

Project Title: Visa Management System (VMS) for the Fictitious Country of Eldoria

Objective:

Develop a Salesforce-based Visa Management System (VMS) for Eldoria, a fictitious country, to manage and automate its visa approval process for foreign tourists based on specific country regulations.

Case Description:

Eldoria has diverse visa regulations for different nations:

- **Visa-Free Nations:** Citizens can travel without a visa.
- **Visa-Restricted Nations:** Citizens are not eligible for a visa.
- **Visa-on-Arrival Nations:** Citizens receive a visa upon arrival.
- **Regular Visa Nations:** Citizens must apply for a visa beforehand, requiring specific approvals.

Functional Requirements (No Code Activities)

First Create an App

Home Page

1. Data Model:

- **Country Object:** Stores details of countries and their visa status with Eldoria.
 - Fields: Country Name, Visa Status (Picklist: Visa-Free, Visa-Restricted, Visa-on-Arrival, Regular Visa), Other Details.

Fig-1 Country Object

- Fields of Country Object

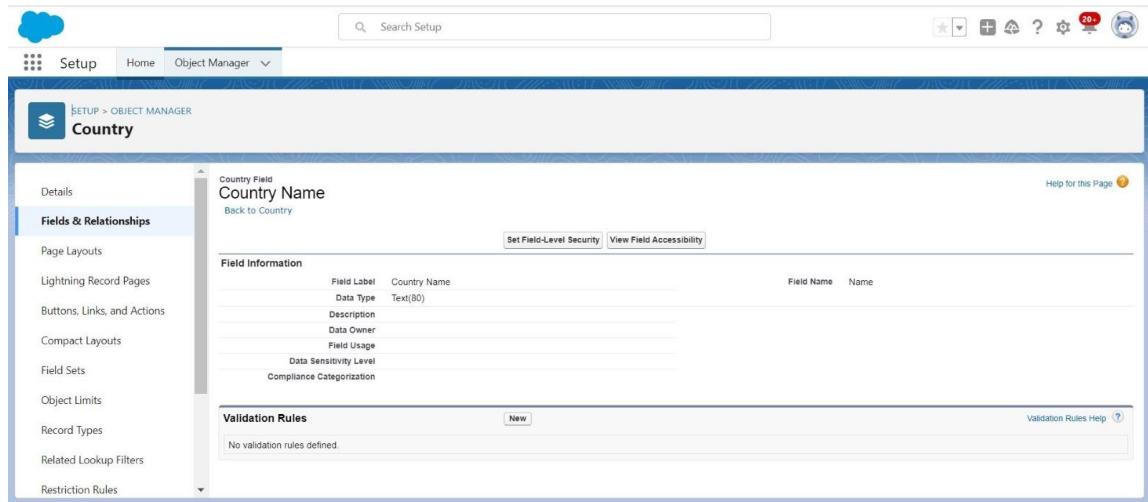


Fig-2 Country Name Field

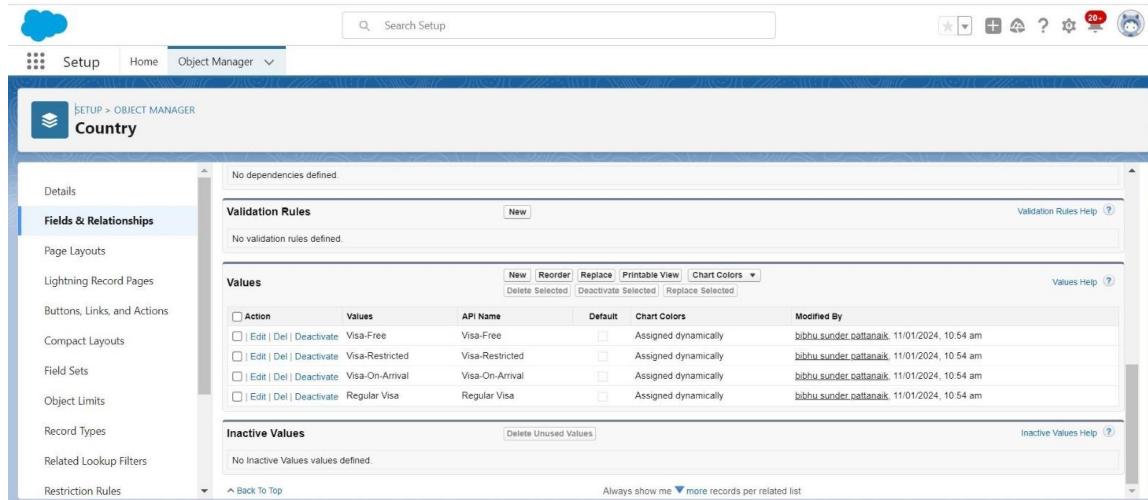


Fig-3 Country Status Field

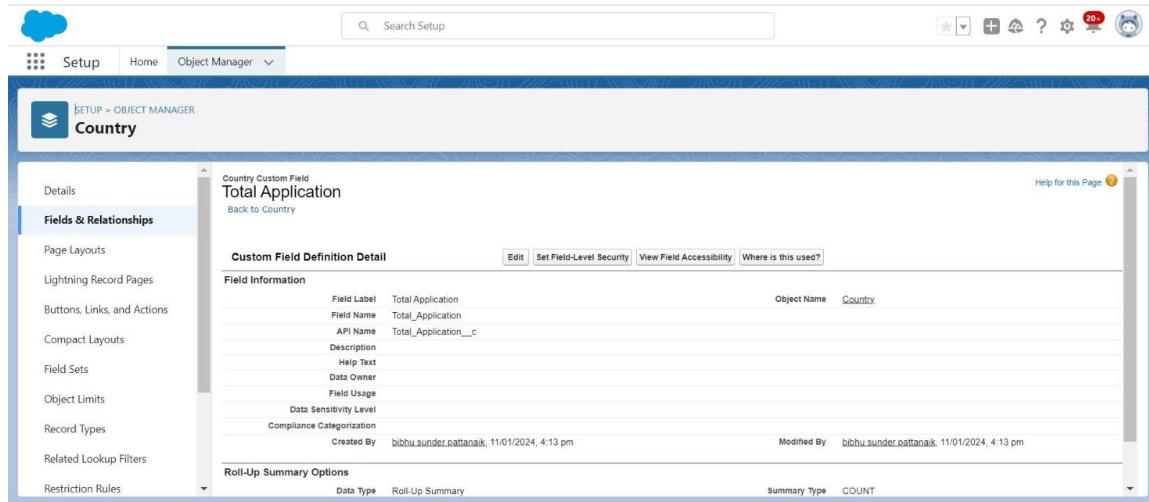


Fig-4 Country Total Application Field

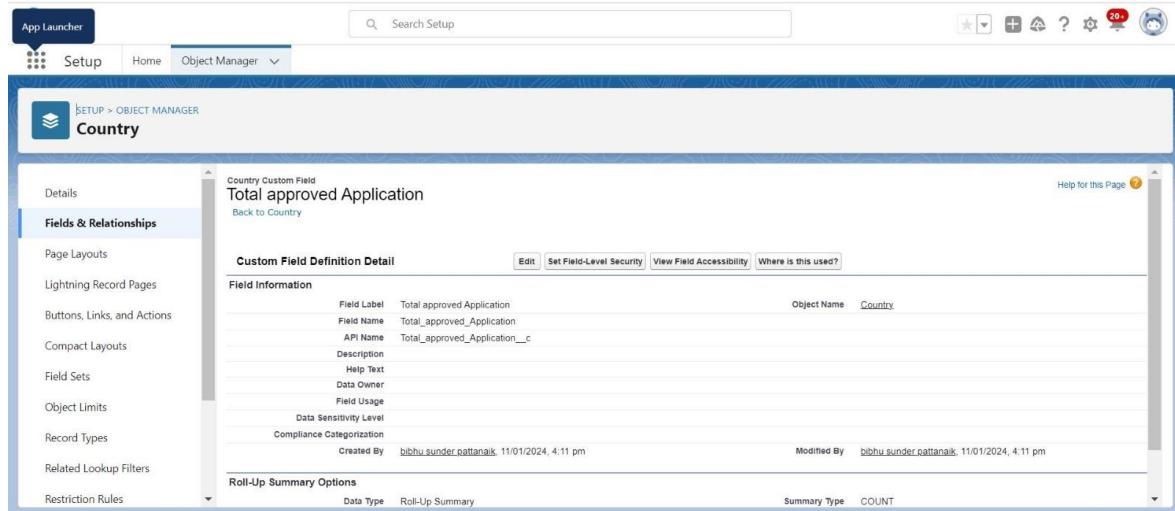


Fig-5 Country Total Approved Field

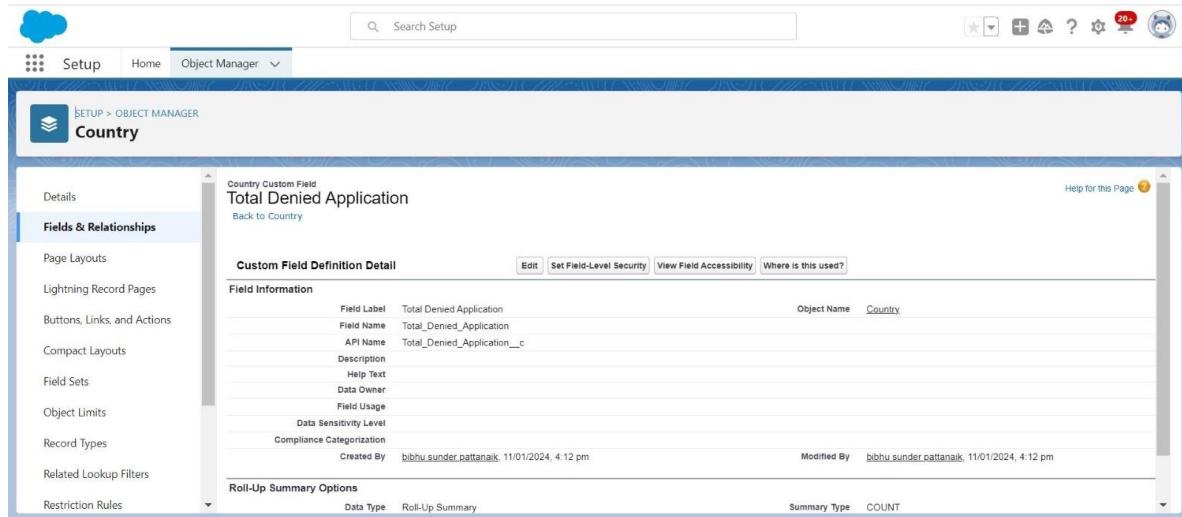


Fig-6 Country Total Denied Field

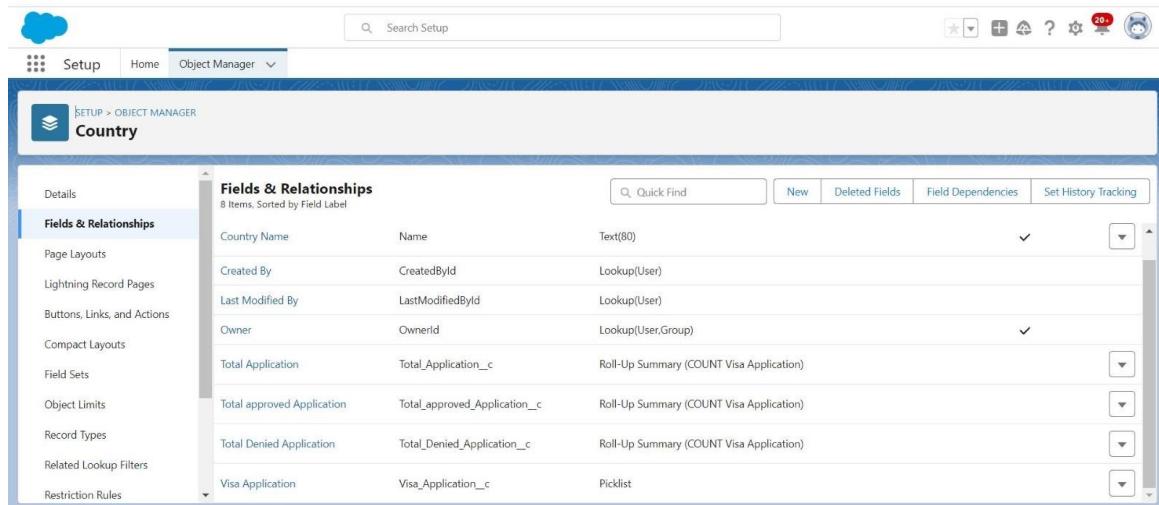


Fig-7 Country All Field & Relationship

- **Applicant Object:** Stores applicant's personal details.
 - Fields: Name, Passport Number, Nationality (Lookup to Country), Purpose of Visit, etc.

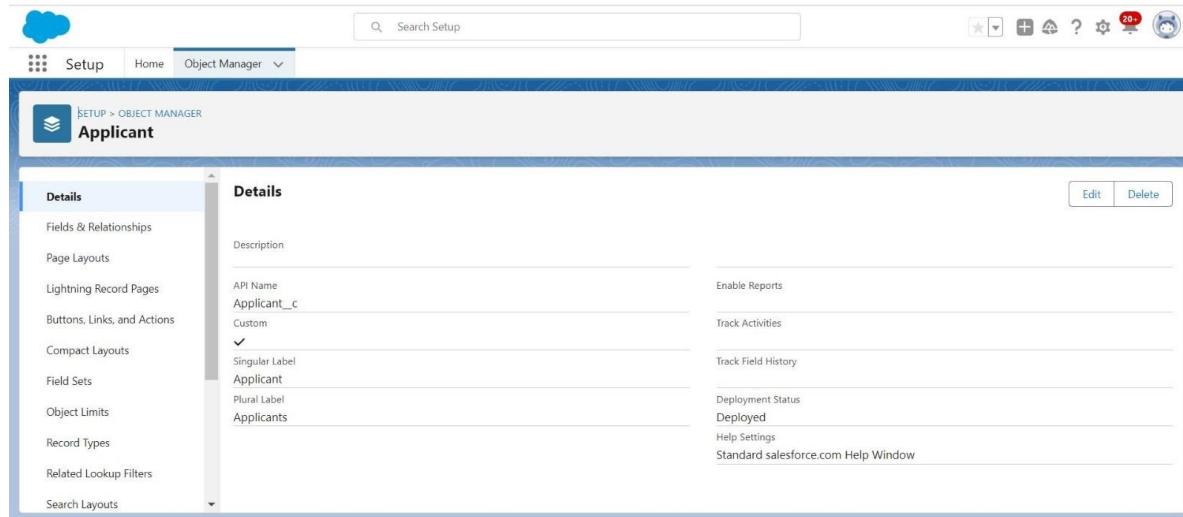


Fig-8 Applicant Object

- Fields of Applicant Object

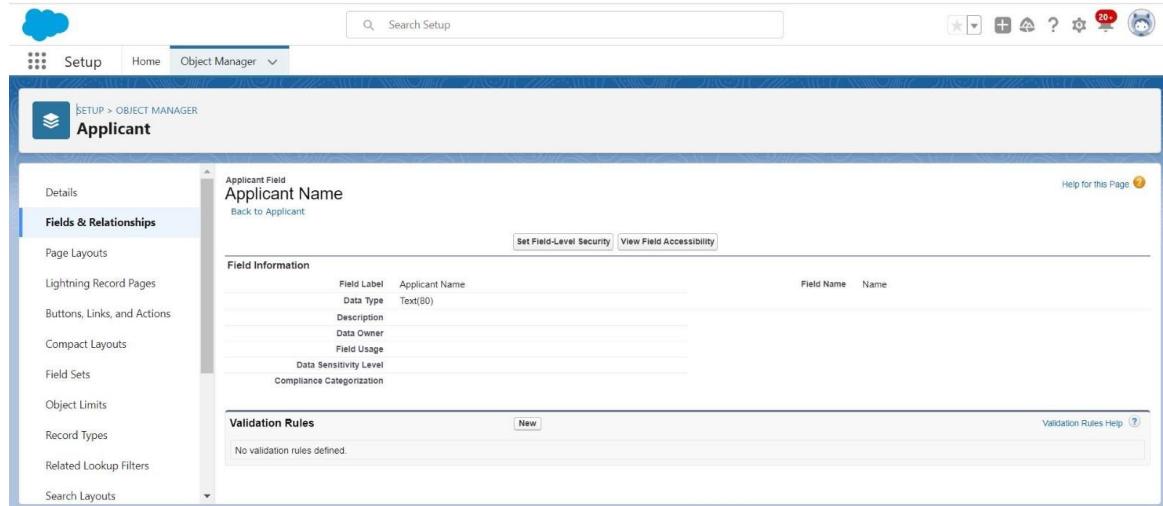


Fig-9 Applicant Name Field

The screenshot shows the Salesforce Object Manager interface for the 'Applicant' object. On the left, a sidebar lists various setup options like Details, Fields & Relationships, Page Layouts, and Lightning Record Pages. The main content area displays the 'Custom Field Definition Detail' for the 'Nationality' field. The field information includes:

Field Label	Nationality	Object Name	Applicant
Field Name	Nationality	Data Type	Lookup
API Name	Nationality__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	bibhu.sunder.pattanaik 11/01/2024, 11:01 am	Modified By	bibhu.sunder.pattanaik 11/01/2024, 11:01 am
Lookup Options	Related To Country	Child Relationship Name	Applicants

Fig-10 Applicant Nationality Field

The screenshot shows the Salesforce Object Manager interface for the 'Applicant' object. The sidebar lists various setup options. The main content area displays the 'Custom Field Definition Detail' for the 'Passport Number' field. The field information includes:

Field Label	Passport Number	Object Name	Applicant
Field Name	Passport_Number	Data Type	Text
API Name	Passport_Number__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	bibhu.sunder.pattanaik 11/01/2024, 10:57 am	Modified By	bibhu.sunder.pattanaik 11/01/2024, 11:40 am
General Options	Required ✓		

Fig-11 Applicant Passport Number(Autogenerated)

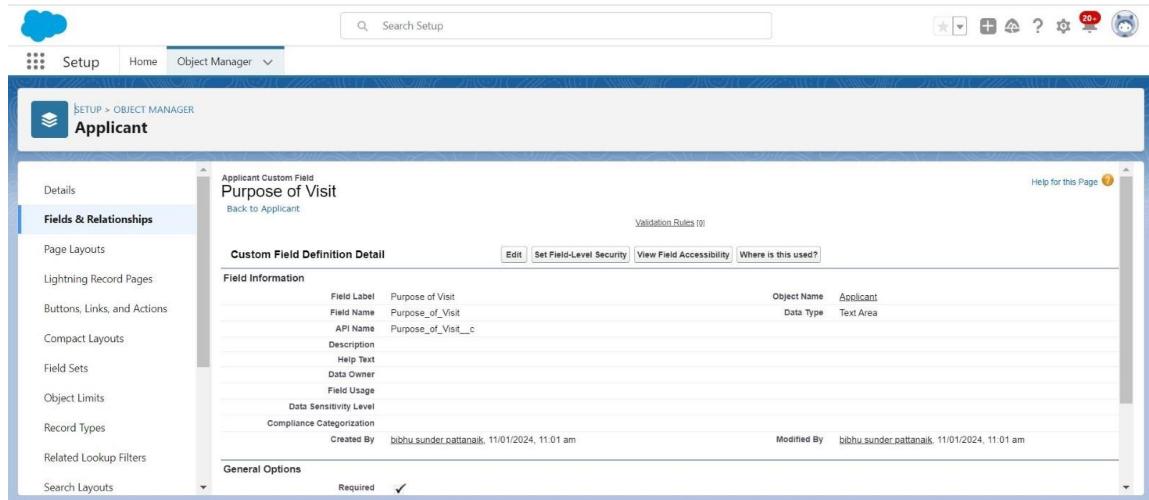


Fig-12 Applicant purpose of visit Field

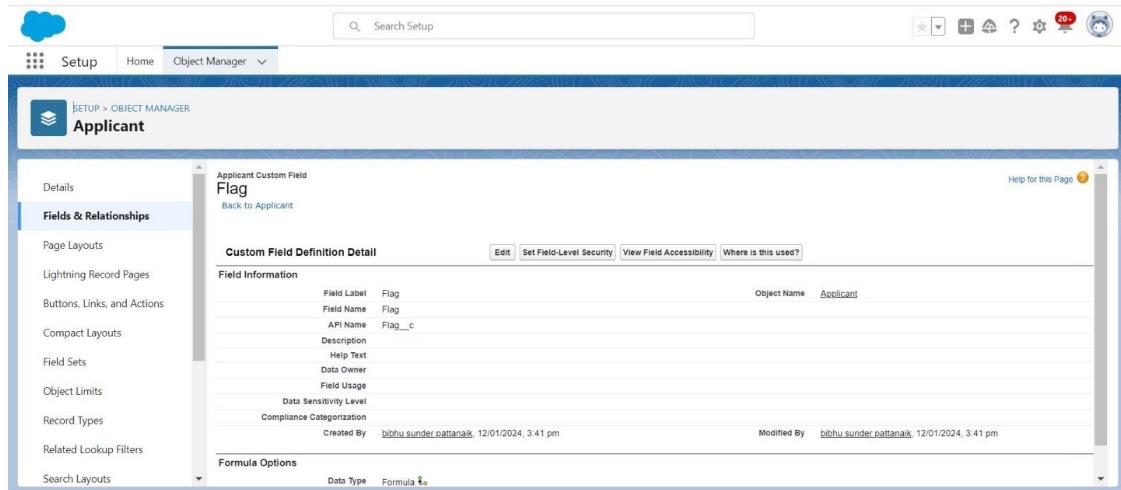


Fig-13 Applicant Flag Field (It is a formula field which automatically ticks if the applicant is from a visa-restricted country)

The screenshot shows the Salesforce Object Manager interface for the 'Applicant' object. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main area displays the 'Fields & Relationships' section, which lists eight items sorted by Field Label. The fields are:

	Name	Type	Default Value
Applicant Name	Name	Text(50)	
Created By	CreatedBy	Lookup(User)	
Flag	Flag_c	Formula (Checkbox)	
Last Modified By	LastModifiedBy	Lookup(User)	
Nationality	Nationality_c	Lookup(Country)	
Owner	OwnerId	Lookup(User,Group)	
Passport Number	Passport_Number_c	Text(8) (Unique Case Insensitive)	
Purpose of Visit	Purpose_of_Visit_c	Text Area(255)	

Fig-14 Applicant All Field and relationships

- **Visa Application Object** (Junction Object): Connects Applicant to Country. Stores visa application details.
 - Fields: Application Date, Status (Picklist: Pending, Approved, Denied), Related Applicant (Lookup to Applicant), Destination Country (Lookup to Country), Travel Dates, Supporting Documents, etc.

The screenshot shows the Salesforce Object Manager interface for the 'Visa Application' object. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main area displays the 'Details' section for the 'Visa Application' object. The left sidebar lists various configuration options: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts, Field Sets, Object Limits, Record Types, Related Lookup Filters, and Search Layouts. The right panel shows the following details:

Field	Value
Description	
API Name	Visa_Application_c
Custom	✓
Singular Label	Visa Application
Plural Label	Visa Applications
Enable Reports	
Track Activities	
Track Field History	
Deployment Status	Deployed
Help Settings	Standard salesforce.com Help Window

Fig-15 Visa Application Object

- Fields of Visa Application Object

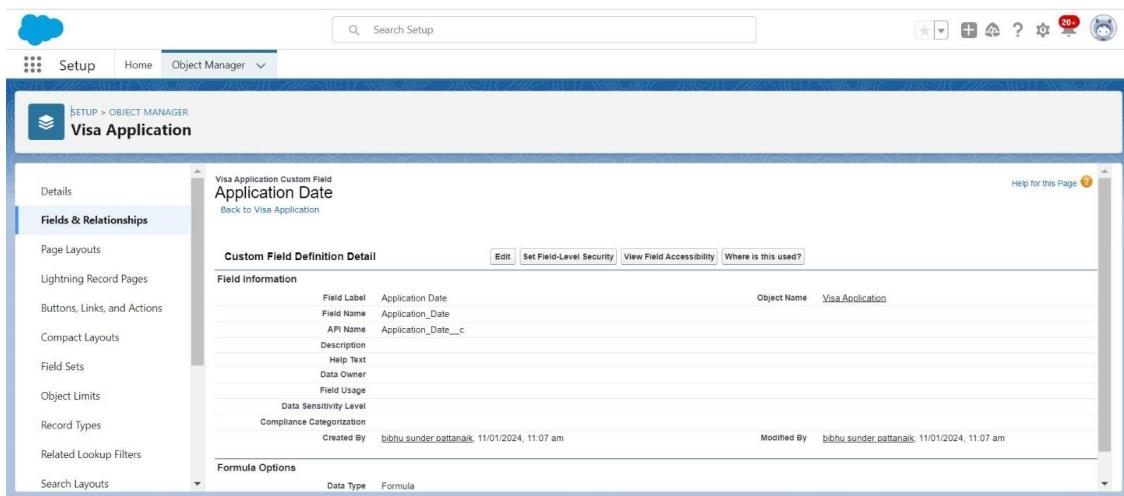


Fig-16 Visa Application Date field(AutoFilling)

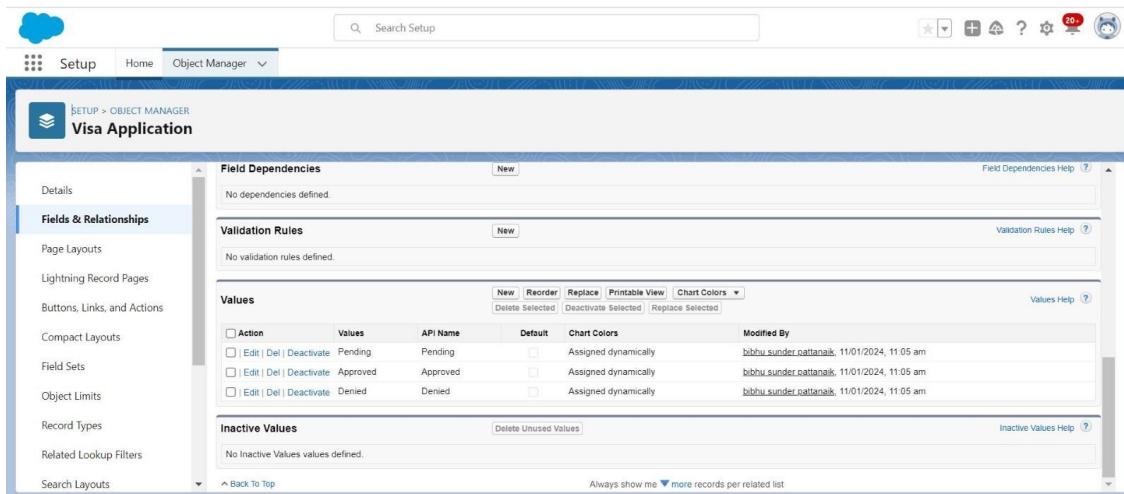


Fig-17 Visa Application Final destination field(lookup to country object)

Action	Values	API Name	Default	Chart Colors	Modified By
<input type="checkbox"/> Edit Del Deactivate	Pending	Pending	<input type="checkbox"/> Assigned dynamically	blue	bibhu.sunder.pattanaik, 11/01/2024, 11:05 am
<input type="checkbox"/> Edit Del Deactivate	Approved	Approved	<input type="checkbox"/> Assigned dynamically	green	bibhu.sunder.pattanaik, 11/01/2024, 11:05 am
<input type="checkbox"/> Edit Del Deactivate	Denied	Denied	<input type="checkbox"/> Assigned dynamically	red	bibhu.sunder.pattanaik, 11/01/2024, 11:05 am

Fig-18 Visa Application Status field(Picklist)

Field Label	Travel Dates	Object Name	Visa Application
Field Name	Travel_Dates	Data Type	Date
API Name	Travel_Dates__c		
Description			
Help Text			
Data Owner			
Field Usage			
Data Sensitivity Level			
Compliance Categorization			
Created By	bibhu.sunder.pattanaik, 11/01/2024, 11:14 am	Modified By	bibhu.sunder.pattanaik, 11/01/2024, 11:14 am

Fig-19 Visa Application Travel Date field

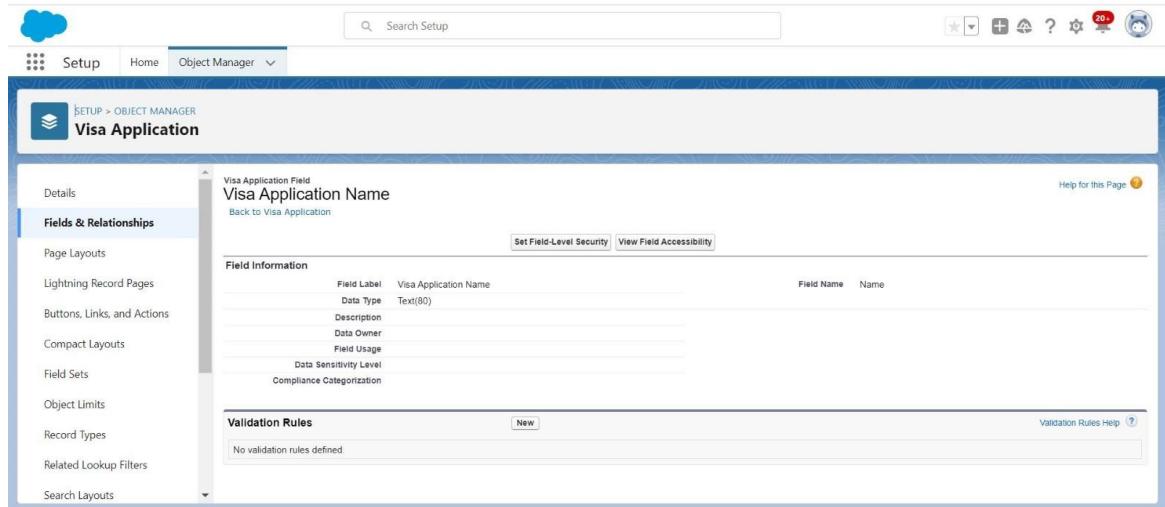


Fig-20 Visa Application Name field

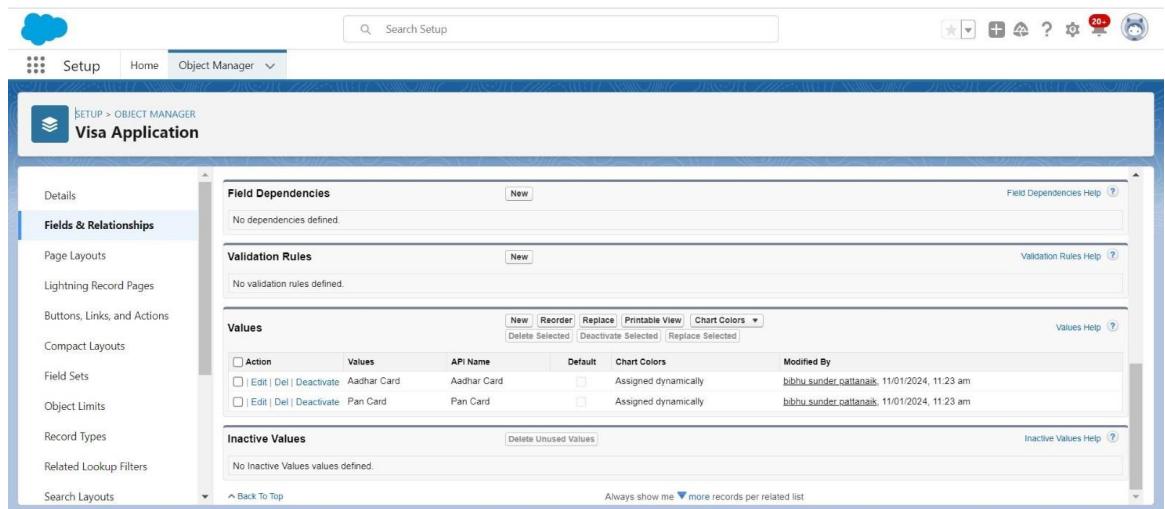


Fig-21 Visa Application Supporting document field(picklist)

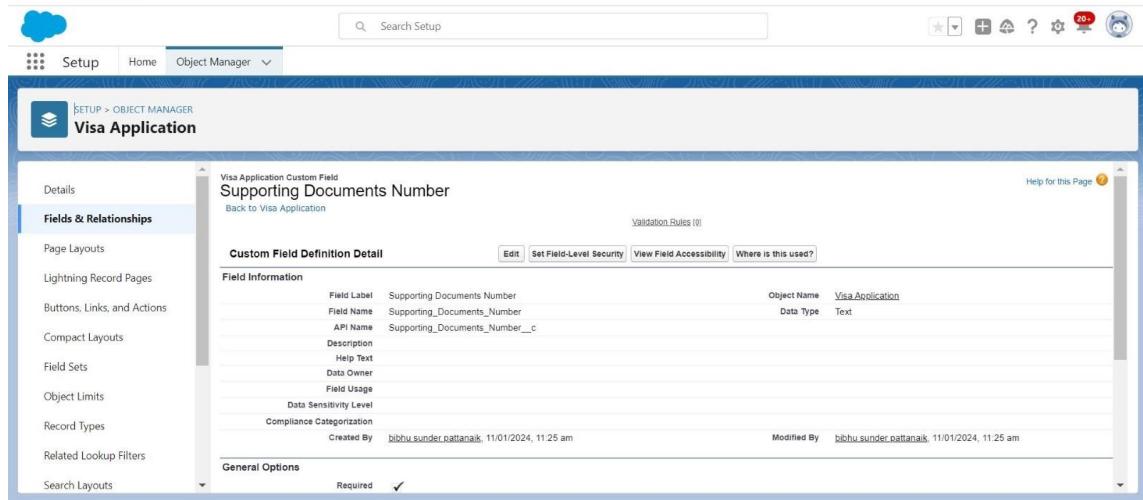


Fig-22 Visa Application Supporting Document Number field

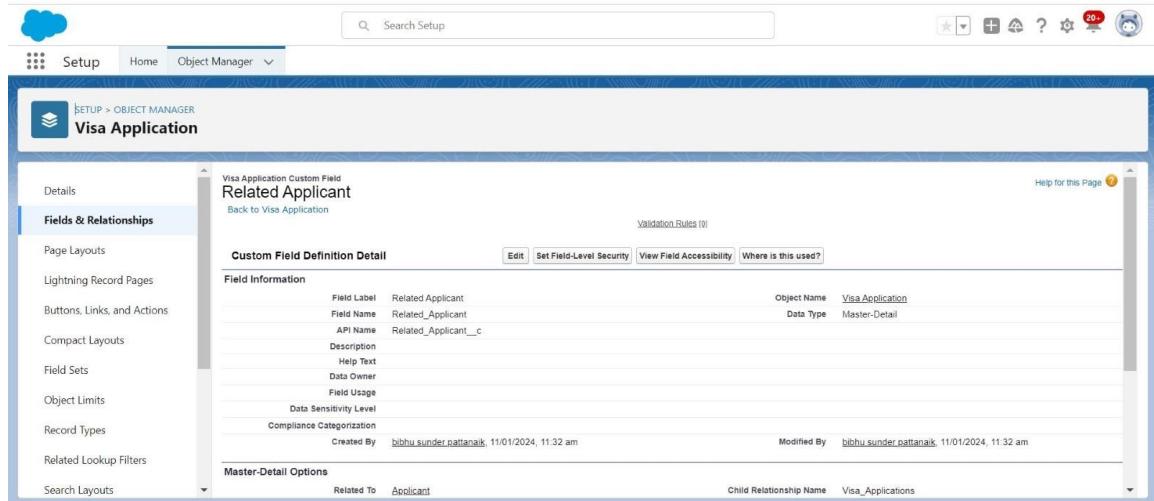


Fig-23 Visa Application Related Applicant field(Master-detailed to Applicant)

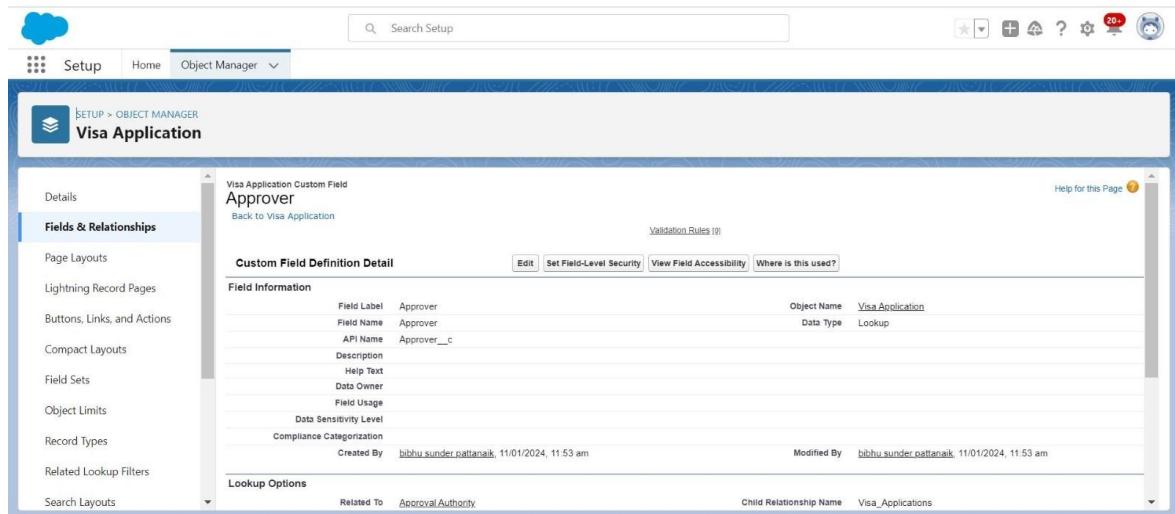


Fig-24 Visa Application Approver field(Master-detail to Approval Authority)

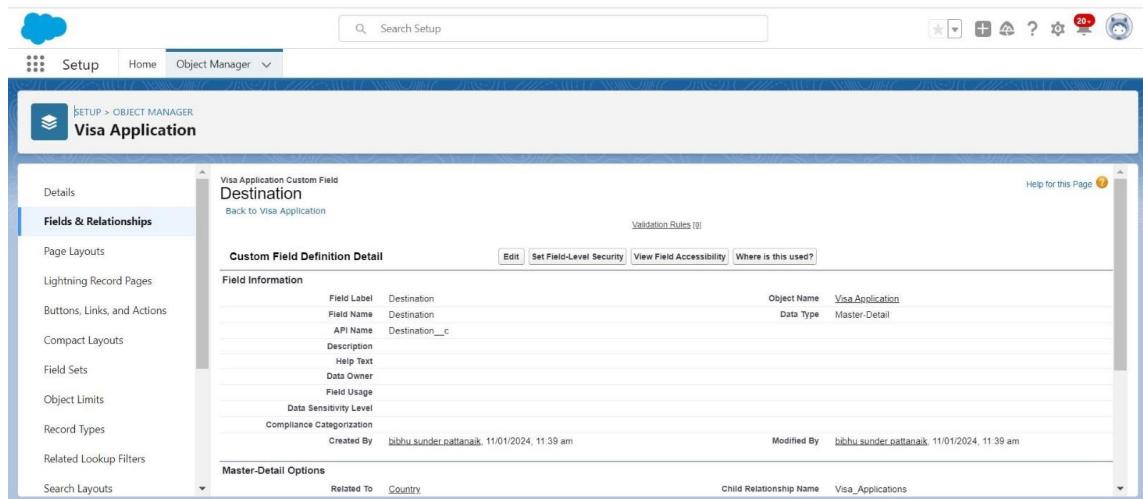


Fig-25 Visa Application All Object Field

- **Approval Authority Object:** Stores details of officials who approve visas.
 - Fields: Name, Role, Assigned Countries (Many-to-Many relationship with Country using a Junction Object).

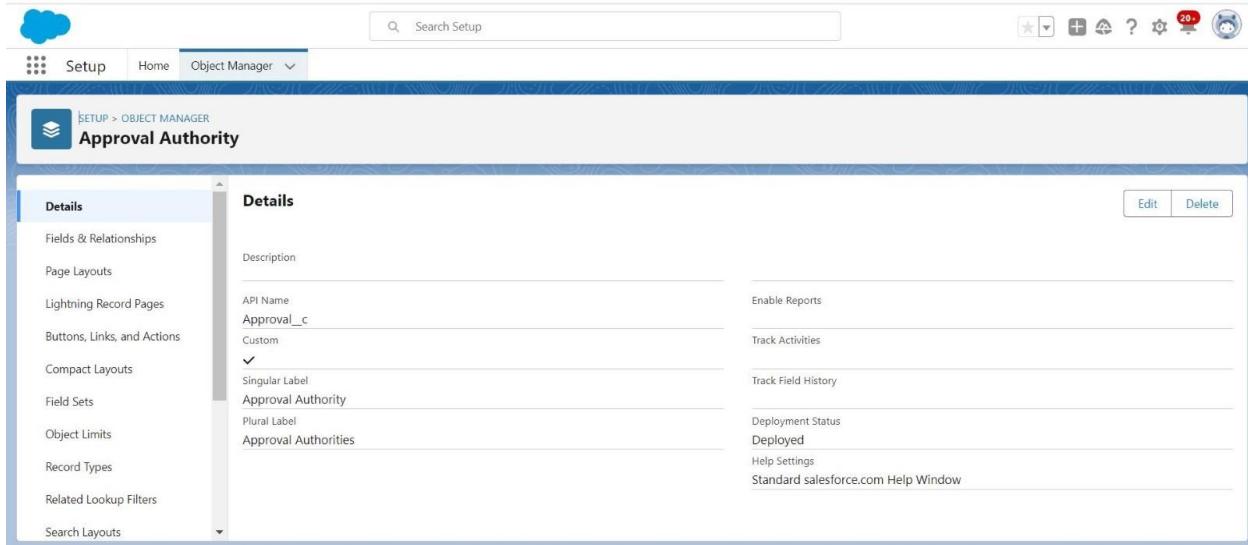


Fig-26 Approval Authority Object

- Fields of Approval Authority Object

Fig-27 Approval Authority Assigned Country(Lookup to country)

Fig-28 Approval Authority Name field

Fig-29 Approval Authority Role(Picklist)

Fig-30 Approval Authority All Fields and Relationship

Workflow Rules and Process Builder:

- Auto-update visa application status based on country status.
- Send email notifications to applicants on status change.
- Reminders to Approval Authority for pending applications.

Fig-31 Flow (Auto-update visa application status based on country status.)

Fig-32 Flow Decision for Status update(Approval)



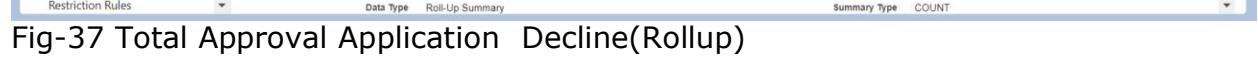
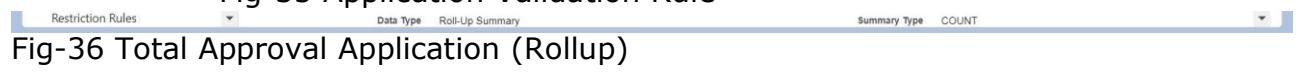
Data Validation and Roll-up Summary Fields:

- Ensure valid passport numbers through validation rules.
- Roll-up summaries on Country to count total applications, approved, and denied visas.

Applicant Validation Rule

Rule Name	Passport_No_Validation_Rule	Active	✓
Error Condition Formula	OR(LEN(Passport_Number__c) != 8, NOT(REGEX(Passport_Number__c, '[a-zA-Z0-9]+')))		
Error Message	You have entered an invalid passport number.	Error Location	Passport Number
Description	bibhu.sunder.pattanaik; 11/01/2024, 3:57 pm	Modified By	bibhu.sunder.pattanaik; 11/01/2024, 4:00 pm
Created By			

Fig-35 Application Validation Rule



Developer Activities (requires Apex and LWC)

1. Apex Triggers

Activity: Auto-Update Application Status

- **Scenario:** When a country's visa policy changes in the Country object, automatically update the status of all pending Visa Applications for that country.

Execute Use Tooling API

Fig-38 Trigger Class

Test Class

Activity: Test for Country Policy Change Trigger

- **Scenario:** Write a test class for the *CountryPolicyChangeTrigger* to ensure it updates visa applications correctly when a country's visa policy changes.

```

1  @isTest
2  private class CountryUpdateTriggerTest {
3
4      @isTest
5      static void testCountryUpdateTrigger() {
6          // Create a test country
7          Country__c testCountry = new Country__c(Name = 'Test Country', Visa_Application__c = 'Visa-Free');
8          insert testCountry;
9
10         // Applicant__c newApplicant = new Applicant(Name = 'Ankan', Passport_Number__c = '85ytjjg94', Nationality__c = 'India',
11         //                                         Purpose_of_Visit__c = 'study');
12
13         List<Applicant__c> applicant = [select id from Applicant__c where Name = 'Likhit' limit 1];
14         //List<Approval__c> approver = [select id from Approval__c where Name = 'Gaurav' limit 1];
15
16         // Create a test pending visa application for the test country
17         Visa_Application__c testVisaApplication = new Visa_Application__c(
18             Name = '04523',
19             Status__c = 'Pending',
20             Travel_Dates__c = Date.newInstance(2024, 2, 17),
21             Supporting_Documents__c = 'Pan Card',
22             Supporting_Documents_Number__c = '1rnnnt3456',
23             Related_Applicant__c = applicant.get(0).Id,
24             Destination__c = testCountry.Id
25             //Approver__c = approver.get(0).id
26         );
27
28         insert testVisaApplication;
29
30         // Update the test country's visa policy
31         testCountry.Visa_Application__c = 'Visa-Restricted';
32         update testCountry;
33
34         // Retrieve the updated visa application
35         testVisaApplication = [SELECT Id, Status__c FROM Visa_Application__c WHERE Id = :testVisaApplication.Id LIMIT 1];
36
37         // Assert that the visa application status is updated as expected
38         System.assertEquals('Denied', testVisaApplication.Status__c, 'Visa application status not updated as expected');
39     }
40 }

```

Loc Tests Checkpoints Query Editor View State Progress Problems

Fig-39 Trigger Class Test

SOSL

Activity: Search for Applicants

- Scenario:** Create a global search functionality to find applicants across multiple objects like Applicant, Visa Application, and Approval Authority based on a search string.

Execute Use Tooling API

Fig-40 SOSL global search

Batch Apex Class

Activity: Clean-Up Old Applications

- **Scenario:** Implement a batch job to archive or delete visa applications that are more than 5 years old.

(you should create relevant records to show the usage)



Fig-41 BatchApex class

Schedulable Apex

Activity: Monthly Visa Application Summary

- **Scenario:** Schedule a monthly job to calculate and email a summary of visa applications for the month.

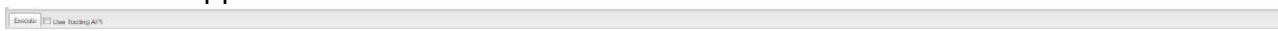


Fig-42 Schedule apex class

Lightning Web Components (LWC)

Activity: Visa Application Dashboard

- **Scenario:** Create a simple LWC to display a dashboard on the home page for Visa Officers, showing the number of applications processed today.

[LWC Example:](#)

- **HTML Template:** Display blocks for total applications, approved, and denied.
- **JavaScript:** Use wire service to fetch data from Apex controller.
- **Apex Controller:** SOQL query to retrieve today's application statistics.

```

Developer Console - Google Chrome
prodapt-5c-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage
File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < >
CountryUpdateTrigger.apxc [ ] CountryUpdateTriggerTest.apxc [ ] VisaApplicationDashboardController.apxc [ ] MonthlyVisaApplicationSummary.apxc [ ] DeleteOldVisaApplicationsBatch.apxc [ ] VisaDashboardController.apxc [ ]
Code Coverage: None ▾ API Version: 59 ▾
1 public class VisaDashboardController {
2
3     public class ApplicationStatistics {
4         @AuraEnabled public Integer totalApplications;
5         @AuraEnabled public Integer approvedApplications;
6         @AuraEnabled public Integer deniedApplications;
7
8         public ApplicationStatistics(Integer totalApplications, Integer approvedApplications, Integer deniedApplications) {
9             this.totalApplications = totalApplications;
10            this.approvedApplications = approvedApplications;
11            this.deniedApplications = deniedApplications;
12        }
13    }
14
15    @AuraEnabled(cacheable=true)
16    public static ApplicationStatistics getApplicationStatistics() {
17        Date today = Date.today();
18
19        Integer totalApplications = [SELECT COUNT() FROM Visa_Application__c WHERE Application_Date__c = :today];
20        Integer approvedApplications = [SELECT COUNT() FROM Visa_Application__c WHERE Application_Date__c = :today AND Status__c = 'Approved'];
21        Integer deniedApplications = [SELECT COUNT() FROM Visa_Application__c WHERE Application_Date__c = :today AND Status__c = 'Denied'];
22
23        return new ApplicationStatistics(totalApplications, approvedApplications, deniedApplications);
24    }
25
26 }

```

Fig-43 Apex Controller class



Fig-44 VisaDashboardView.html



Fig-45 VisaDashboardView.js

- Home page of visa management system

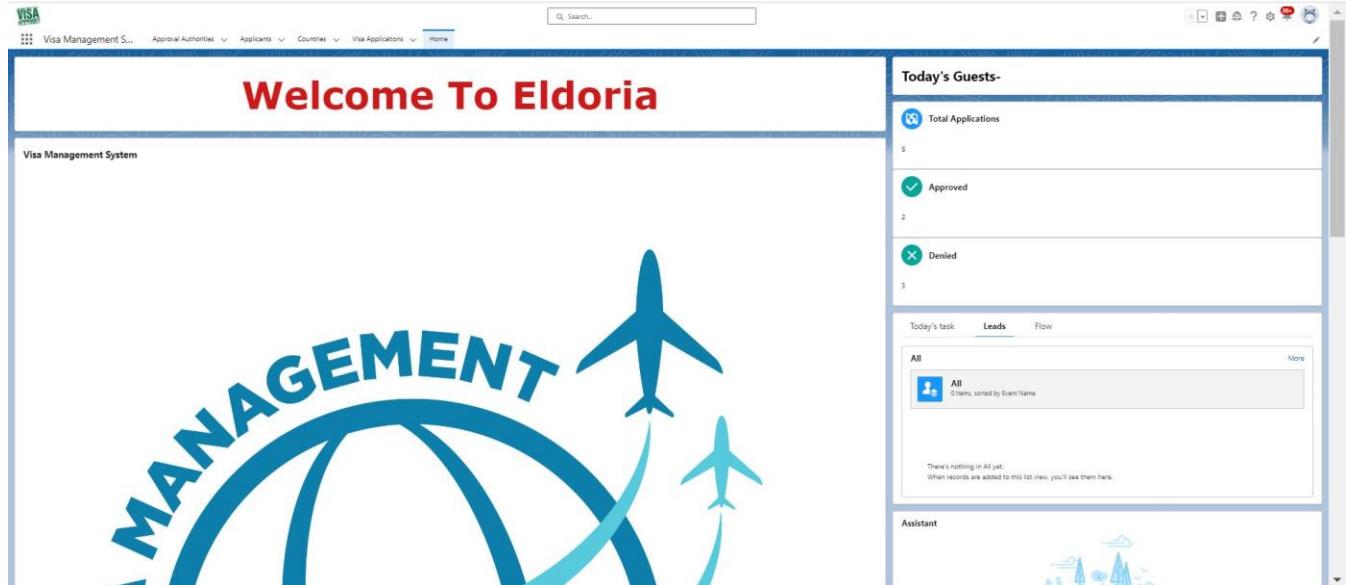


Fig-46 Home page