Title:

A CRM Application for Banquet Hall Booking

College: Seshadri Rao Gudlavalleru Engineering College

Team Details:

1. T. Gayatri   IT(23481A12N5)
2. S. Silas   ECE(23481A04B8)
3. N. Kowshik   CSE(22481A05G6)
4. N. Lakshman Kumar   AIML(22481A4281)

User Story:

Banquet hall booking involves the process of reserving a venue space for various events such as weddings, corporate gatherings, birthday parties, and other special occasions. This process typically begins with a client's inquiry about the availability of the banquet hall for their desired date and event type. The client discusses their event requirements including the date, expected number of guests, preferred layout, catering options, and any specific requests they may have. Based on the client's requirements, the banquet hall provides a detailed proposal outlining the available packages, pricing, inclusions, and any additional services or amenities. After the booking is confirmed, the banquet hall's event coordinator or planning team works closely with the client to plan and coordinate all aspects of the event. This includes finalizing event details, such as seating arrangements, menu selection, decor preferences, audiovisual requirements, and any other special requests. On the day of the event, the banquet hall staff ensure everything runs smoothly, troubleshoot any issues that may arise, and provide assistance to the client and their guests as needed.

1. INTRODUCTION:

1.1 Project Overview:

Banquet halls serve as key venues for various types of events including weddings, birthdays, corporate meetings, cultural functions, and other celebrations. Managing these events involves handling multiple bookings, customizing services based on client needs, and ensuring seamless coordination between internal teams and vendors. Traditionally, the booking process involves phone calls, spreadsheets, and manual follow-ups which are error-prone and inefficient.

This project aims to create a **CRM (Customer Relationship Management) system using Salesforce** to streamline the entire banquet hall booking lifecycle. The system centralizes client information, automates service assignments, and offers dashboards to track ongoing and upcoming events. By leveraging Salesforce's robust features such as Lightning components, Apex triggers, flows, and dashboards, this project demonstrates how automation and centralization can transform the client experience and business efficiency.

1.2 Purpose:

The primary goal of this project is to design and develop a **scalable, customizable, and user-friendly banquet hall booking system** built on the Salesforce platform. The system provides:

- A streamlined interface to handle booking requests
- Automation for assigning event-related services (e.g., catering, DJ, decoration)
- Prevention of double bookings and scheduling conflicts
- Real-time visibility into bookings and services
- Dashboard and reporting capabilities for management

This application also lays the foundation for future enhancements like mobile access, online payments, and customer feedback tracking.

2. IDEATION PHASE

2.1 Problem Statement

Many banquet hall businesses face common challenges such as:

- Lack of a centralized platform to manage client data and service requirements
- Miscommunication between staff and clients due to manual coordination
- Double bookings or scheduling conflicts due to outdated systems
- No structured way to track bookings, services, or revenue

These challenges result in poor customer experience, lost revenue opportunities, and inefficient internal operations. The need for a CRM-based solution becomes evident to address these pain points.

2.2 Empathy Map Canvas

Customer Persona: Event Organizer / Customer

- **Says**: "I need everything perfect for the event!"
- **Thinks**: "Will they manage everything on time?"
- **Does**: Calls to inquire, follows up frequently
- **Feels**: Anxious about service quality and punctuality

Goals:

- Hassle-free booking
- Clear visibility of services and pricing
- Confirmation and timely updates

Challenges:

- Unclear communication about services
- Delays in confirmations
- Lack of post-event feedback mechanisms

2.3 Brainstorming

From various brainstorming sessions, the following ideas were shortlisted:

- A **centralized CRM platform** to track all bookings
- **Automated workflows** for service assignment and updates
- **Validation rules** to prevent errors and duplicate records
- **Reports and dashboards** for management to review business performance
- **User role-based permissions** to control access based on responsibilities
- A base system that can be extended to include mobile integration and payments

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

The customer journey was mapped across the following stages:

- **Inquiry** – Customer contacts the business via call or website
- **Consultation** – Event type, requirements, budget, date selection
- **Quotation & Confirmation** – Booking details are entered in CRM
- **Service Allocation** – Services are scheduled automatically
- **Event Execution** – Staff and vendors coordinate execution
- **Post-Event** – Feedback is collected and stored

Pain Points Identified:

- Booking clashes
- Lack of automation in service assignment
- Manual tracking of services and feedback

3.2 Solution Requirement

· **Salesforce Developer Org**

· **Custom Objects** like Hall_Booking__c, Service__c

· **Standard Objects** like Account, Contact

· **Automation** through Flows and Apex Triggers

· **Validation Rules** for data integrity

· **Reports and Dashboards** for visualization

· **Lightning App Builder** for UI customization

3.3 Data Flow Diagram

User enters booking data → Triggers automated flows → Services assigned → Reports updated

Admins use dashboards for overview and management
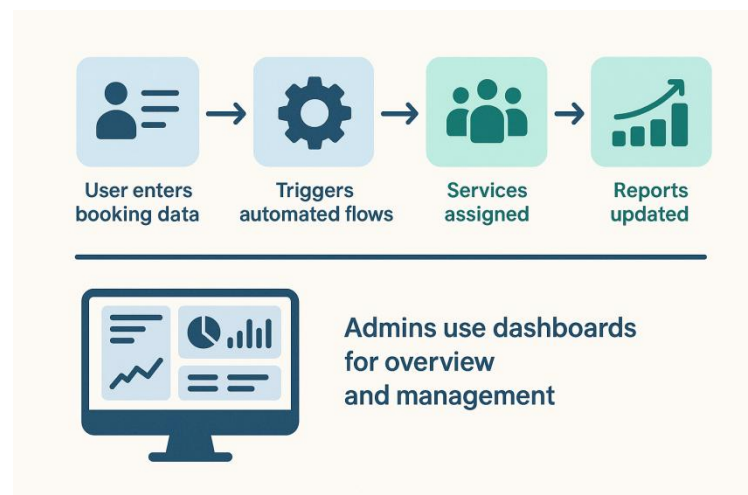
3.4 Technology Stack

Component	Technology Used
Platform	Salesforce Developer Org
Backend Logic	Apex Triggers, Flows
Data Management	Custom & Standard Objects
User Interface	Salesforce Lightning
Reporting	Dashboards & Reports
Automation	Process Builder, Flow

4. PROJECT DESIGN

4.1 Problem Solution Fit

The current market gap in banquet booking management is addressed by this solution through automation and data centralization. Instead of using Excel sheets or manual registers, this solution gives users a modern CRM-based system with real-time updates, role-based views, and streamlined workflows.

4.2 Proposed Solution

A Salesforce application that:

- Allows users to create and manage bookings with ease
- Automatically assigns relevant services using Flow logic
- Provides dashboards to monitor event status and service delivery
- Ensures error-free entries using validation and trigger logic

4.3 Solution Architecture

Layers:

- **Presentation Layer**: Salesforce Lightning Experience

- **Logic Layer**: Apex Triggers, Flows, Validation Rules

- **Data Layer**: Custom Objects, Standard Objects, Metadata
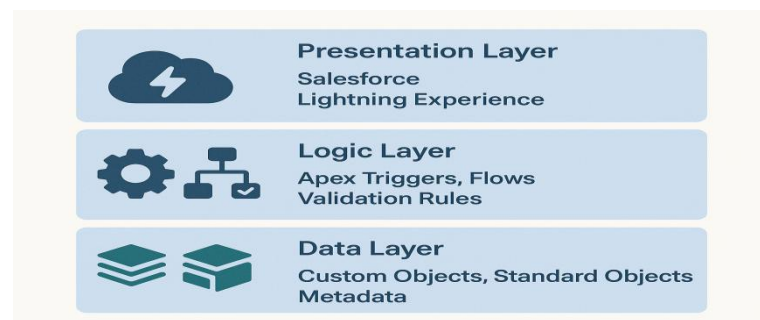
5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning
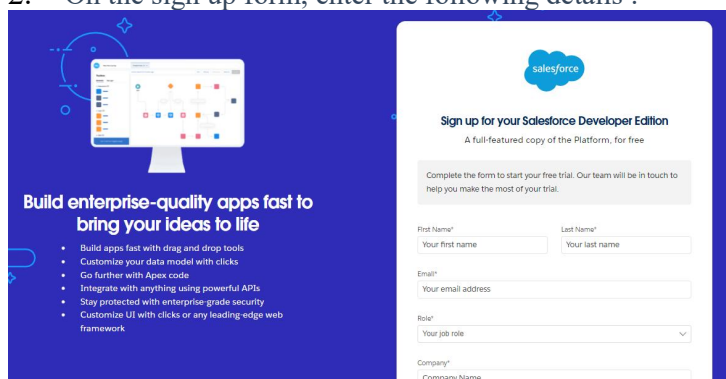
Milestones with relevant screenshots provided:

I. **Salesforce Account Setup:**

Activity 1: Creating Developer Account

Creating a developer org in salesforce.

1. Go to https://developer.salesforce.com/signup
2. On the sign up form, enter the following details :

1. First name & Last name
2. Email
3. Role : Developer
4. Company : College Name
5. Country : India
6. Postal Code : pin code
7. Username : should be a combination of your name and company

This need not be an actual email id, you can give anything in the format : username@organization.com
Click on sign me up after filling these.

Activity 2: Account Activation:

1. Go to the inbox of the email that you used while signing up. Click on the verify account to activate your account. The email may take 5-10mins.
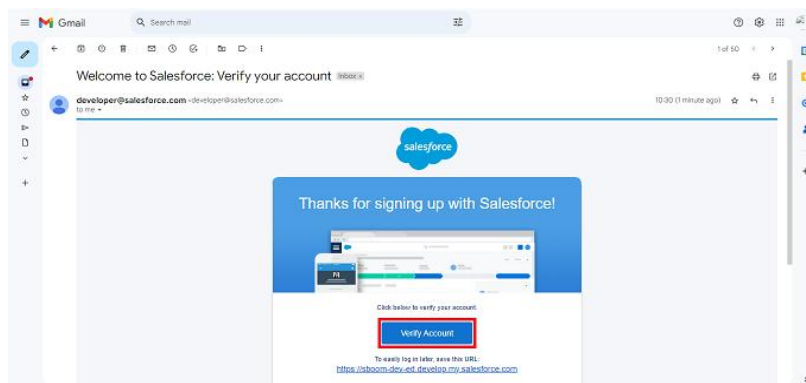
2. Click on Verify Account
3. Give a password and answer a security question and click on change password.

4. Then you will redirect to your salesforce setup page.
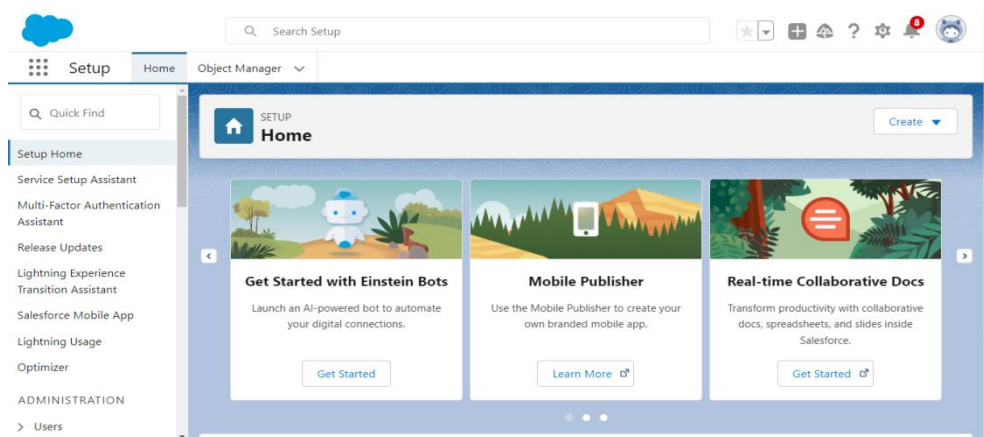
2. Object Creation:

The purpose of creating a Banquet Hall custom object is to store and manage information about Banquet Hall Bookings.
For that we need to create 3 objects:

i. Banquet Hall

ii.   Billing

iii.   Staff

3.   Tabs:

For this milestone we need to create 3 tabs for the CRM application Banquet hall, Billings and Staff tabs

4. Lightning App for Banquet Hall Booking:

Create a lightning app for the booking :
i.   From Setup, enter App Manager in the Quick Find and select App Manager.
ii.   Click New Lightning App.
iii.   Enter Banquet Hall Booking as the App Name, then click next
iv.   Under App Options, leave the default selections and click next.
v.   Under Utility Items, leave as is and click Next.
vi.   From Available Items, select Banquet Halls, Billings, Staff, Reports, and Dashboards and move them to Selected Item and Click Next.
vii.   From Available Profiles, select System Administrator and move it to Selected Profiles.
viii.   Click Save & Finish.

5. Fields:

Create respective fields for the objects

i.   Banquet Hall:

ii. **Billing**:

iii. **Staff**:

6. Creation of Page Layouts:

Here we have to create Two(2) Page Layouts (Official Booking, Celebrations).

i. Official Booking

ii. Celebrations

Finally it should look like this after the creation of the two layouts

7. Creation of Record Types:

A record type is a template that determines the layout and behavior of a record based on the user's profile, the record's criteria, or the user's selection. Each record type defines a distinct set of picklist values for certain fields, specific page layouts, and unique business processes. Here we create 2 record types Celebrations and Official Booking.

8. Validation Rules:

Validation rules are created using Salesforce's point-and-click interface, typically within the Object Manager. They consist of a formula that evaluates to either true or false based on certain conditions.
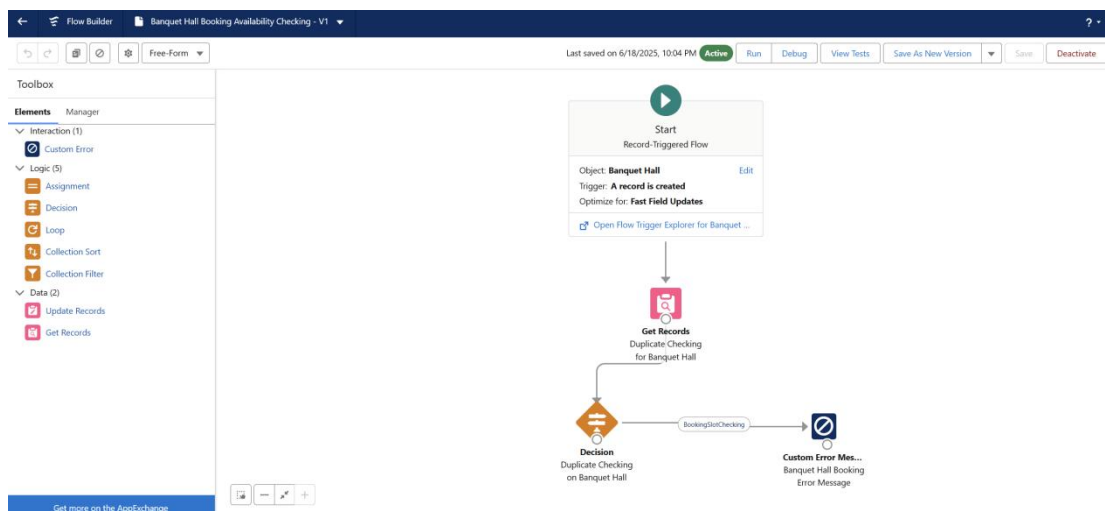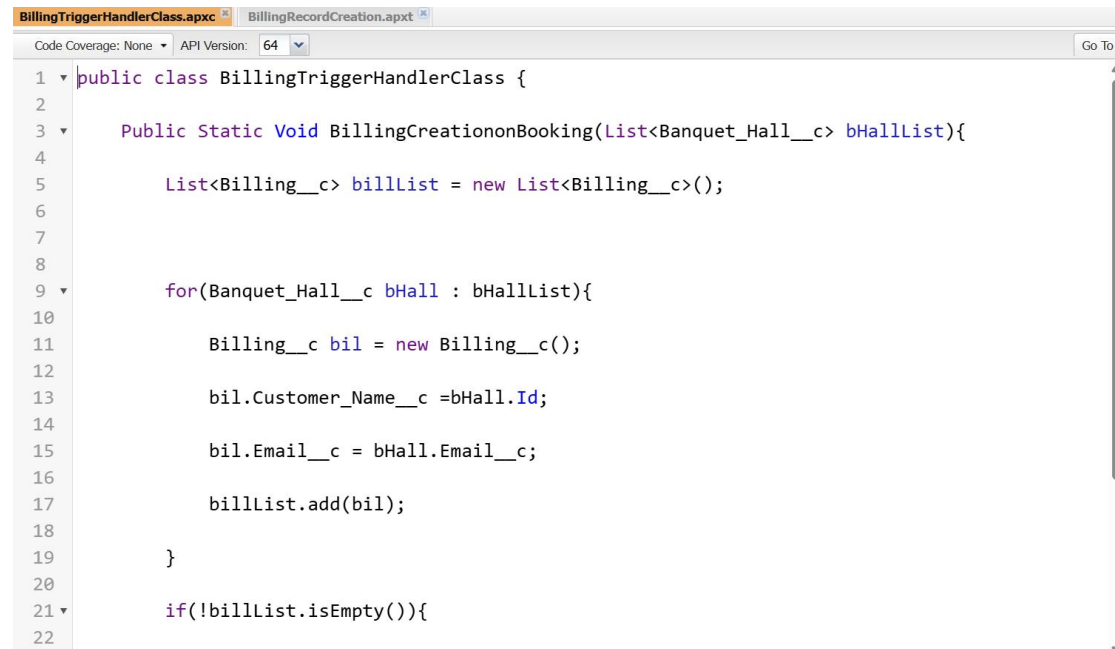

9. Flows:

Flows are declarative, drag-and-drop tools that allow administrators to create guided experiences for users, automate complex business processes, and integrate with other Salesforce features and external systems.

10. Triggers:
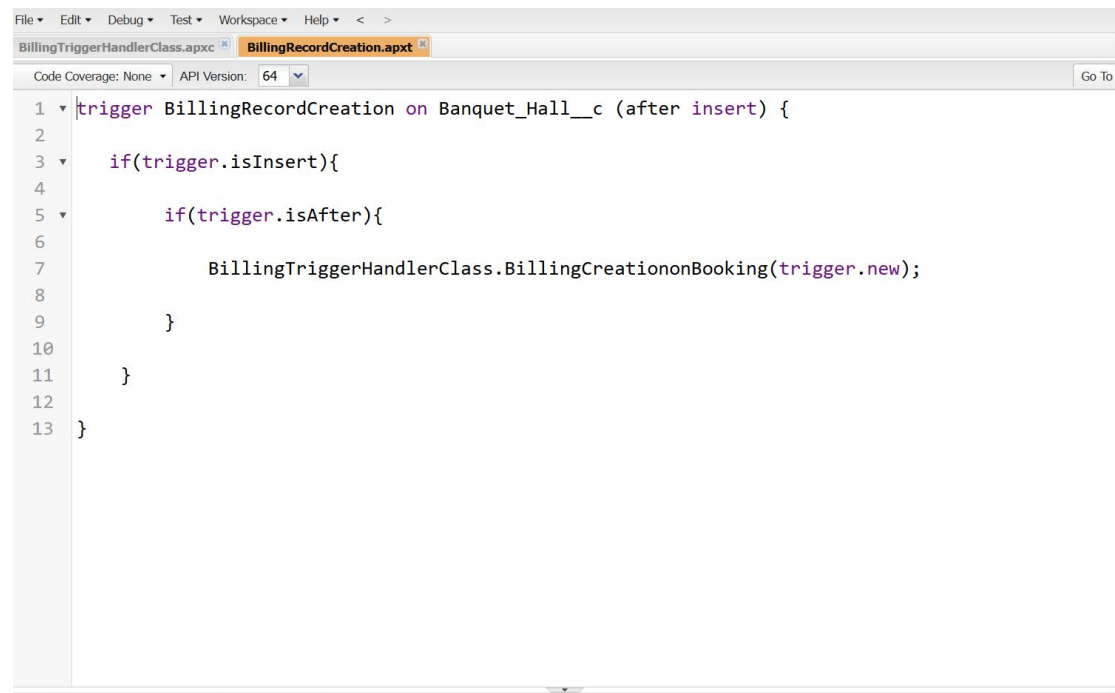
Triggers in Salesforce are pieces of Apex code that execute before or after records are inserted, updated, deleted, or undeleted in the Salesforce database. They enable developers to implement custom business logic and automate processes in response to data changes.

BillingTriggerHandlerClass

```
public class BillingTriggerHandlerClass {

    Public Static Void BillingCreationonBooking(List<Banquet_Hall__c> bHallList){

        List<Billing__c> billList = new List<Billing__c>();


        for(Banquet_Hall__c bHall : bHallList){

            Billing__c bil = new Billing__c();

            bil.Customer_Name__c =bHall.Id;

            bil.Email__c = bHall.Email__c;

            billList.add(bil);

        }

        if(!billList.isEmpty()){
```

BillingRecordCreation Trigger

```
trigger BillingRecordCreation on Banquet_Hall__c (after insert) {

    if(trigger.isInsert){

        if(trigger.isAfter){

            BillingTriggerHandlerClass.BillingCreationonBooking(trigger.new);

        }

    }

}
```

11. Reports:

In Salesforce, reports are tools that allow users to visualize and analyze their data to gain insights into their business processes. They enable users to create, customize, and share various types of reports based on the data stored in their Salesforce org.

12. Dashboards:

In Salesforce, dashboards are powerful tools for visualizing and analyzing data from reports, providing users with at-a-glance insights into key metrics and performance indicators.

Dashboard Creation:
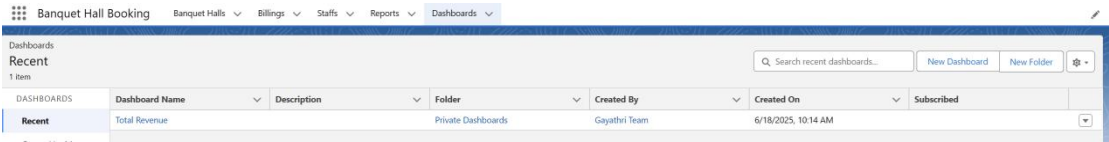
Preview:
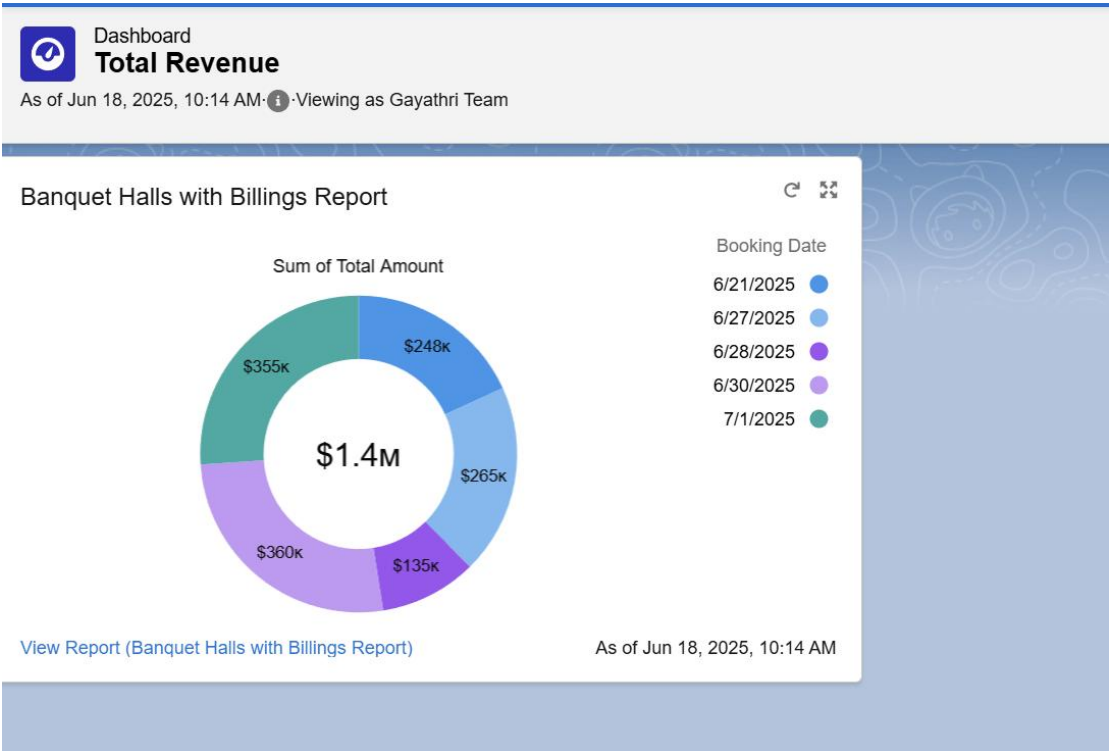
13. Conclusion:

By implementing the enhanced Banquet Hall Booking Management System in Salesforce, the venue can offer a seamless experience for clients seeking banquet hall services, catering/menu options, DJ services, decorations, and marriage-related services. This comprehensive solution will streamline operations, enhance customer experience, and drive revenue growth through efficient booking management and personalized service delivery.

The project demonstrates the potential of Salesforce in managing real-world event booking scenarios. It simplifies complex booking procedures, reduces human error, and provides automation for routine tasks. The application is scalable and forms a strong base for further enhancements, making it a valuable asset for banquet hall businesses.

6. FUNCTIONAL AND PERFORMANCE TESTING

As the sole developer and tester, the application was manually validated using various scenarios to confirm correct behavior, automation flow, and error prevention.

Functional Checks:

- Booking creation, editing, cancellation
- Service assignments via Flows
- Record-type based page layout rendering
- Lookup relationships between contacts and bookings
- Validation of date, guest count, and event type fields

Performance Validation:

- Tested by creating 30+ test bookings and service requests
- Observed fast dashboard loading and accurate metrics
- No duplicate entries or crashes observed
- Automation flows triggered correctly and promptly

7. RESULTS

The final Salesforce CRM application successfully handles:

- End-to-end banquet hall bookings
- Automated service allocation
- Real-time data visualization
- Data validation and integrity

All expected features work as designed, and the system is ready for actual deployment or demonstration purposes.

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Cloud-based access from anywhere
- Automated service coordination
- Accurate data validation
- Visual dashboards for real-time monitoring

Disadvantages:

- Learning curve for Salesforce beginners
- Requires stable internet access
- Limited to Salesforce ecosystem without integration

10. FUTURE SCOPE

Future improvements can make the application more robust and accessible:

- Integration with **online payment gateways** for full-cycle automation
- Development of a **mobile interface** using Salesforce Mobile Publisher or LWC
- **Customer feedback** module using custom surveys
- SMS/Email notifications using Salesforce APIs
- Event calendar integration for staff planning and reminders

11. APPENDIX

- **Source Code / Configurations**: Configured within Salesforce Developer Org
- **Dataset**: Dummy entries created for testing purposes
- **Demo video link:**
  https://drive.google.com/file/d/1KgZ2hgTrMBOMplasYwaIWO5axC5xETWm/view?usp=sharing