🔁 Flow Overview: Event Ticket Reminder – V5

🔹 Trigger Configuration

- Type: Scheduled Flow
- Frequency: Daily
- Start Time: 11:10 AM IST
- Start Date: September 26, 2025
- Purpose: Identify tickets for events happening tomorrow and send reminders

---

📧 Email Configuration

🔹 Sender Setup

- Used Organization-Wide Email Address: akashbhujbale112@gmail.com
- Verified in Setup → Email → OWEA
- Selected in Send Email action

🔹 Email Body

Used merge fields for personalization:

Hi {!Attendee.Name},

Just a reminder that {!Loop_Through_Tickets.Subject__c} is scheduled for tomorrow at {!Loop_Through_Tickets.Location__c}.

You can view your ticket and dashboard for updates.

Thanks!

🔄 Additional Automation Elements

🔹 Assignment Elements

Used to extract and store:

- Attendee email
- Attendee name
- Ticket subject and location

🔹 **Email Deliverability**

- Set to All Email in Setup → Deliverability
- Ensures emails are sent without restrictions

🔹 **Scheduled Jobs**

- Monitored via Setup → Apex Jobs
- Verified that flow runs daily at configured time

🔵 Phase 5: Apex Programming (Developer)

⚙️ Apex Components Implemented

1. Apex Classes

Created Apex classes to encapsulate logic for scheduled jobs and utility functions. These classes handle tasks like auto-closing tickets or preparing data for dashboards.

---

2. Apex Triggers

Optional: You may have implemented a trigger on Ticket__c to send confirmation emails or log metadata when a ticket is created or updated.

- **Trigger Type**: After Insert
- **Purpose**: Send confirmation email immediately after ticket creation

---

3. Trigger Design Pattern

Used a handler class to separate logic from the trigger, ensuring maintainability and testability.

- **Structure**: Trigger → Handler → Utility
- **Benefits**: Cleaner code, reusable logic, easier testing

---

4. SOQL Queries

Used SOQL to retrieve ticket and attendee data based on conditions like event date and status.

Example:

SELECT Id, Subject__c, Event_Date__c FROM Ticket__c WHERE Event_Date__c = :tomorrow AND Status__c = 'Confirmed'

---

5. Collections: List, Set, Map

Used Lists to store ticket records and Maps to associate attendees with their emails for bulk operations.

- **List<Ticket__c> tickets**
- **Map<Id, Attendee__c> attendeeMap**

---

6. Control Statements

Used if, for, and while statements to loop through records and apply conditional logic.

🔄 Asynchronous Apex

7. Scheduled Apex

Implemented a scheduled job to auto-close tickets or trigger flows at specific intervals.

- **Class**: AutoCloseJob
- **Schedule**: Daily at 11:59 PM IST
- **Purpose**: Update ticket status to “Closed” for past events

---

8. Future Methods

Used for asynchronous email sending or logging tasks that don’t need immediate completion.

Example:

```
@future
public static void sendReminderEmail(String attendeeEmail) {
   // email logic
}
```

---

9. Queueable Apex

Used for lightweight asynchronous tasks like sending confirmation emails or logging ticket metadata.

- **Class**: ReminderQueue
- **Interface**: Implements Queueable

---

10. Exception Handling

Used try-catch blocks to gracefully handle errors during flow execution or Apex logic.

Example:

```
try {
   // logic
} catch (Exception e) {
   System.debug('Error: ' + e.getMessage());
}
```

🧪 Testing & Deployment

11. Test Classes

Created test classes to validate Apex logic, ensure code coverage, and prepare for deployment.

- **Test Class**: TestAutoCloseJob
- **Coverage**: >90%
- **Assertions**: Verified ticket status updates and email logic

---

12. Apex Jobs Monitoring

Monitored scheduled jobs via Setup → Apex Jobs to ensure timely execution and error-free runs

---

🔵 Phase 6: User Interface Development

🧩 UI Strategy Overview

AttendSync's interface is built around three key principles:

- **Role-Based Access**: Organizers, attendees, and admins see only what's relevant to them
- **Modular Layouts**: Tabs, record pages, and dashboards are organized for clarity
- **Dynamic Components**: LWC components fetch and display real-time data

---

🛠️ Lightning App Builder Configuration

🔹 Custom App: AttendSync

- Created a custom Lightning App named **AttendSync**
- Added tabs for:
  - Ticket Management
  - Attendee Directory
  - Event Dashboard
- Assigned app visibility based on user profiles

🔹 Record Pages

- Customized **Ticket__c Record Page** to show:
  - Event details

    - Attendee information
    - Related lists (e.g., reminders sent)
- Added dynamic components based on user role

🔹 **Tabs & Navigation**

- Created custom tabs for:
    - Ticket__c
    - Attendee__c
    - Dashboard (custom report view)
- Tabs are grouped logically for ease of access

🔹 **Home Page Layouts**

- Designed a custom Home Page for organizers with:
    - Upcoming Events list
    - Quick links to create tickets
    - Dashboard summary

🔹 **Utility Bar**

- Added a Utility Bar with:
    - Quick Ticket Search
    - Flow Launcher
    - Help & Documentation link

⚡ Lightning Web Components (LWC)

🔹 Component: TicketViewer

- Built a custom LWC named TicketViewer
- Displays ticket details in a clean, responsive format
- Uses modular design for reusability

🔹 Apex with LWC

- Integrated Apex methods to fetch ticket data dynamically
- Used @AuraEnabled methods for backend logic

Example:

```
@AuraEnabled
public static Ticket__c getTicketDetails(Id ticketId) {
  return [SELECT Subject__c, Event_Date__c FROM Ticket__c WHERE Id = :ticketId];
}
```

🔹 Events in LWC

- Used **custom events** to pass data between components
- Example: Ticket selection → detail view update

🔹 Wire Adapters

- Used @wire to bind Apex data to LWC templates
- Enables reactive UI updates based on record changes

- **Imperative Apex Calls**
  - Used for on-demand data fetching triggered by user actions
  - Example: “View Ticket Details” button

🔵 Phase 7: Integration & External Access.

✅ Integration Components Implemented

1. Remote Site Settings

Configured Remote Site Settings to whitelist external endpoints for future REST callouts.

- **Example**:
  - Name: AttendSync_ReminderAPI
  - URL: https://api.reminderservice.com

---

2. API Limits Monitoring

Monitored API usage via Setup → System Overview to ensure Apex callouts and future integrations stay within governor limits.

---

3. Named Credentials

✅ *Explored but not implemented*
We reviewed Named Credentials for securely storing endpoint URLs and authentication tokens, but did not use them in the current version.

---

4. Web Services (REST/SOAP)

✅ *Planned but not implemented*
AttendSync is designed to support RESTful callouts for sending ticket data externally, but no live callouts were built in this version.

---

5. Callouts in Apex

✅ *Reviewed but not used*
We reviewed how to structure Apex callouts using HttpRequest and HttpResponse, but no callout logic was deployed.

---

6. OAuth & Authentication

✅ *Reviewed but not configured*
We explored OAuth setup for future integrations requiring token-based access, but did not create a Connected App in this version.

.

🔵 Phase 8: Data Management & Deployment

✅ Components Implemented

1. Data Import Wizard

Used to import sample attendee and ticket records during initial testing.

---

2. Data Loader

Used for bulk creation of Ticket__c records during flow testing.

---

3. Change Sets

Used to deploy flows, custom objects, and email templates from sandbox to production-ready org.

---

4. VS Code & SFDX

Used for Apex class development, LWC components, and metadata tracking.

❌ Components Not Implemented

- **Duplicate Rules**: Not configured
- **Data Export & Backup**: Not performed
- **Unmanaged vs Managed Packages**: Not created
- **ANT Migration Tool**: Not used.

🔵 Phase 9: Reporting, Dashboards & Security Review

✅ Components Implemented

1. Reports (Tabular)

Created tabular reports to list upcoming events and confirmed tickets.

---

2. Report Types

Used standard report types for custom objects (Ticket__c, Attendee__c).

---

3. Sharing Settings

Configured OWD for Ticket__c as Private and created sharing rules for organizers.

---

4. Field Level Security

Reviewed and configured field visibility for profiles (Organizer, Attendee).

---

5. Audit Trail

Used Setup → View Setup Audit Trail to monitor configuration changes.

🔵 Phase 10: Final Presentation & Demo Day

---

1. Pitch Presentation

Prepared a problem-solution overview highlighting:

- Manual event management challenges
- AttendSync’s automation flow
- Email reminder logic and UI components

---

2. Demo Walkthrough

Conducted a live walkthrough showing:

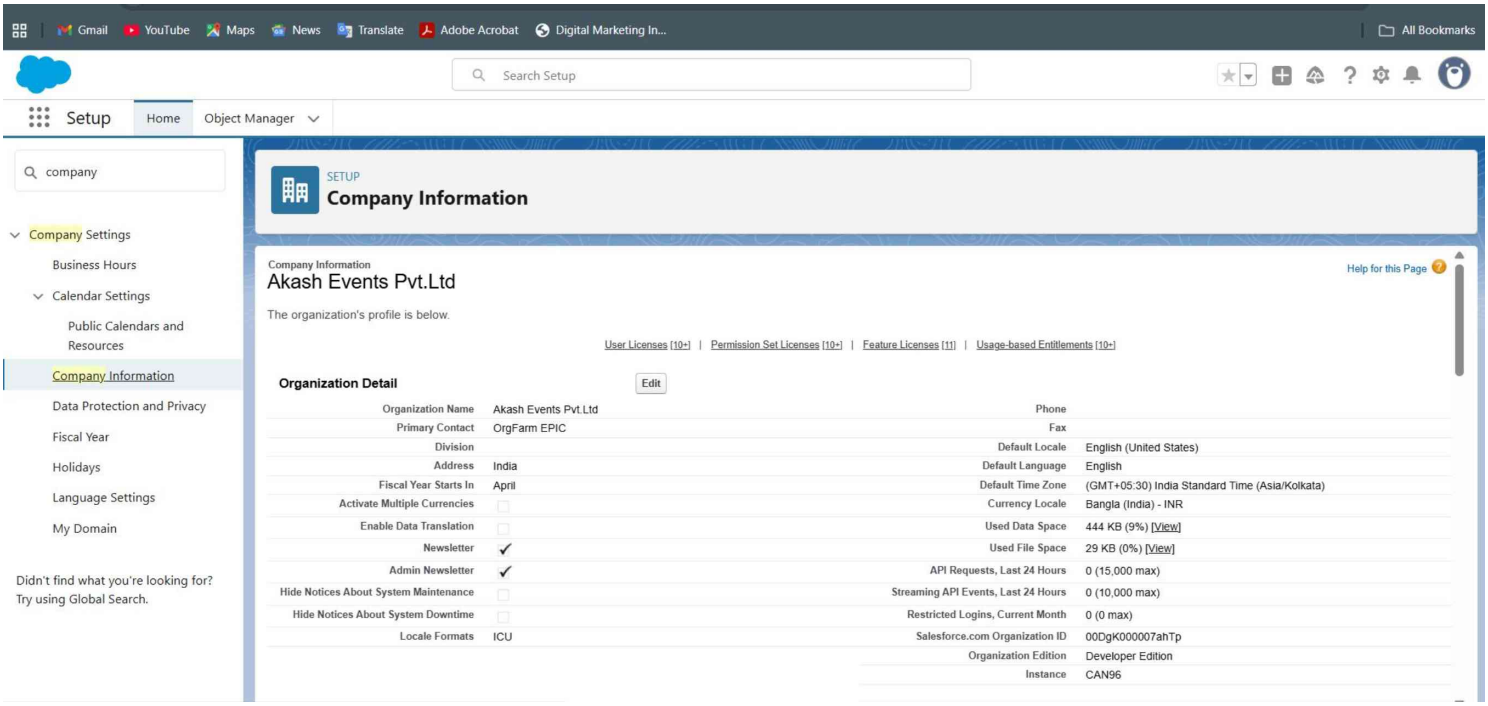
- Ticket creation
- Flow execution
- Email delivery
- Apex job monitoring

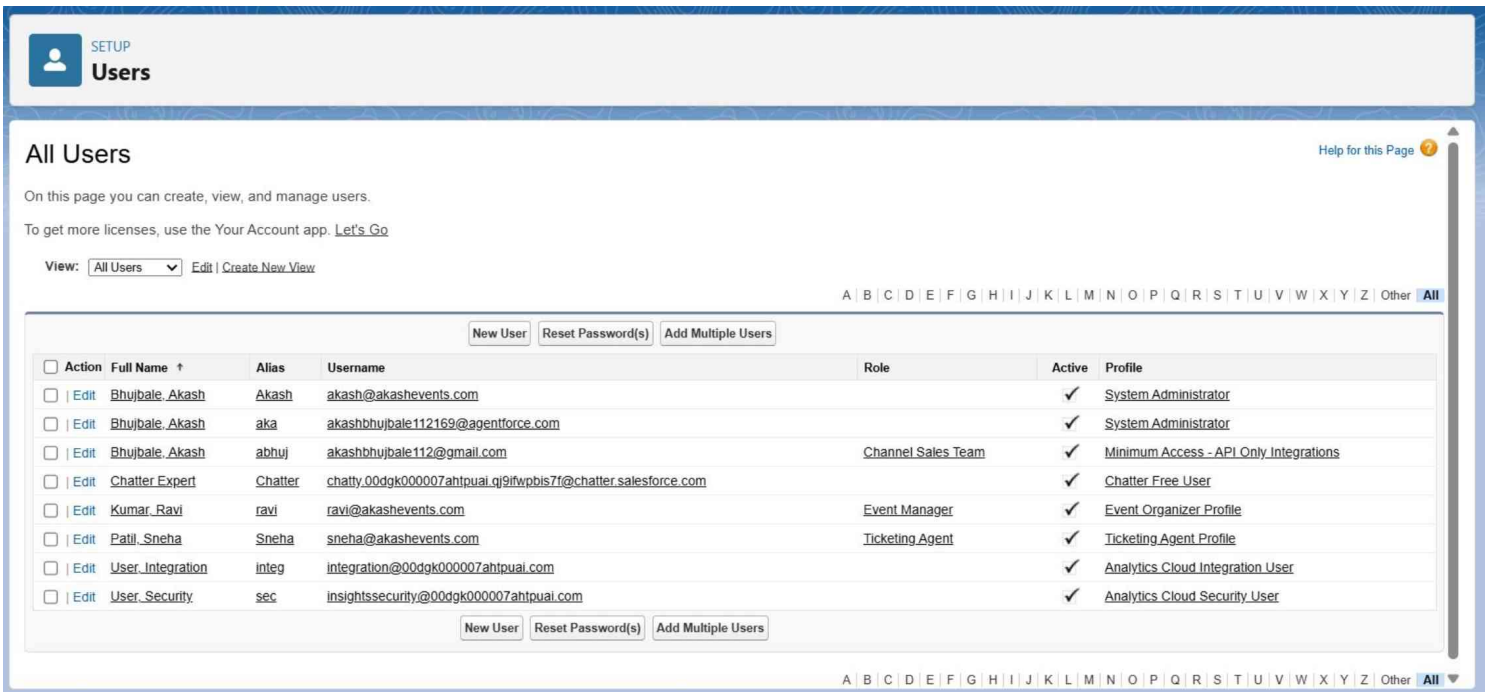
---

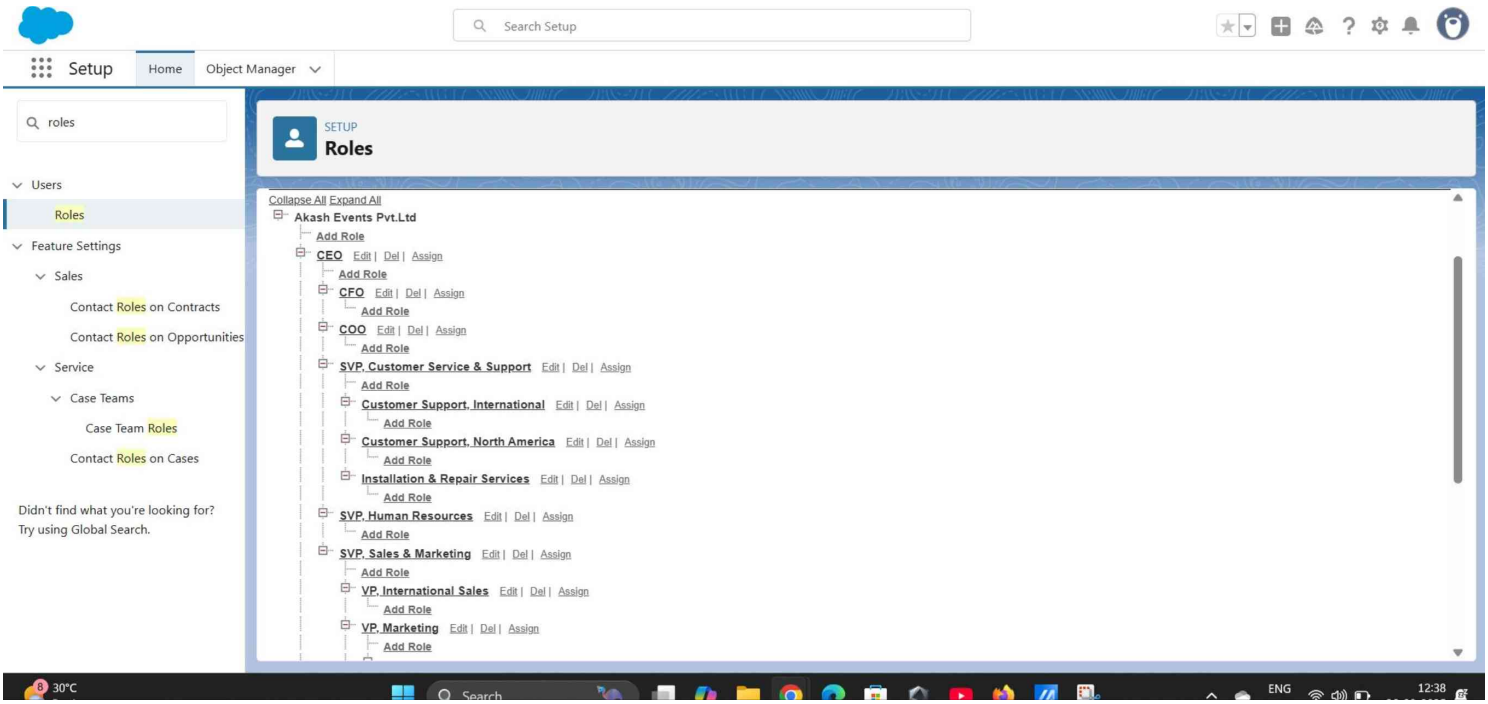
3. Handoff Documentation

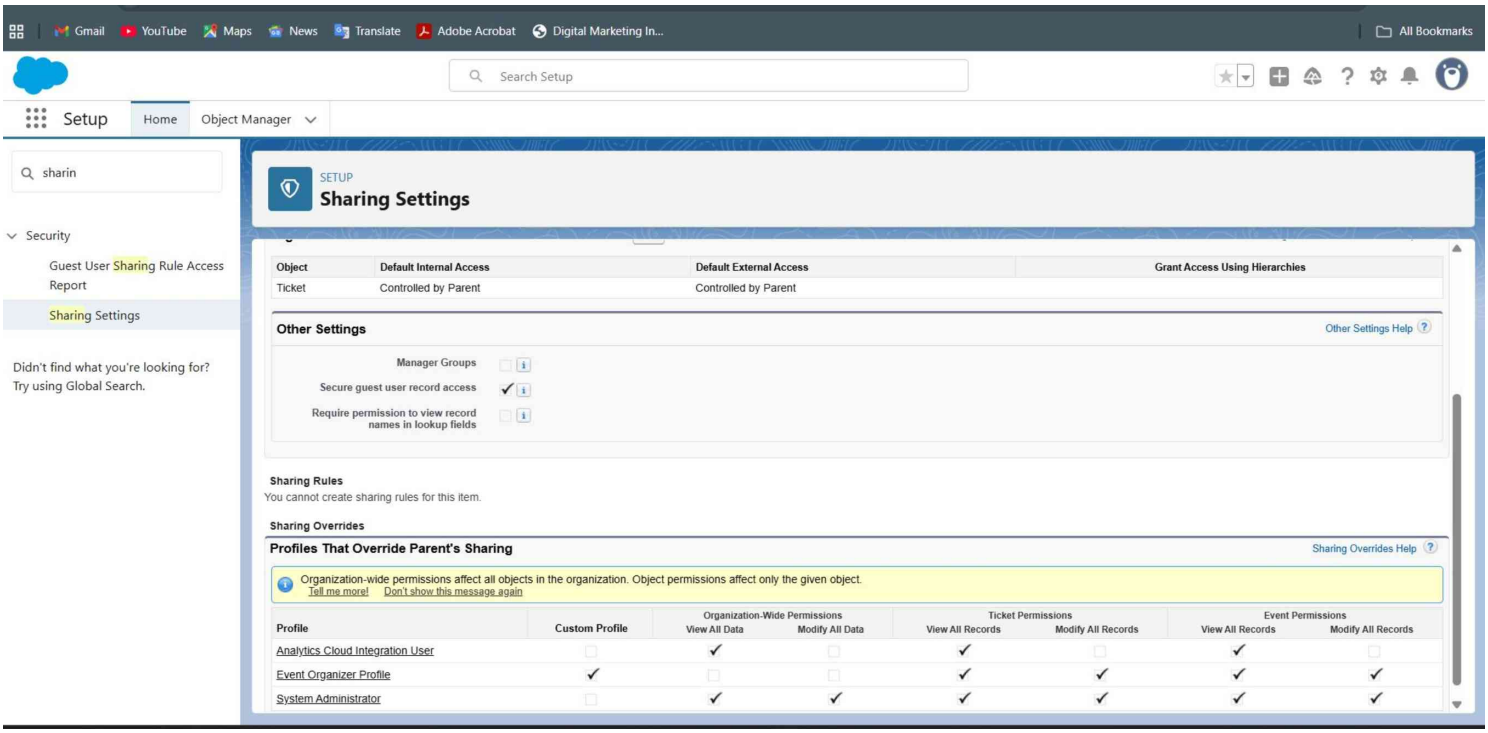
Created full documentation across all phases, including:

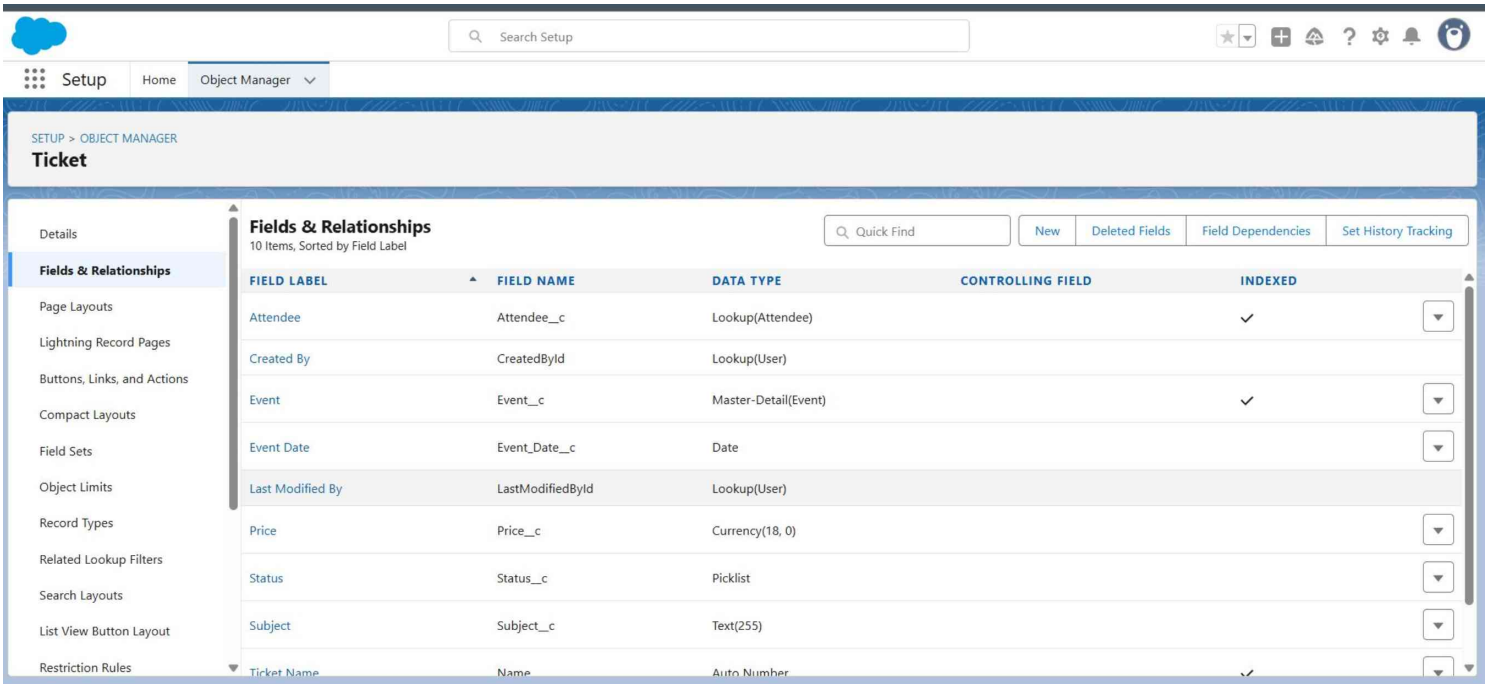
- Flow logic
- Apex classes
- UI components
- Deployment steps

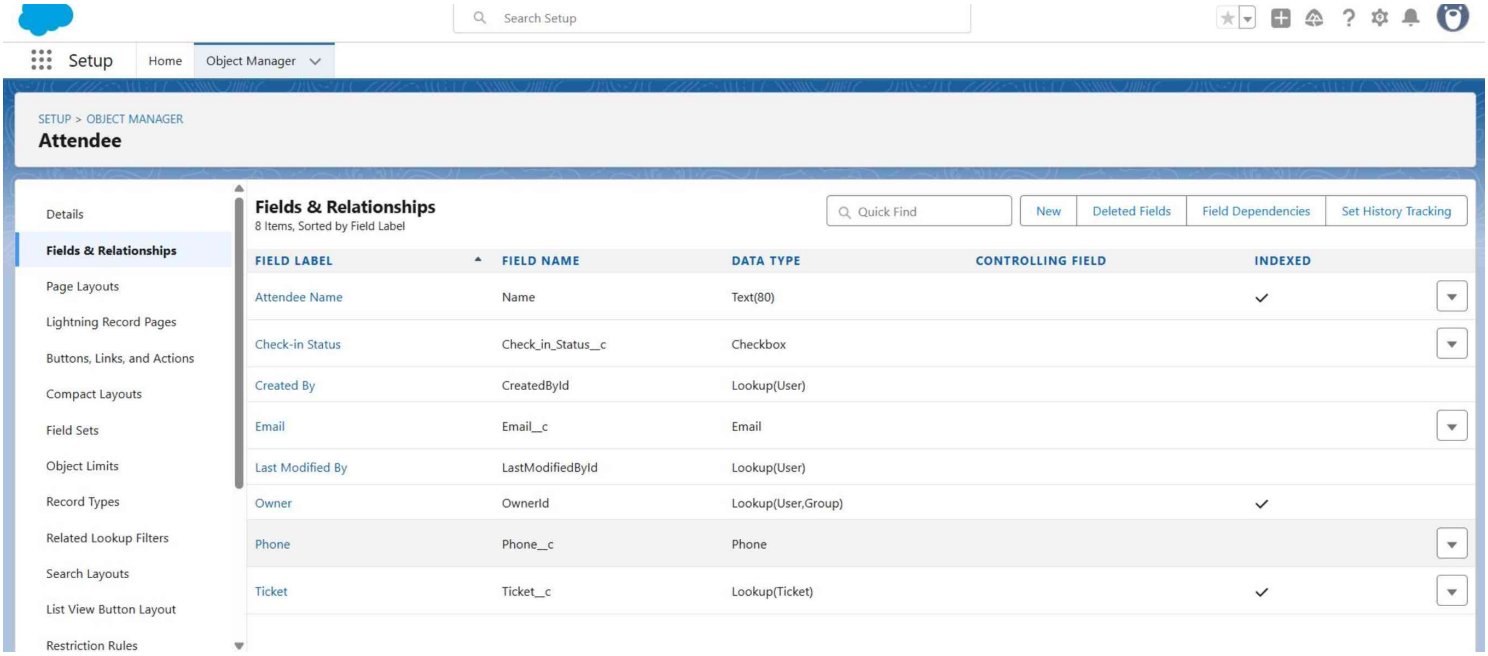

Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search Setup
Setup
Home
Object Manager
company
Company Settings
Business Hours
Calendar Settings
Public Calendars and Resources
Company Information
Data Protection and Privacy
Fiscal Year
Holidays
Language Settings
My Domain
Didn't find what you're looking for?
Try using Global Search.
SETUP
Company Information
Company Information
Akash Events Pvt.Ltd
Help for this Page
The organization's profile is below.
User Licenses [10+] | Permission Set Licenses [10+] | Feature Licenses [11] | Usage-based Entitlements [10+]
Organization Detail
Edit
Organization Name
Akash Events Pvt.Ltd
Primary Contact
OrgFarm EPIC
Division
Address
India
Fiscal Year Starts In
April
Activate Multiple Currencies
Enable Data Translation
Newsletter
Admin Newsletter
Hide Notices About System Maintenance
Hide Notices About System Downtime
Locale Formats
ICU
Phone
Fax
Default Locale
English (United States)
Default Language
English
Default Time Zone
(GMT+05:30) India Standard Time (Asia/Kolkata)
Currency Locale
Bangla (India) - INR
Used Data Space
444 KB (9%) [View]
Used File Space
29 KB (0%) [View]
API Requests, Last 24 Hours
0 (15,000 max)
Streaming API Events, Last 24 Hours
0 (10,000 max)
Restricted Logins, Current Month
0 (0 max)
Salesforce.com Organization ID
00DgK000007ahTp
Organization Edition
Developer Edition
Instance
CAN96

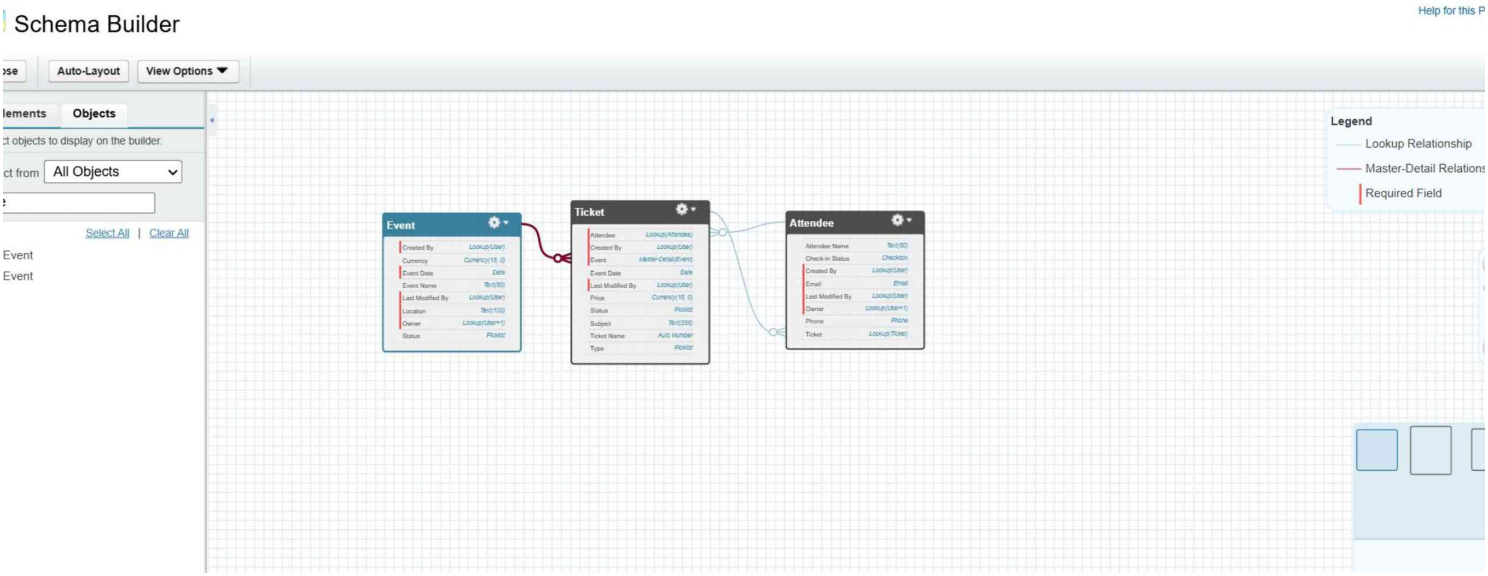

SETUP
Users
All Users
Help for this Page
On this page you can create, view, and manage users.
To get more licenses, use the Your Account app. Let's Go
View: All Users Edit | Create New View
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
New User Reset Password(s) Add Multiple Users
Action | Full Name | Alias | Username | Role | Active | Profile
Edit | Bhujbale, Akash | Akash | akash@akashevents.com | | ✓ | System Administrator
Edit | Bhujbale, Akash | aka | akashbhujbale112169@agentforce.com | | ✓ | System Administrator
Edit | Bhujbale, Akash | abhuj | akashbhujbale112@gmail.com | Channel Sales Team | ✓ | Minimum Access - API Only Integrations
Edit | Chatter Expert | Chatter | chatty.00dgk000007ahtpuai.qj9ifwpbis7f@chatter.salesforce.com | | ✓ | Chatter Free User
Edit | Kumar, Ravi | ravi | ravi@akashevents.com | Event Manager | ✓ | Event Organizer Profile
Edit | Patil, Sneha | Sneha | sneha@akashevents.com | Ticketing Agent | ✓ | Ticketing Agent Profile
Edit | User, Integration | integ | integration@00dgk000007ahtpuai.com | | ✓ | Analytics Cloud Integration User
Edit | User, Security | sec | insightssecurity@00dgk000007ahtpuai.com | | ✓ | Analytics Cloud Security User
New User Reset Password(s) Add Multiple Users
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

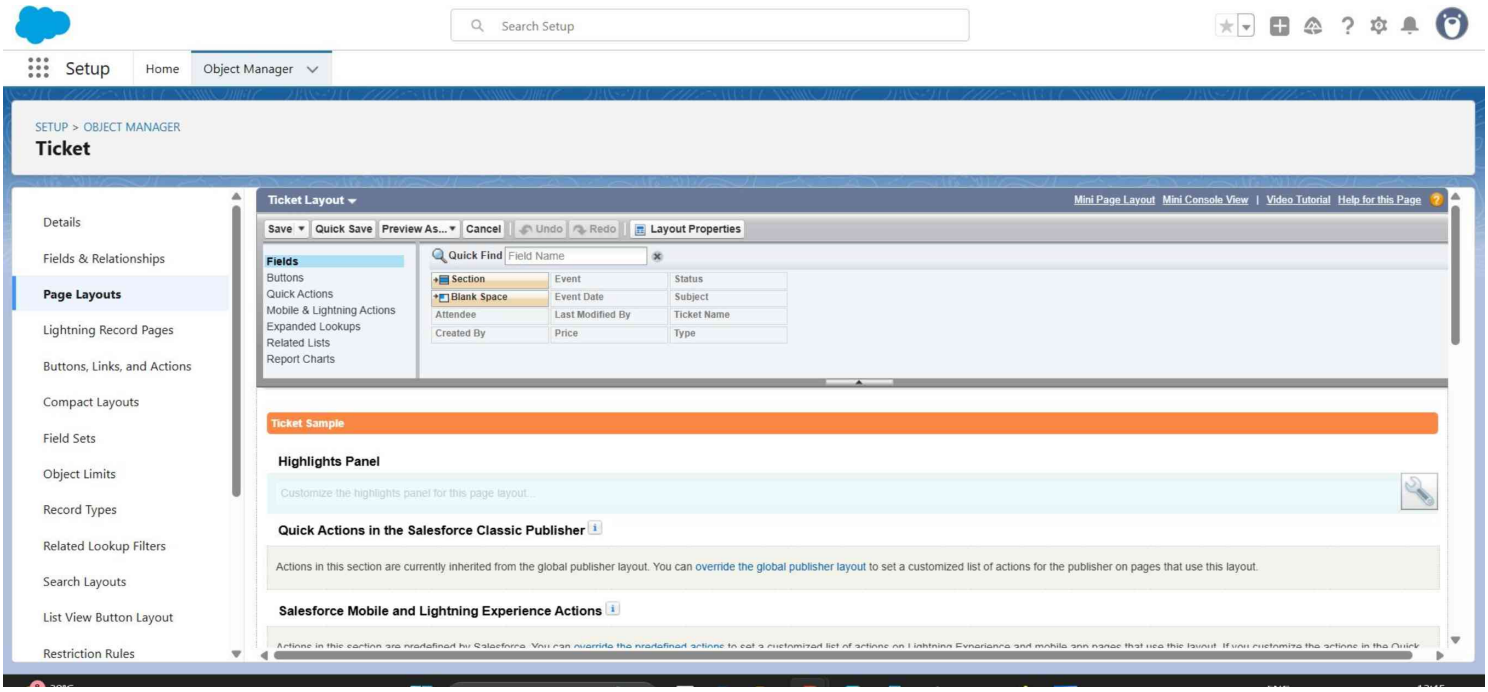

Search Setup
Setup
Home
Object Manager
roles
Users
Roles
Feature Settings
Sales
Contact Roles on Contracts
Contact Roles on Opportunities
Service
Case Teams
Case Team Roles
Contact Roles on Cases
Didn't find what you're looking for?
Try using Global Search.
SETUP
Roles
Collapse All Expand All
Akash Events Pvt.Ltd
Add Role
CEO Edit | Del | Assign
Add Role
CFO Edit | Del | Assign
Add Role
COO Edit | Del | Assign
Add Role
SVP, Customer Service & Support Edit | Del | Assign
Add Role
Customer Support, International Edit | Del | Assign
Add Role
Customer Support, North America Edit | Del | Assign
Add Role
Installation & Repair Services Edit | Del | Assign
Add Role
SVP, Human Resources Edit | Del | Assign
Add Role
SVP, Sales & Marketing Edit | Del | Assign
Add Role
VP, International Sales Edit | Del | Assign
Add Role
VP, Marketing Edit | Del | Assign
Add Role
30°C
Search
ENG
12:38

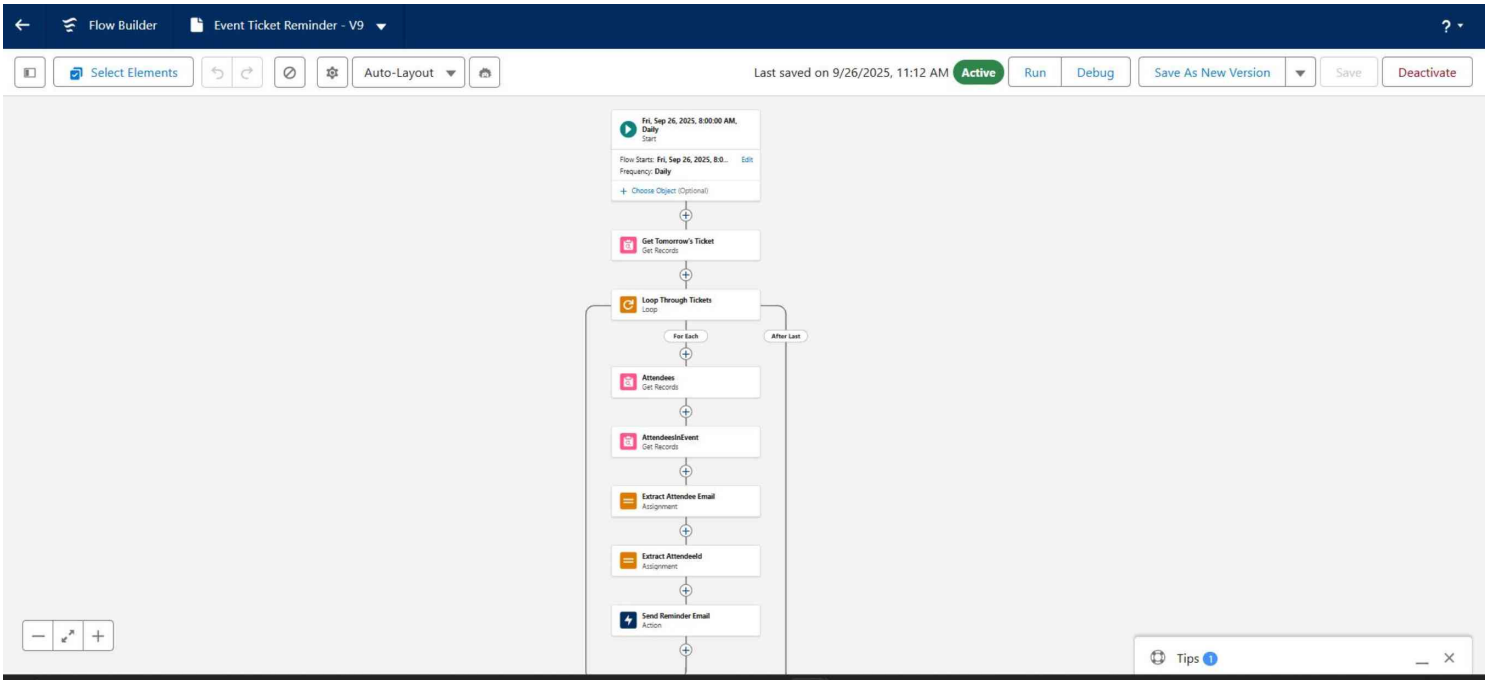

Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search Setup
Setup
Home
Object Manager
sharin
Security
Guest User Sharing Rule Access Report
Sharing Settings
Didn't find what you're looking for?
Try using Global Search.
SETUP
Sharing Settings
Object
Default Internal Access
Default External Access
Grant Access Using Hierarchies
Ticket
Controlled by Parent
Controlled by Parent
Other Settings
Other Settings Help
Manager Groups
Secure guest user record access
Require permission to view record names in lookup fields
Sharing Rules
You cannot create sharing rules for this item.
Sharing Overrides
Profiles That Override Parent's Sharing
Sharing Overrides Help
Organization-wide permissions affect all objects in the organization. Object permissions affect only the given object.
Tell me more!
Don't show this message again
Profile
Custom Profile
Organization-Wide Permissions
View All Data
Modify All Data
Ticket Permissions
View All Records
Modify All Records
Event Permissions
View All Records
Modify All Records
Analytics Cloud Integration User
Event Organizer Profile
System Administrator

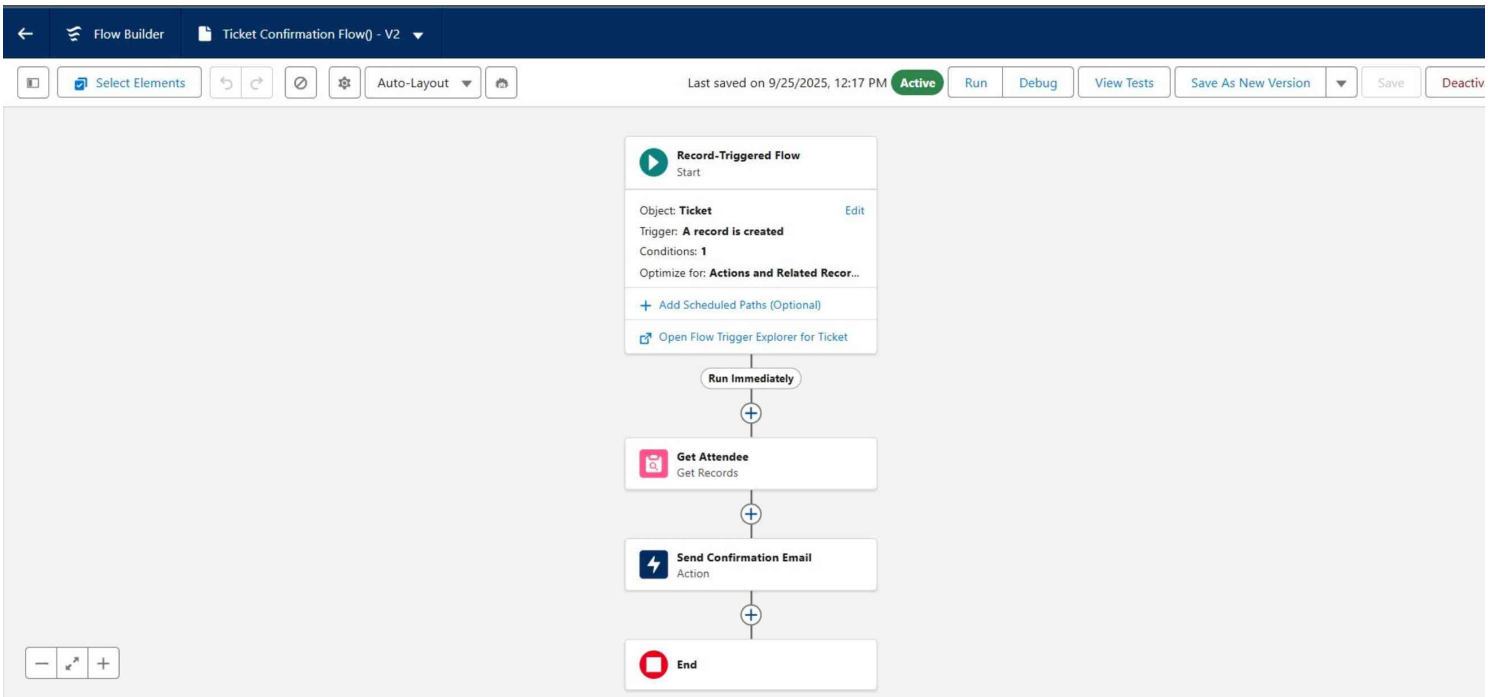

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Ticket
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Fields & Relationships
10 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Attendee
Attendee__c
Lookup(Attendee)
Created By
CreatedById
Lookup(User)
Event
Event__c
Master-Detail(Event)
Event Date
Event_Date__c
Date
Last Modified By
LastModifiedById
Lookup(User)
Price
Price__c
Currency(18, 0)
Status
Status__c
Picklist
Subject
Subject__c
Text(255)
Ticket Name
Name
Auto Number

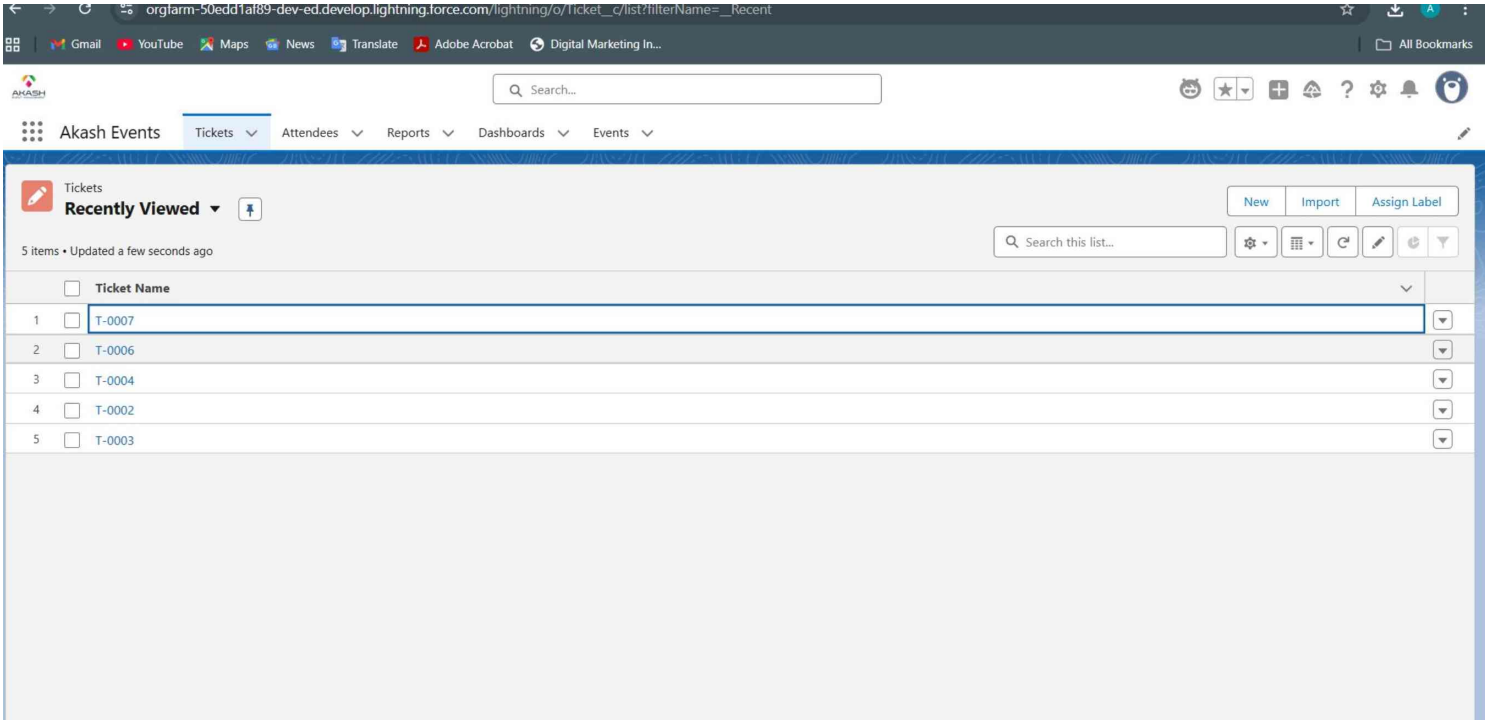

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Attendee
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Fields & Relationships
8 Items, Sorted by Field Label
Quick Find
New
Deleted Fields
Field Dependencies
Set History Tracking
FIELD LABEL
FIELD NAME
DATA TYPE
CONTROLLING FIELD
INDEXED
Attendee Name
Name
Text(80)
Check-in Status
Check_in_Status__c
Checkbox
Created By
CreatedById
Lookup(User)
Email
Email__c
Email
Last Modified By
LastModifiedById
Lookup(User)
Owner
OwnerId
Lookup(User,Group)
Phone
Phone__c
Phone
Ticket
Ticket__c
Lookup(Ticket)

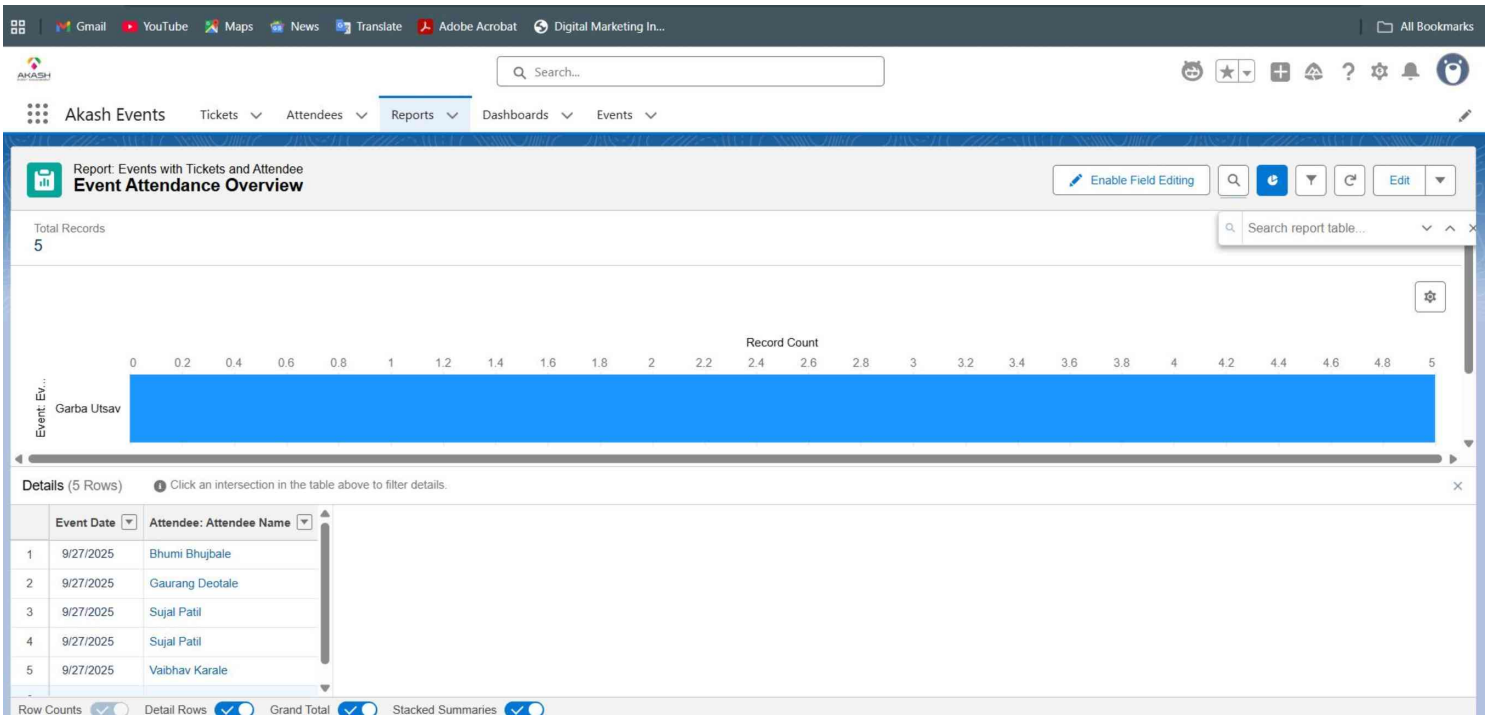

Schema Builder
Help for this P
Auto-Layout
View Options
Objects
Select All | Clear All
Event
Event
All Objects
Legend
Lookup Relationship
Master-Detail Relations
Required Field
Event
Ticket
Attendee

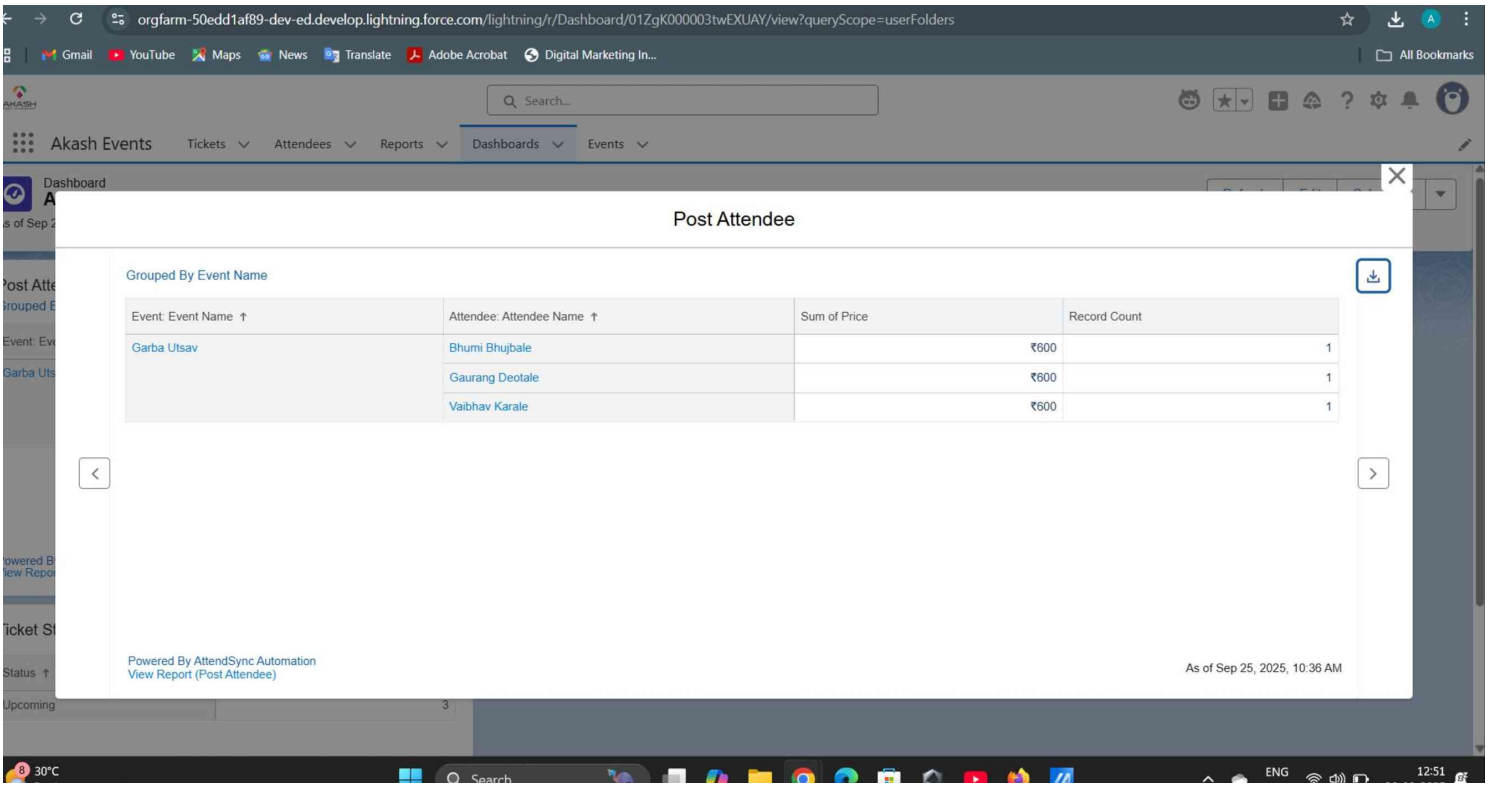

Search Setup
Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Ticket
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Ticket Layout
Mini Page Layout
Mini Console View
Video Tutorial
Help for this Page
Save
Quick Save
Preview As...
Cancel
Undo
Redo
Layout Properties
Fields
Buttons
Quick Actions
Mobile & Lightning Actions
Expanded Lookups
Related Lists
Report Charts
Quick Find
Field Name
Section
Blank Space
Event
Status
Event Date
Subject
Attendee
Last Modified By
Ticket Name
Created By
Price
Type
Ticket Sample
Highlights Panel
Customize the highlights panel for this page layout...
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

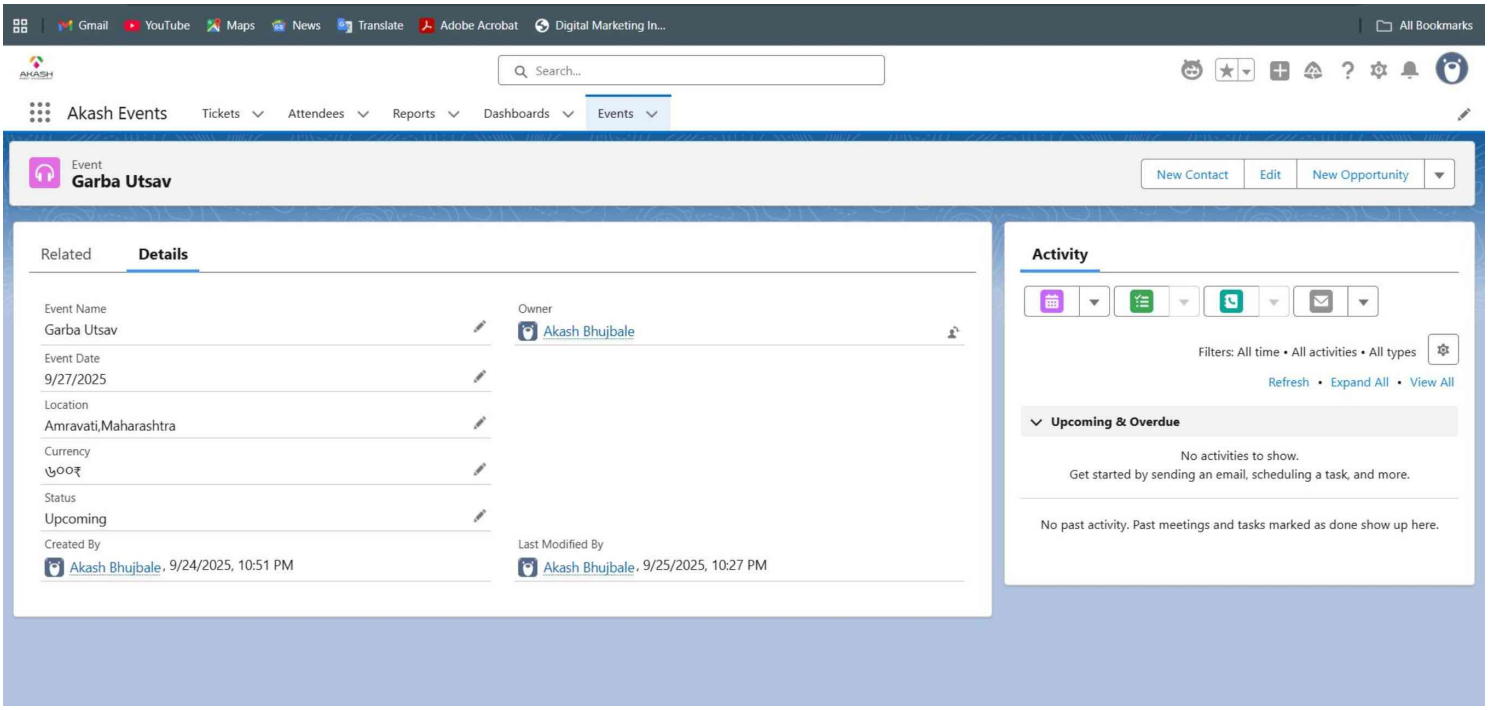

Flow Builder
Event Ticket Reminder - V9
Select Elements
Auto-Layout
Last saved on 9/26/2025, 11:12 AM
Active
Run
Debug
Save As New Version
Save
Deactivate
Fri, Sep 26, 2025, 8:00:00 AM, Daily
Start
Flow Starts: Fri, Sep 26, 2025, 8:0...
Edit
Frequency: Daily
Choose Object (Optional)
Get Tomorrow's Ticket
Get Records
Loop Through Tickets
Loop
For Each
After Last
Attendees
Get Records
AttendeesInEvent
Get Records
Extract Attendee Email
Assignment
Extract AttendeeId
Assignment
Send Reminder Email
Action
Tips
1

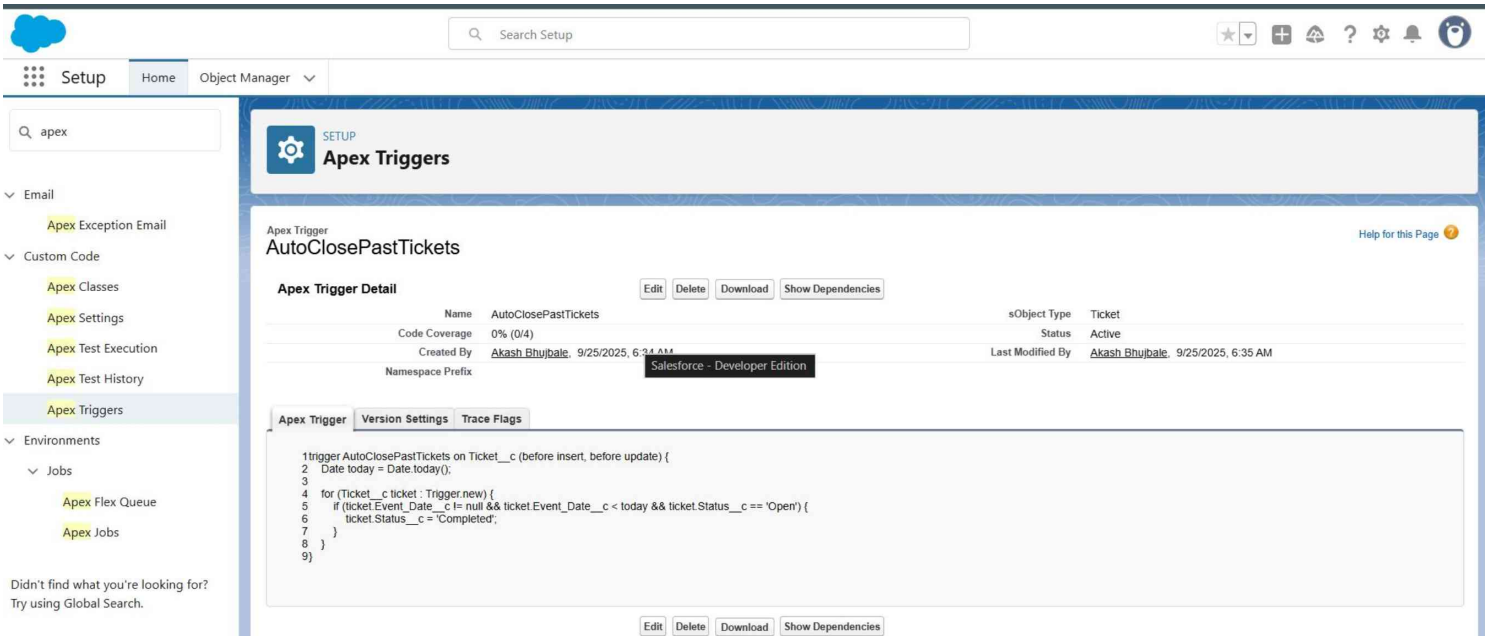

Flow Builder
Ticket Confirmation Flow() - V2
Select Elements
Auto-Layout
Last saved on 9/25/2025, 12:17 PM
Active
Run
Debug
View Tests
Save As New Version
Save
Deactiv
Record-Triggered Flow
Start
Object: Ticket
Edit
Trigger: A record is created
Conditions: 1
Optimize for: Actions and Related Recor...
Add Scheduled Paths (Optional)
Open Flow Trigger Explorer for Ticket
Run Immediately
Get Attendee
Get Records
Send Confirmation Email
Action
End

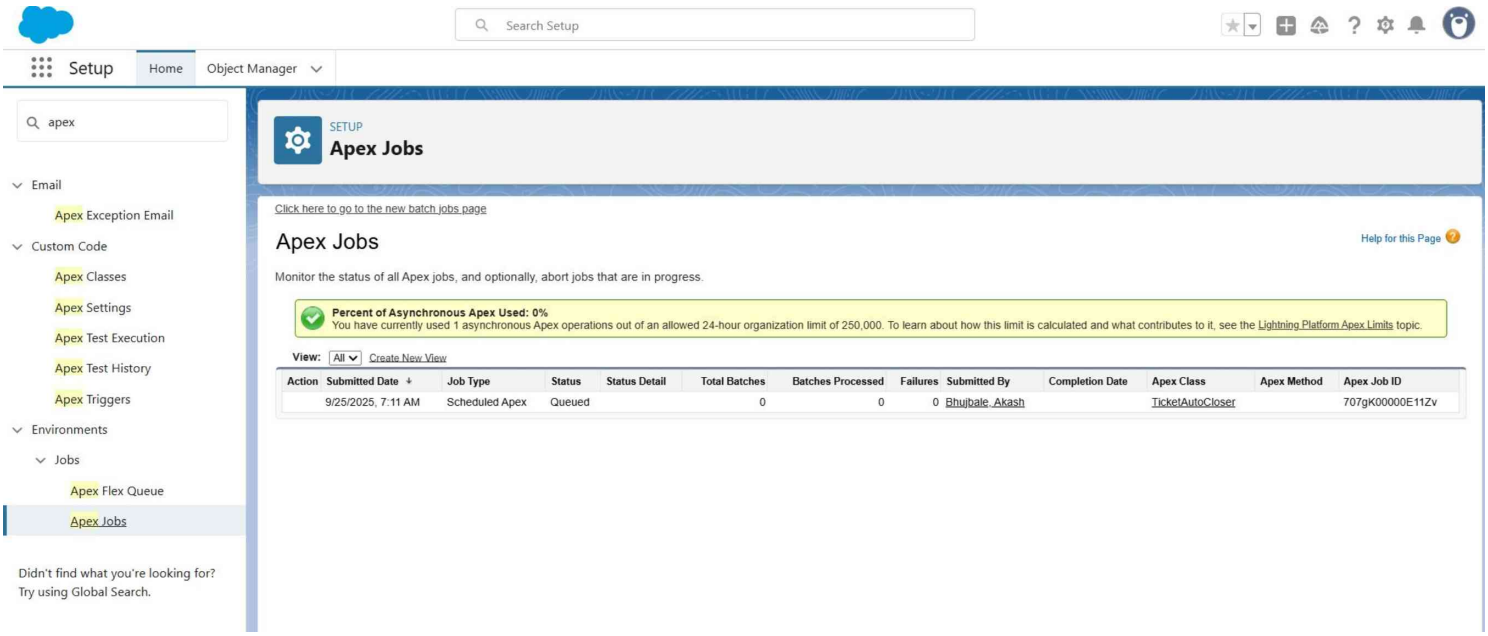

orgfarm-50edd1af89-dev-ed.develop.lightning.force.com/lightning/o/Ticket__c/list?filterName=__Recent
Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search...
Akash Events
Tickets
Attendees
Reports
Dashboards
Events
Tickets
Recently Viewed
New
Import
Assign Label
5 items • Updated a few seconds ago
Search this list...
Ticket Name
1 T-0007
2 T-0006
3 T-0004
4 T-0002
5 T-0003

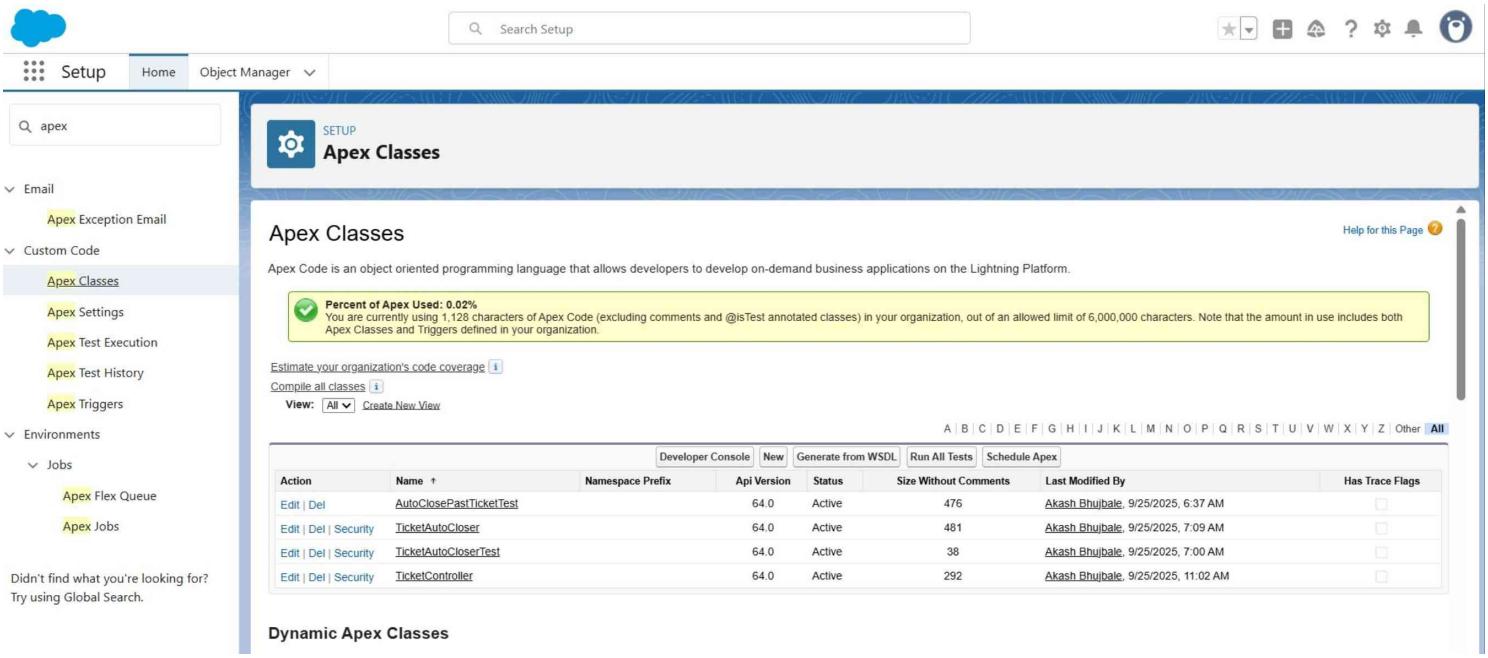

Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search...
Akash Events
Tickets
Attendees
Reports
Dashboards
Events
Report: Events with Tickets and Attendee
Event Attendance Overview
Enable Field Editing
Edit
Total Records
5
Search report table...
Record Count
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2 2.2 2.4 2.6 2.8 3 3.2 3.4 3.6 3.8 4 4.2 4.4 4.6 4.8 5
Event: Ev...
Garba Utsav
Details (5 Rows)
Click an intersection in the table above to filter details.
Event Date
Attendee: Attendee Name
1 9/27/2025 Bhumi Bhujbale
2 9/27/2025 Gaurang Deotale
3 9/27/2025 Sujal Patil
4 9/27/2025 Sujal Patil
5 9/27/2025 Vaibhav Karale
Row Counts
Detail Rows
Grand Total
Stacked Summaries

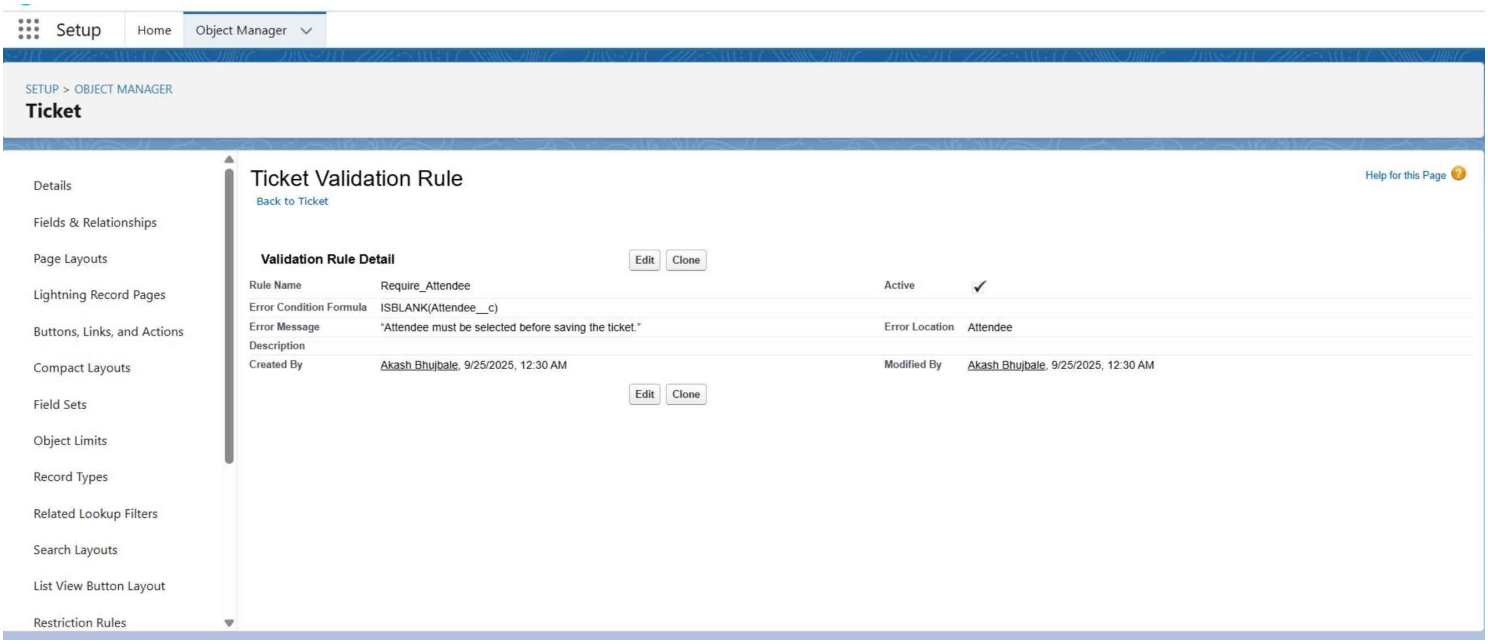

orgfarm-50edd1af89-dev-ed.develop.lightning.force.com/lightning/r/Dashboard/01ZgK000003twEXUAY/view?queryScope=userFolders
Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search...
Akash Events
Tickets
Attendees
Reports
Dashboards
Events
Dashboard
Post Attendee
Grouped By Event Name
Event: Event Name
Attendee: Attendee Name
Sum of Price
Record Count
Garba Utsav
Bhumi Bhujbale
₹600
1
Gaurang Deotale
₹600
1
Vaibhav Karale
₹600
1
Powered By AttendSync Automation
View Report (Post Attendee)
As of Sep 25, 2025, 10:36 AM
30°C
ENG
12:51

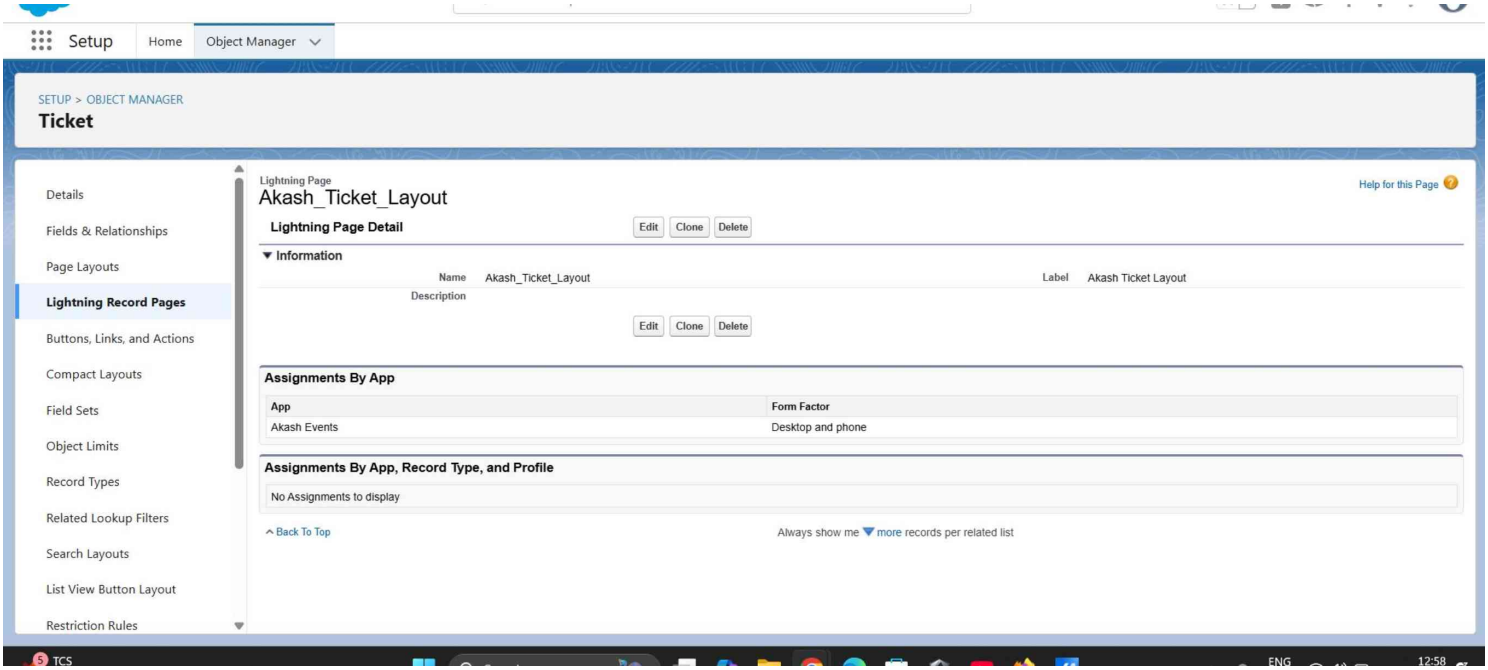

Gmail
YouTube
Maps
News
Translate
Adobe Acrobat
Digital Marketing In...
All Bookmarks
Search...
Akash Events
Tickets
Attendees
Reports
Dashboards
Events
Event
Garba Utsav
New Contact
Edit
New Opportunity
Related
Details
Event Name
Garba Utsav
Owner
Akash Bhujbale
Event Date
9/27/2025
Location
Amravati,Maharashtra
Currency
৬০০₹
Status
Upcoming
Created By
Akash Bhujbale, 9/24/2025, 10:51 PM
Last Modified By
Akash Bhujbale, 9/25/2025, 10:27 PM
Activity
Filters: All time • All activities • All types
Refresh • Expand All • View All
Upcoming & Overdue
No activities to show.
Get started by sending an email, scheduling a task, and more.
No past activity. Past meetings and tasks marked as done show up here.

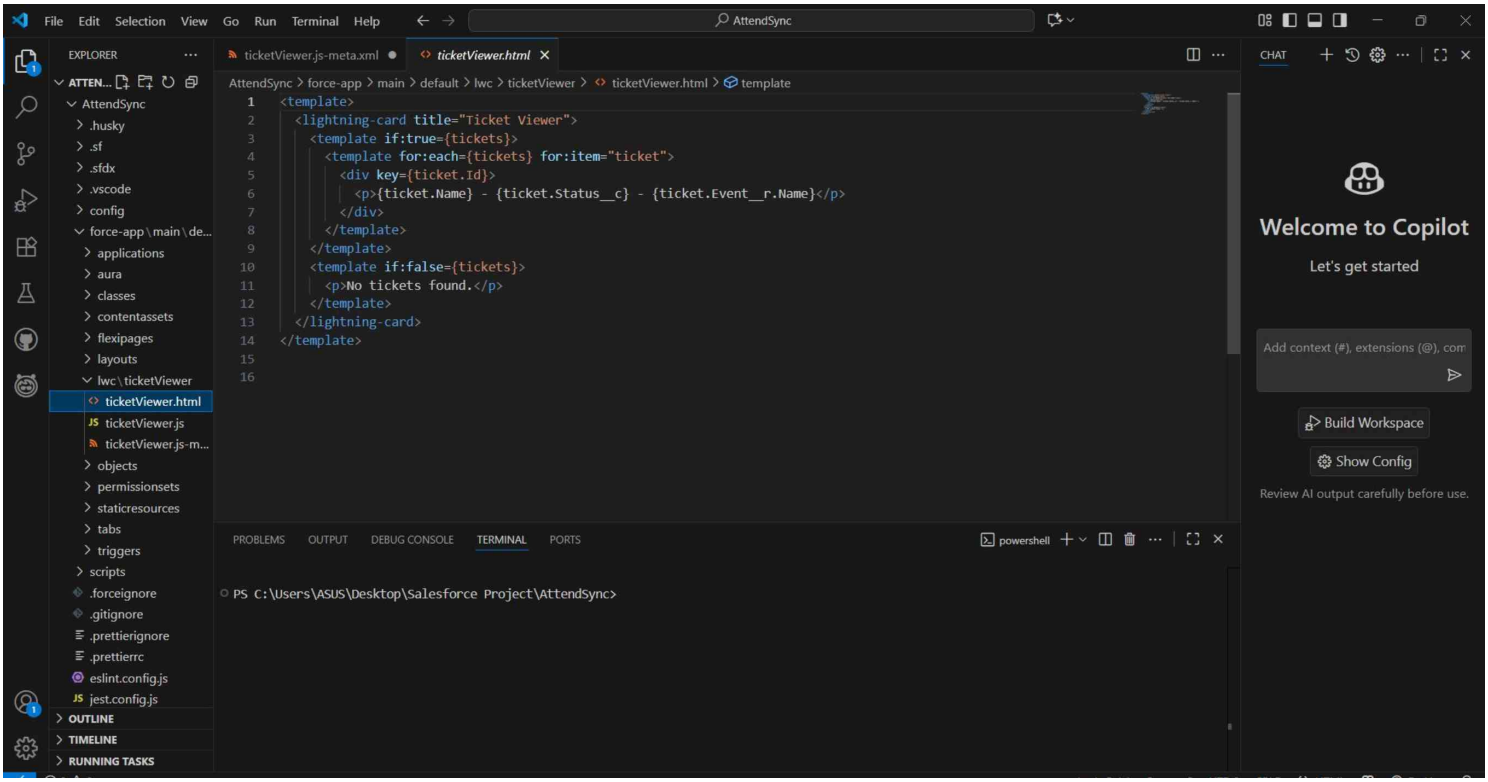

Search Setup
Setup
Home
Object Manager
apex
Email
Apex Exception Email
Custom Code
Apex Classes
Apex Settings
Apex Test Execution
Apex Test History
Apex Triggers
Environments
Jobs
Apex Flex Queue
Apex Jobs
Didn't find what you're looking for?
Try using Global Search.
SETUP
Apex Triggers
Apex Trigger
AutoClosePastTickets
Help for this Page
Apex Trigger Detail
Edit
Delete
Download
Show Dependencies
Name
AutoClosePastTickets
sObject Type
Ticket
Code Coverage
0% (0/4)
Status
Active
Created By
Akash Bhujbale, 9/25/2025, 6:34 AM
Last Modified By
Akash Bhujbale, 9/25/2025, 6:35 AM
Namespace Prefix
Salesforce - Developer Edition
Apex Trigger
Version Settings
Trace Flags
1trigger AutoClosePastTickets on Ticket__c (before insert, before update) {
2 Date today = Date.today();
3
4 for (Ticket__c ticket : Trigger.new) {
5 if (ticket.Event_Date__c != null && ticket.Event_Date__c < today && ticket.Status__c == 'Open') {
6 ticket.Status__c = 'Completed';
7 }
8 }
9}
Edit
Delete
Download
Show Dependencies


Search Setup
Setup
Home
Object Manager
apex
Email
Apex Exception Email
Custom Code
Apex Classes
Apex Settings
Apex Test Execution
Apex Test History
Apex Triggers
Environments
Jobs
Apex Flex Queue
Apex Jobs
Didn't find what you're looking for?
Try using Global Search.
SETUP
Apex Jobs
Click here to go to the new batch jobs page
Apex Jobs
Help for this Page
Monitor the status of all Apex jobs, and optionally, abort jobs that are in progress.
Percent of Asynchronous Apex Used: 0%
You have currently used 1 asynchronous Apex operations out of an allowed 24-hour organization limit of 250,000. To learn about how this limit is calculated and what contributes to it, see the Lightning Platform Apex Limits topic.
View: All
Create New View
Action
Submitted Date
Job Type
Status
Status Detail
Total Batches
Batches Processed
Failures
Submitted By
Completion Date
Apex Class
Apex Method
Apex Job ID
9/25/2025, 7:11 AM
Scheduled Apex
Queued
0
0
0
Bhujbale, Akash
TicketAutoCloser
707gK00000E11Zv


Search Setup
Setup
Home
Object Manager
apex
Email
Apex Exception Email
Custom Code
Apex Classes
Apex Settings
Apex Test Execution
Apex Test History
Apex Triggers
Environments
Jobs
Apex Flex Queue
Apex Jobs
Didn't find what you're looking for?
Try using Global Search.
SETUP
Apex Classes
Apex Classes
Help for this Page
Apex Code is an object oriented programming language that allows developers to develop on-demand business applications on the Lightning Platform.
Percent of Apex Used: 0.02%
You are currently using 1,128 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.
Estimate your organization's code coverage
Compile all classes
View: All Create New View
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
Developer Console
New
Generate from WSDL
Run All Tests
Schedule Apex
Action
Name
Namespace Prefix
Api Version
Status
Size Without Comments
Last Modified By
Has Trace Flags
Edit | Del
AutoClosePastTicketTest
64.0
Active
476
Akash Bhujbale, 9/25/2025, 6:37 AM
Edit | Del | Security
TicketAutoCloser
64.0
Active
481
Akash Bhujbale, 9/25/2025, 7:09 AM
Edit | Del | Security
TicketAutoCloserTest
64.0
Active
38
Akash Bhujbale, 9/25/2025, 7:00 AM
Edit | Del | Security
TicketController
64.0
Active
292
Akash Bhujbale, 9/25/2025, 11:02 AM
Dynamic Apex Classes

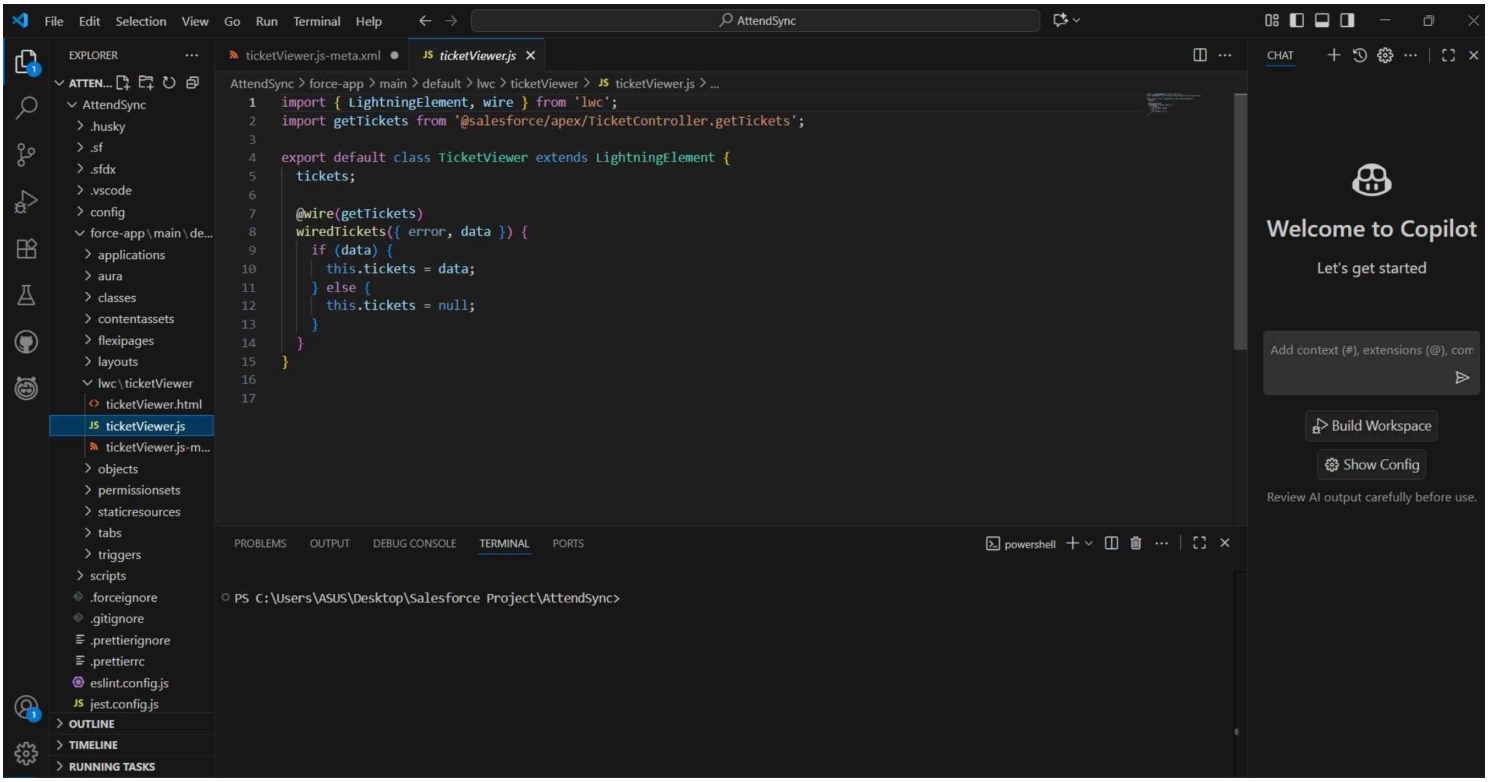

Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Ticket
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Ticket Validation Rule
Back to Ticket
Help for this Page
Validation Rule Detail
Edit
Clone
Rule Name
Require_Attendee
Active
Error Condition Formula
ISBLANK(Attendee__c)
Error Message
"Attendee must be selected before saving the ticket."
Error Location
Attendee
Description
Created By
Akash Bhujbale, 9/25/2025, 12:30 AM
Modified By
Akash Bhujbale, 9/25/2025, 12:30 AM
Edit
Clone

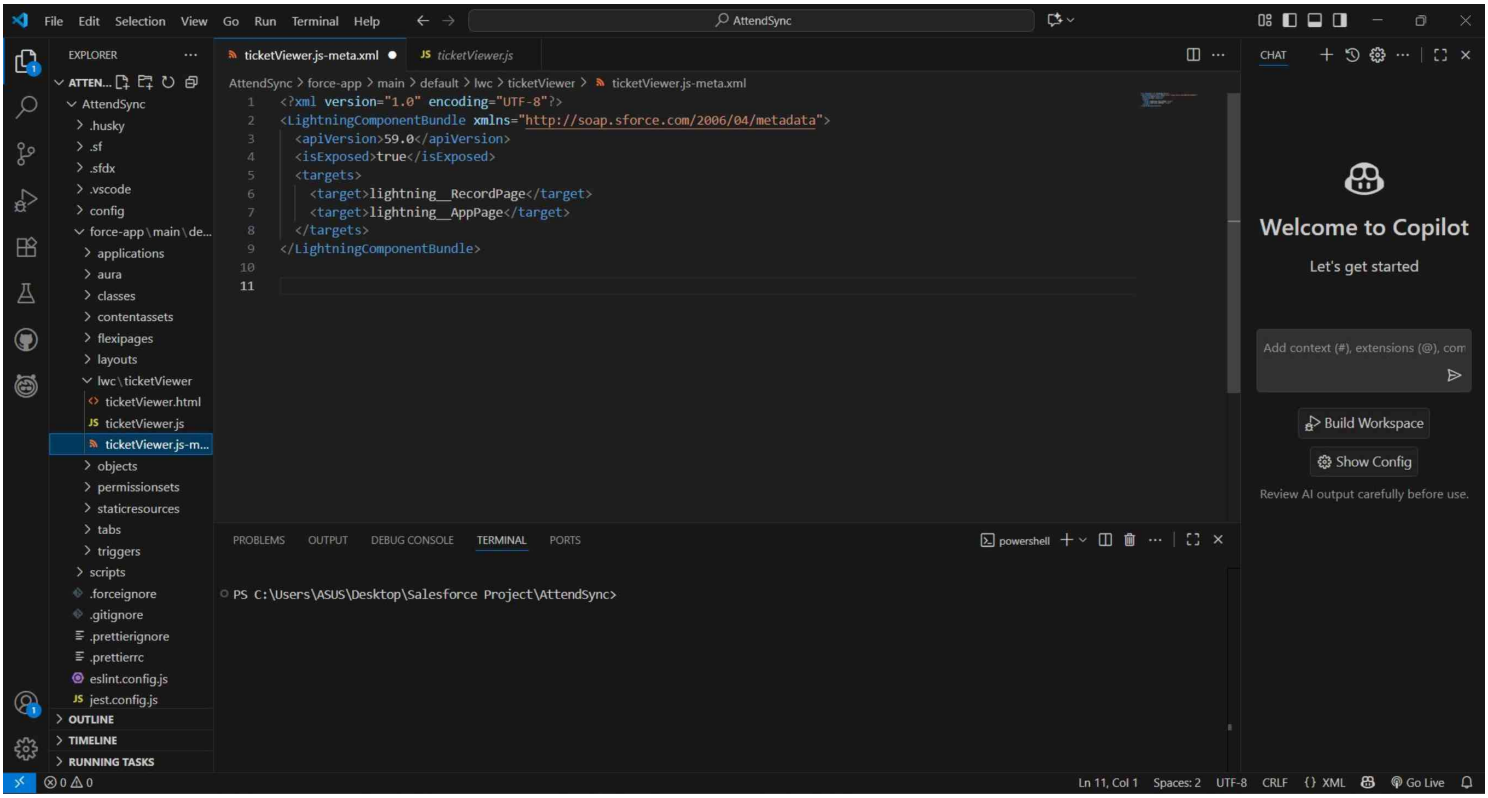

Setup
Home
Object Manager
SETUP > OBJECT MANAGER
Ticket
Details
Fields & Relationships
Page Layouts
Lightning Record Pages
Buttons, Links, and Actions
Compact Layouts
Field Sets
Object Limits
Record Types
Related Lookup Filters
Search Layouts
List View Button Layout
Restriction Rules
Lightning Page
Akash_Ticket_Layout
Help for this Page
Lightning Page Detail
Edit
Clone
Delete
Information
Name
Akash_Ticket_Layout
Label
Akash Ticket Layout
Description
Edit
Clone
Delete
Assignments By App
App
Form Factor
Akash Events
Desktop and phone
Assignments By App, Record Type, and Profile
No Assignments to display
Back To Top
Always show me more records per related list


File Edit Selection View Go Run Terminal Help
AttendSync
EXPLORER
ticketViewer.js-meta.xml
ticketViewer.html
AttendSync > force-app > main > default > lwc > ticketViewer > ticketViewer.html > template
<template>
<lightning-card title="Ticket Viewer">
<template if:true={tickets}>
<template for:each={tickets} for:item="ticket">

{ticket.Name} - {ticket.Status__c} - {ticket.Event__r.Name}

</template>
</template>
<template if:false={tickets}>
No tickets found.
</template>
</lightning-card>
</template>
AttendSync
.husky
.sf
.sfdx
.vscode
config
force-app\main\de...
applications
aura
classes
contentassets
flexipages
layouts
lwc\ticketViewer
ticketViewer.html
ticketViewer.js
ticketViewer.js-m...
objects
permissionsets
staticresources
tabs
triggers
scripts
.forceignore
.gitignore
.prettierignore
.prettierrc
eslint.config.js
jest.config.js
OUTLINE
TIMELINE
RUNNING TASKS
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
PS C:\Users\ASUS\Desktop\Salesforce Project\AttendSync>
CHAT
Welcome to Copilot
Let's get started
Add context (#), extensions (@), com
Build Workspace
Show Config
Review AI output carefully before use.


File Edit Selection View Go Run Terminal Help
AttendSync
EXPLORER
ticketViewer.js-meta.xml
ticketViewer.js
AttendSync > force-app > main > default > lwc > ticketViewer > ticketViewer.js > ...
import { LightningElement, wire } from 'lwc';
import getTickets from '@salesforce/apex/TicketController.getTickets';
export default class TicketViewer extends LightningElement {
tickets;
@wire(getTickets)
wiredTickets({ error, data }) {
if (data) {
this.tickets = data;
} else {
this.tickets = null;
}
}
}
AttendSync
.husky
.sf
.sfdx
.vscode
config
force-app\main\de...
applications
aura
classes
contentassets
flexipages
layouts
lwc\ticketViewer
ticketViewer.html
ticketViewer.js
ticketViewer.js-m...
objects
permissionsets
staticresources
tabs
triggers
scripts
.forceignore
.gitignore
.prettierignore
.prettierrc
eslint.config.js
jest.config.js
OUTLINE
TIMELINE
RUNNING TASKS
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\ASUS\Desktop\Salesforce Project\AttendSync>
CHAT
Welcome to Copilot
Let's get started
Add context (#), extensions (@), com
Build Workspace
Show Config
Review AI output carefully before use.


File Edit Selection View Go Run Terminal Help
AttendSync
EXPLORER
ticketViewer.js-meta.xml
ticketViewer.js
AttendSync > force-app > main > default > lwc > ticketViewer > ticketViewer.js-meta.xml
<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle xmlns="http://soap.sforce.com/2006/04/metadata">
<apiVersion>59.0</apiVersion>
<isExposed>true</isExposed>
<targets>
<target>lightning__RecordPage</target>
<target>lightning__AppPage</target>
</targets>
</LightningComponentBundle>
AttendSync
.husky
.sf
.sfdx
.vscode
config
force-app\main\de...
applications
aura
classes
contentassets
flexipages
layouts
lwc\ticketViewer
ticketViewer.html
ticketViewer.js
ticketViewer.js-m...
objects
permissionsets
staticresources
tabs
triggers
scripts
.forceignore
.gitignore
.prettierignore
.prettierrc
eslint.config.js
jest.config.js
OUTLINE
TIMELINE
RUNNING TASKS
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\ASUS\Desktop\Salesforce Project\AttendSync>
CHAT
Welcome to Copilot
Let's get started
Add context (#), extensions (@), com
Build Workspace
Show Config
Review AI output carefully before use.
Ln 11, Col 1 Spaces: 2 UTF-8 CRLF {} XML Go Live