HOSPITAL MANAGEMENT SYSTEM

SAYANTAN MONDAL

SUNNY KUMAR

SUNDAR A

SRIVATHSAN S

SHUBHAM YADAV

Purpose of the Document

The purpose of this document is to identify the business requirements for the project Hospital Management Application. This document also provides the proposed solution for the same.

Project Requirement

1.1 Finding Doctors

1.2 Booking Appointment

1.3 Patient registration

1.4 Department and Specialties

1.5 Medical test and reports

1.6 patient testimonials

PROJECT FUNCTIONAL AND TECHNICAL DETAILS

Scenario 1 :create below custom objects and fields with various data types and relationship.

List of Custom Objects

Doctor

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	First Name	**text**		
2	Last Name	**text**		
3	Department	**picklist**	Neurology , Cardiology , Dermatology , Radiology, Surgery, Orthopedics	
4	Specialty	**picklist**	Allergy and Immunology , Emergency Medicine , Geriatrics , Infectious Disease, Obstetrics and Gynecology	
5	Email	**email**		
6	Phone	**phone**		
7	Doctor id	**Auto number**		

Patient

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	First Name	**Text**		
2	Last Name	**Text**		
3	Email	**Email**		
4	Phone	**Phone**		
5	Address	**Text area**		
6	Date of Birth	**Date**		
7	Gender	**picklist**	**Male,female,others**	
8	Patient id	**Auto number**		

Appointment

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	Patient			lookup to Patient
2	Doctor			lookup to doctor
3	**Date/Time**	**datetime**		
4	**Duration**	**number**		
5	**Reason**	**Text area**		
6	**Appointment id**	**Auto number**		
7	**Appointment status**	**picklist**	**Approved ,declined ,pending**	

Department

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	**Department Name**	**text**		
2	**Doctor name**	**text**		**Lookup to doctor**

Medical Test Report

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	**name**	**Auto number**		

2	**Report cost**	**currency**		
3	**Test type**	**picklist**	**Blood,sugar,BP,eyes**	
4	**Test date**	**date**		
5	**result**	**Text area**		
6	**Doctor**			**Lookup to doctor**
7	**Patient**			**Lookup to patient**
8	**status**	**picklist**	Test Done, Analyzing Test, Report ready	
9	**Report id**	**Auto number**		

Medical Encounter

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	**Patient**			lookup to Patient
2	**Doctor**			lookup to doctor
3	**Date**	**date**		
4	**Medical test**	**picklist**		
5	**Treatment**	**Text area**		
6	**prescription**	**Text area**		
7	**Medical encounter id**	**Auto number**		

Specialty

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	Specialty Name	Text		
2	Doctor name	Text		Lookup to doctor

BILL

SNo	Field Name	Field Types	Default Values / Required	Relationship
1	Patient Name			Lookup to Patient object
2	Doctor Name			Lookup to Doctor object
3	Treatment Cost	currency		
4	Test Report Cost	currency		Lookup to Medical Test object
5	Doctor Fees	currency		
6	Total Bill Amount	formula		calculates the sum of Treatment

				Cost, Test Report Cost, and Doctors Fees Cost
7	Bill Number	**Auto number**		
8	Bill Date	**Date**		
9	Payment Status	**picklist**	**Paid,unpaid**	
10	Payment Date	**Date**		
12	Notes	Long Text Area		
13	Bill id	**Auto number**		

Scenario 2: create below profiles.

Doctor

Patient

Bill manager

Lab staff

Scenario 3 :create 4 users.

VALIDATION RULE

Scenario 3: -

Create a Validation Rule to Validate that the Email Address is in right format.

1. First Go to Patient Object.
2. Go To Validation Rule
3. Click On New
4. Rule Name – Email Address Validation
5. Tick the active Checkbox
6. NOT (REGEX(Email_ID__c , "[a-zA-Z0-9. _%+-] +@[a-zA-Z0-9.-] +. [a-zA-Z] {2,}")) W

Write This Code in the Error Condition Formula Section (Match the Field name according to your field names)

7. Error Message – Email ID Should Be in Correct Format

Scenario 4: -

Create a Validation rule to Ensure that the Date OF Birth of the patient Is not a future Date

1. First Go to Patient Object.
2. Go To Validation Rule

3. Click On New
4. Rule Name – DOB Validation

5. Tick the active Checkbox
6. AND(
   NOT(ISBLANK(Date_of_Birth__c)),
   Date_of_Birth__c > TODAY ()
)

Write This Code in the Error Condition Formula Section (Match the Field name according to your field names)

7. Error Message – Give Correct Date of Birth

Scenario 5: -

C. Create a Validation rule to Ensure that the Date OF Appointment is a Future Date.

1. First Go to Patient Object.
2. Go To Validation Rule
3. Click On New
4. Rule Name – Date of Appointment Validation
5. Tick the active Checkbox
6. Date_Time__c < NOW ()

Write This Code in the Error Condition Formula Section (Match the Field name according to your field names)

7. Error Message – Select Correct date of Appointment.

Scenario 6: -

OWD SETTINGS

1. Go to Setup
2. In the Quick Find box type Sharing setting
3. Click on Edit
4. Change Default Internal Settings to Public Read Only for All the Objects Needed in our Project (Doctor, Patient, Appointment, Department, Specialty, Medical Test Report, Medical Encounter, Bill)

PERMISSION SETS

Scenario 7: -

DOCTOR PROFILE PERMISSIONS: -

1. Go to Setup
2. In the Quick Find box type Permission Sets
3. Click on New

4. Label – Doctors Permission

   License – Salesforce

5. From The apps section click on Assigned Apps
6. Select Our Hospital App Then Click add then Save
7. Click on the Down arrow beside Assigned apps and from the drop-down list select object permission.
8. Click on objects, click edit and give them permission mentioned below.

Appointment Object

9. Tab Setting: - Click on Both Available and Visible
10. Object Permission: – Read, View All
11. Field Permission: -

    Read Access – All the Fields

    Edit Access – Appointment Status

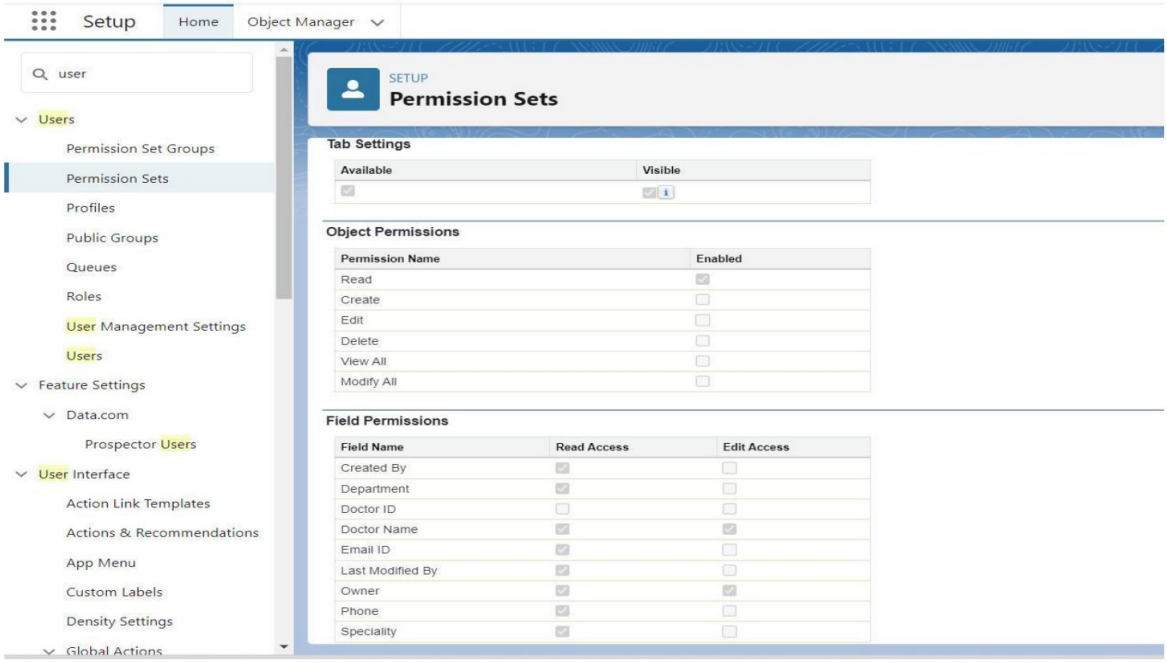
12. Save

Special Note – A Doctor can Read the Appointment details and can edit the appointment status only.

Bills Object

Doctor has No Access to this Object

Doctor Object



Special Note – A Doctor can Read his record but cannot edit it or create it. Doctor records are only created by the Administrator.

Department Object

Tab Setting: - Click on Both Available and Visible

Object Permission: – Read, View All

Field Permission: -

Read Access – All the Fields

Specialty Object

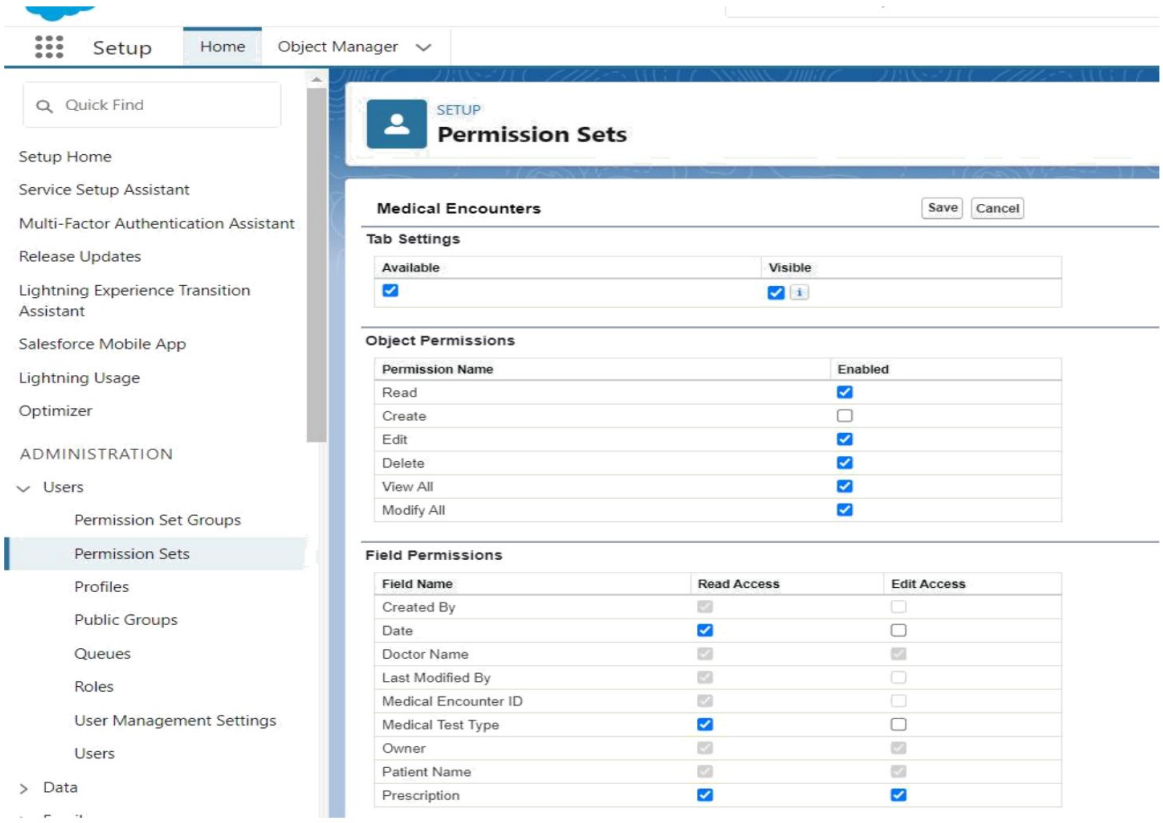
Tab Setting: - Click on Both Available and Visible

Object Permission: – Read, View All

Field Permission: -

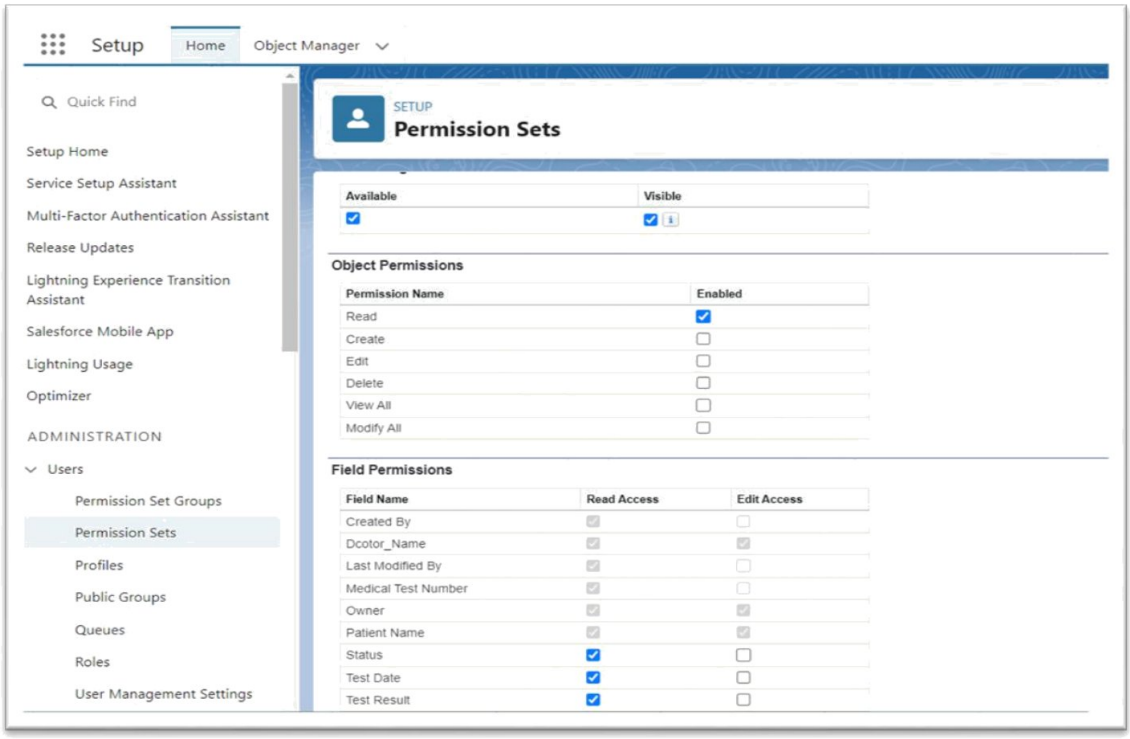
Read Access – All the Fields

Medical Encounter



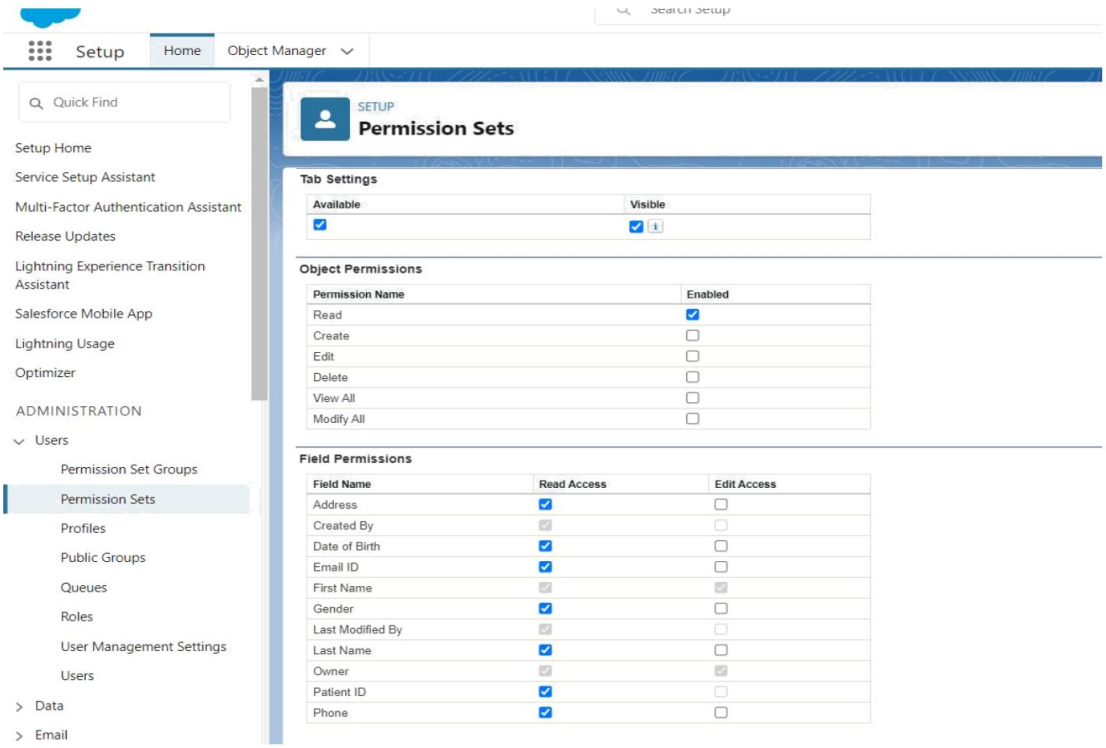
Special Note – A Doctor can Read the medical Encounter. And can edit the Prescription and Date.

Medical Test Report



Special Note - Doctor Can See Medical Test Report. But He Can't Edit Anything.

Patient Object



Special Note - Doctor Can See Patient Record but He Can't edit anything.

Scenario 8: -

PATIENT PROFILE PERMISSION: -

Go to Setup

In the Quick Find box type Permission

Sets Click on New

Label – Patient Permission

License – Salesforce

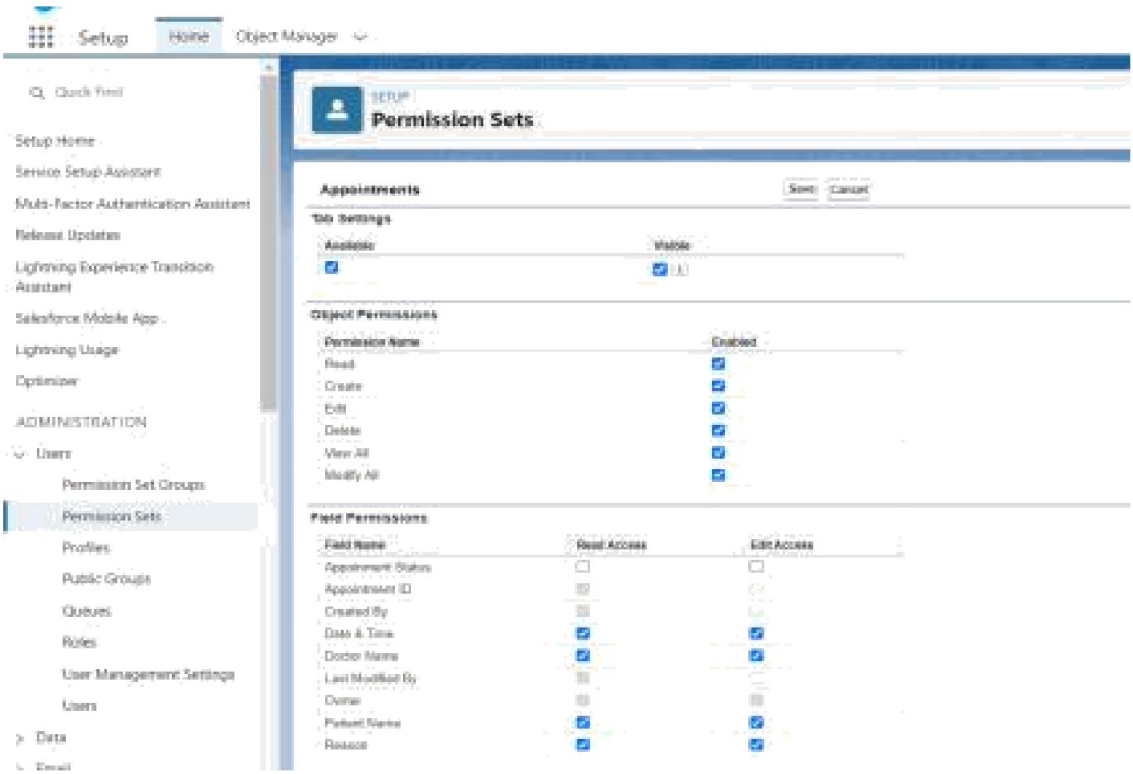
From The apps section click on Assigned Apps

Select Our Hospital App Then Click add then Save

Click on the Down arrow beside Assigned apps and from the drop-down list select object permission.

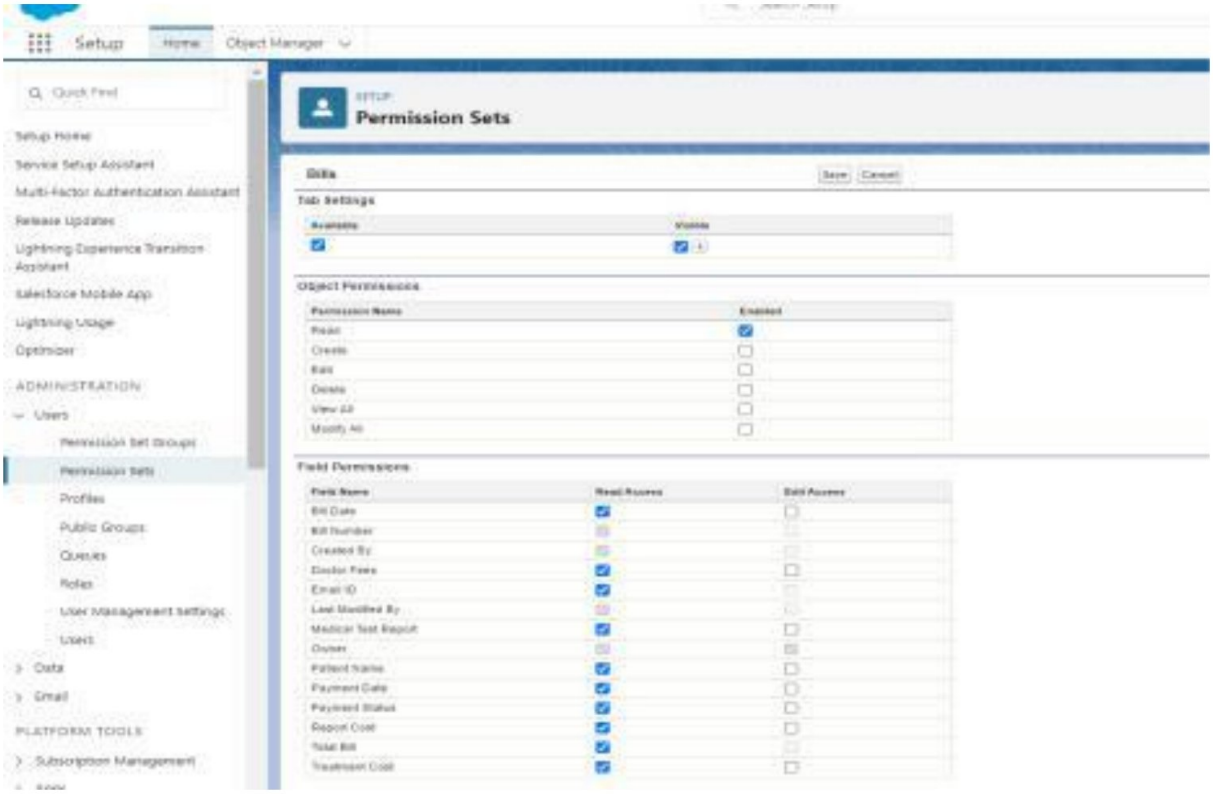
Click on objects, click edit and give them permission mentioned below.

Appointment Object Permission



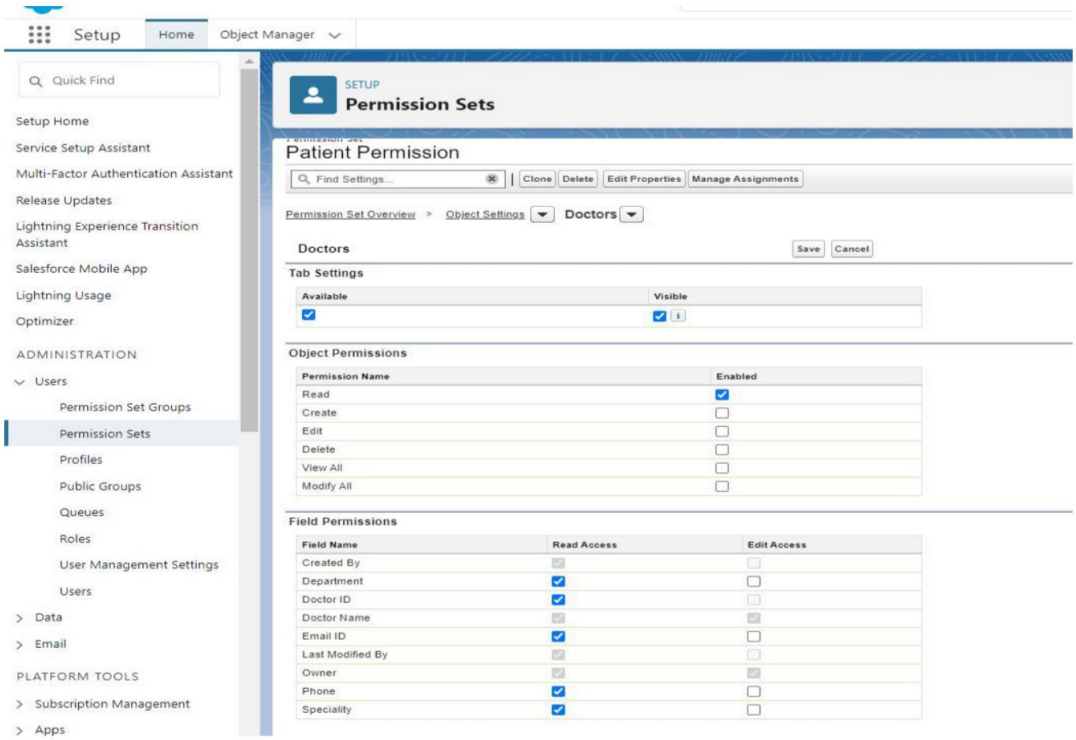
Special Note – Patient can Create, Del and read records in the appointment object.

Bills Object



Special Note – Patient only have read access to the bills.

Doctor Object



Special Note: - Patient can only see Doctor Details

Department Object

Tab Setting: - Click on Both Available and Visible

Object Permission: – Read, View All

Field Permission: -

Read Access – All the Fields

Specialty Object

Tab Setting: - Click on Both Available and Visible

Object Permission: – Read, View All

Field Permission: -

Read Access – All the Fields

Medical Encounter Object

Tab Setting: - Click on Both Available and Visible

Object Permission: – Read, View All

Field Permission: -

Read Access – All the Fields

Medical Test Report Object

Tab Setting: - Click on Both Available and Visible

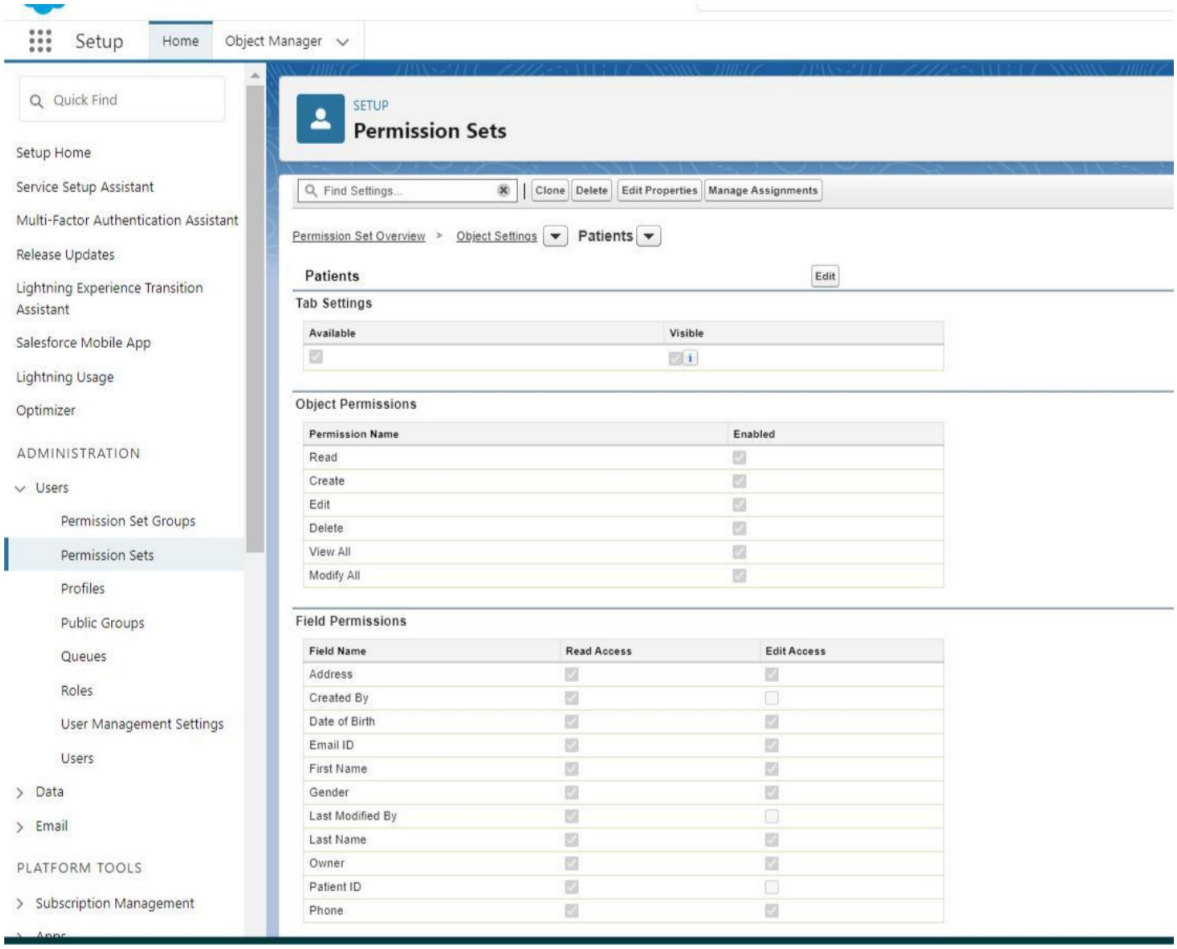
Object Permission: – Read, View All

Field Permission: -

Read Access – All the Fields

(Same as Bills Object)

Patient Object



Special Note – Patient can do everything in the patient object.

Scenario 9: -

LAB STAFF PROFILE PERMISSION

Go to Setup

In the Quick Find box type Permission

Sets Click on New

Label – Patient Permission

License – Salesforce

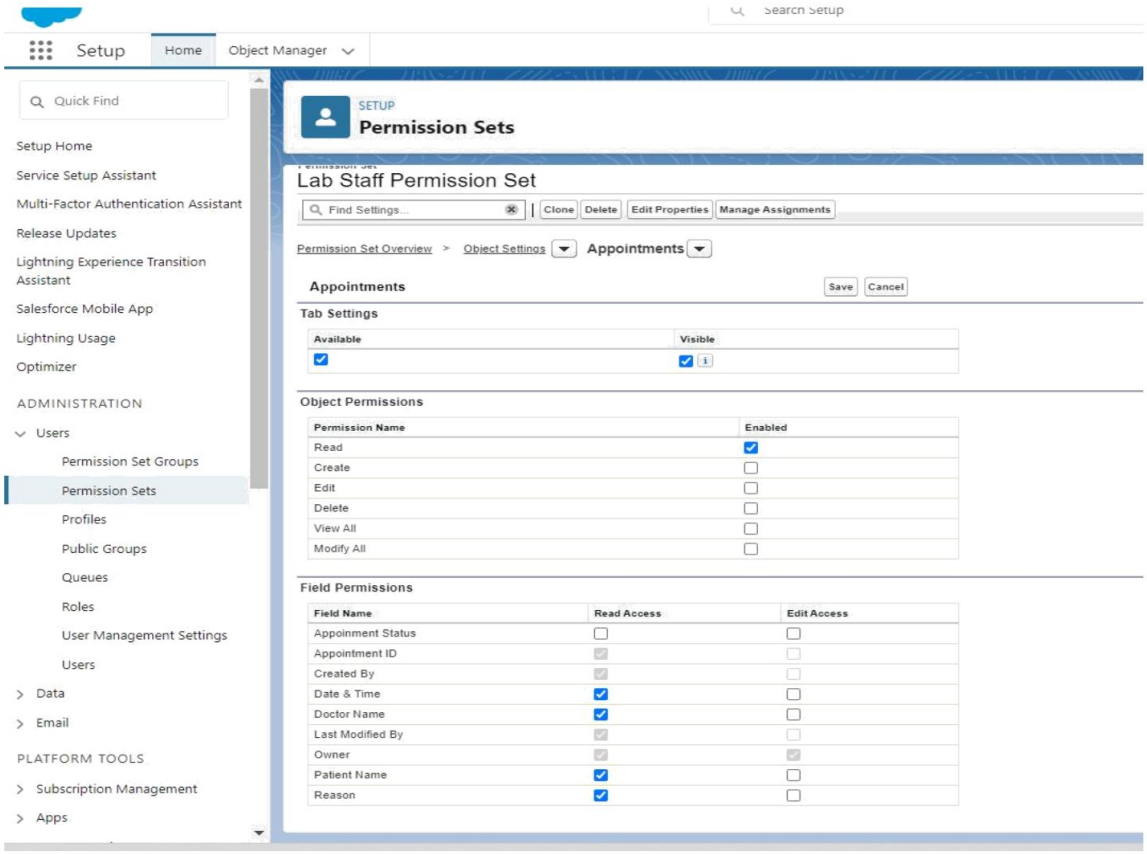
From The apps section click on Assigned Apps

Select Our Hospital App Then Click add then Save

Click on the Down arrow beside Assigned apps and from the drop-down list select object permission.

Click on objects, click edit and give them permission mentioned below.

Appointment Object



Special Note: - Staff Can Only See the appointment.

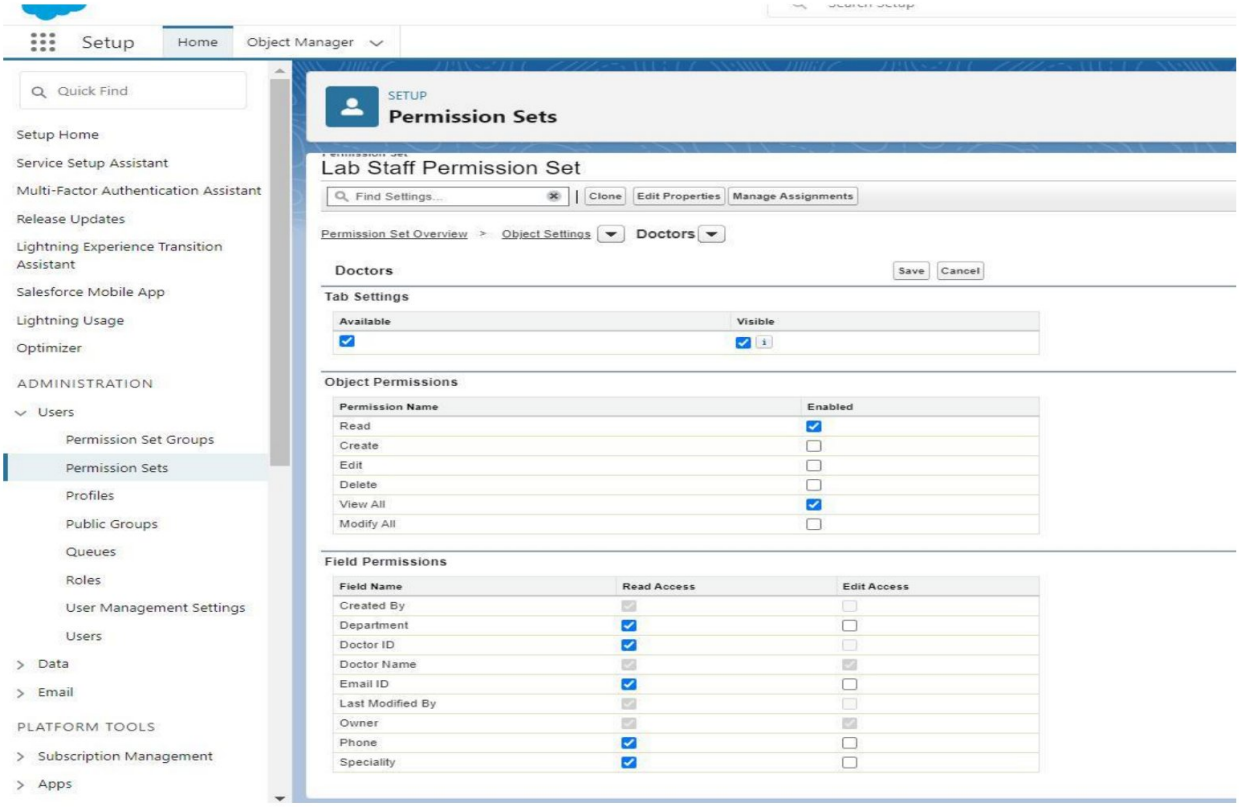
Department Object

NO ACCESS

Specialty Object

NO ACCESS

Doctor Object



Special Note: - Staff can only see the Doctors details.

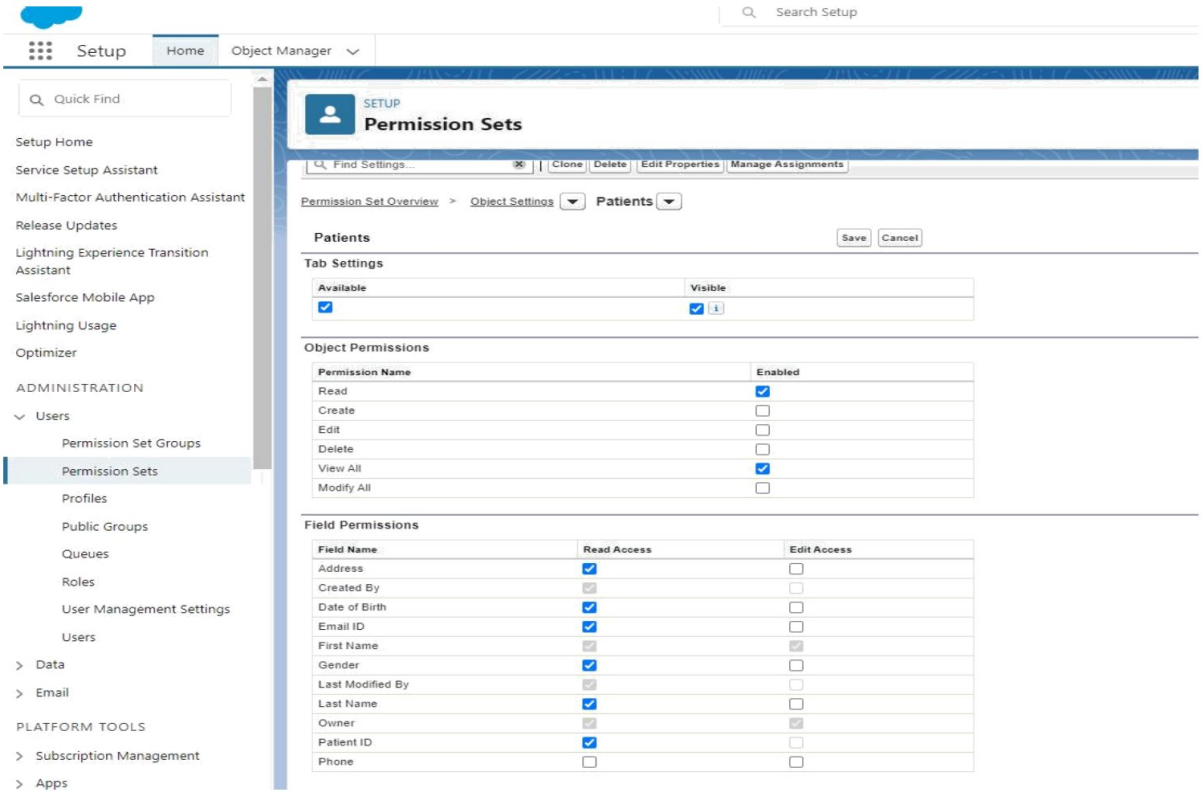
Bill Object

NO ACCESS

Medical Encounter

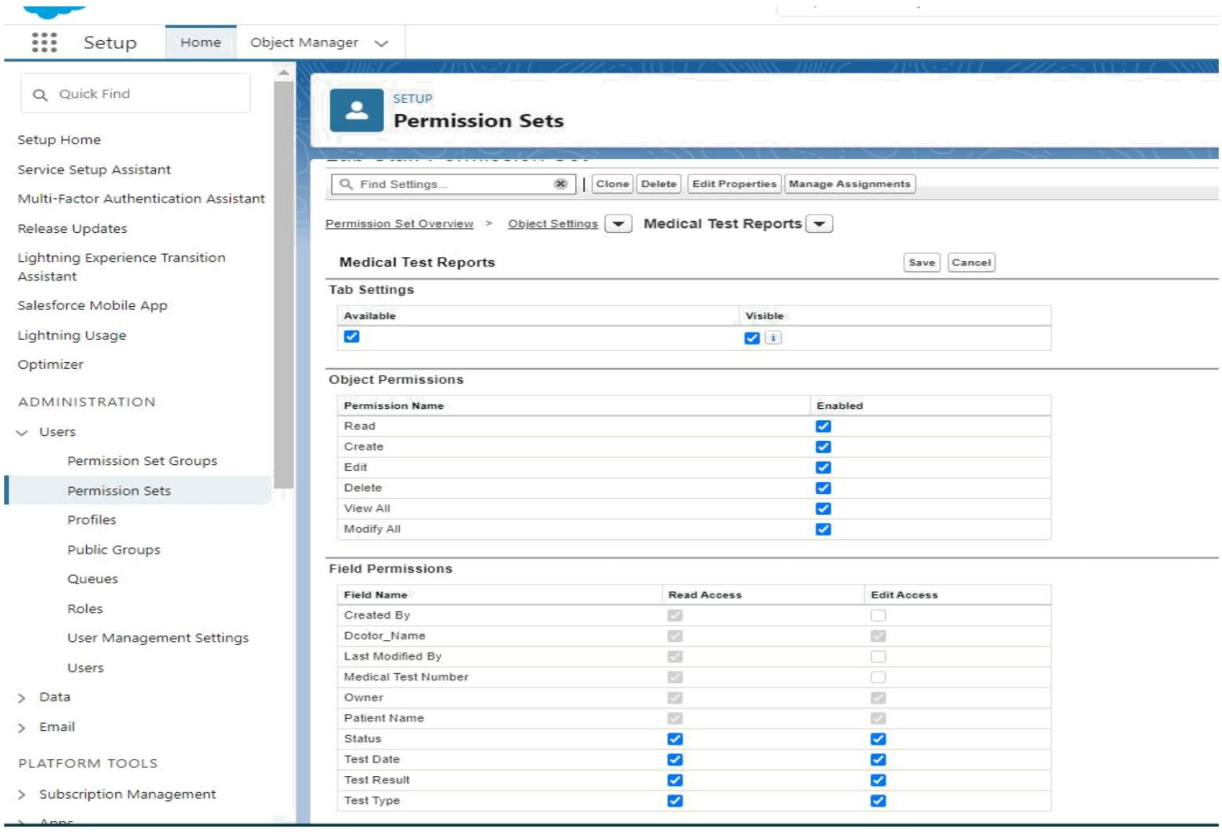
NO ACCESS

Patient Object



Special Note: - Staff only have read access for patient profile.

Medical Test Report



Special Note – Staff can do everything in medical test report object.

Scenario 9: -

Bill Manager Profile Permission Set

Go to Setup

In the Quick Find box type Permission Sets

Click on New

Label – Patient Permission

License – Salesforce

From The apps section click on Assigned Apps

Select Our Hospital App Then Click add then Save

Click on the Down arrow beside Assigned apps and from the drop-down list select object permission.

Click on objects, click edit and give them permission mentioned below.

Department Object

NO ACCESS

Specialty Object

NO ACCESS

Doctor Object

NO ACCESS

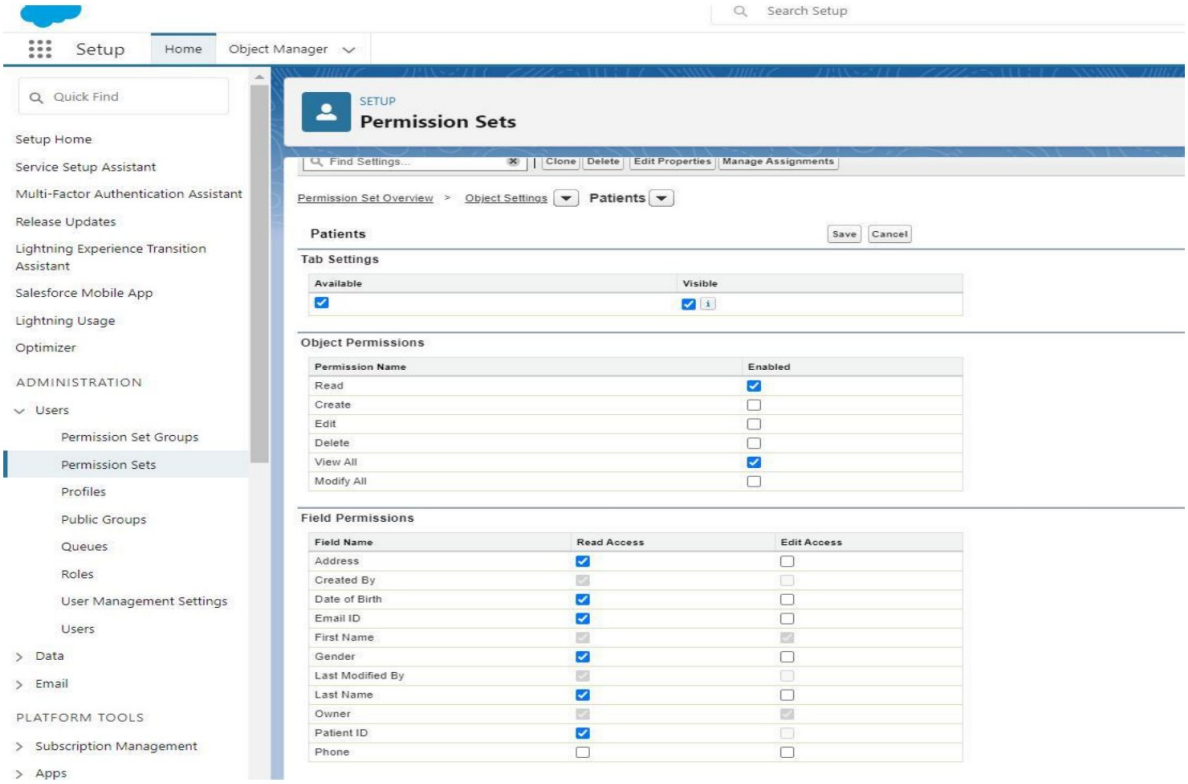
Medical Test Report Object

NO ACCESS

Appointment Object

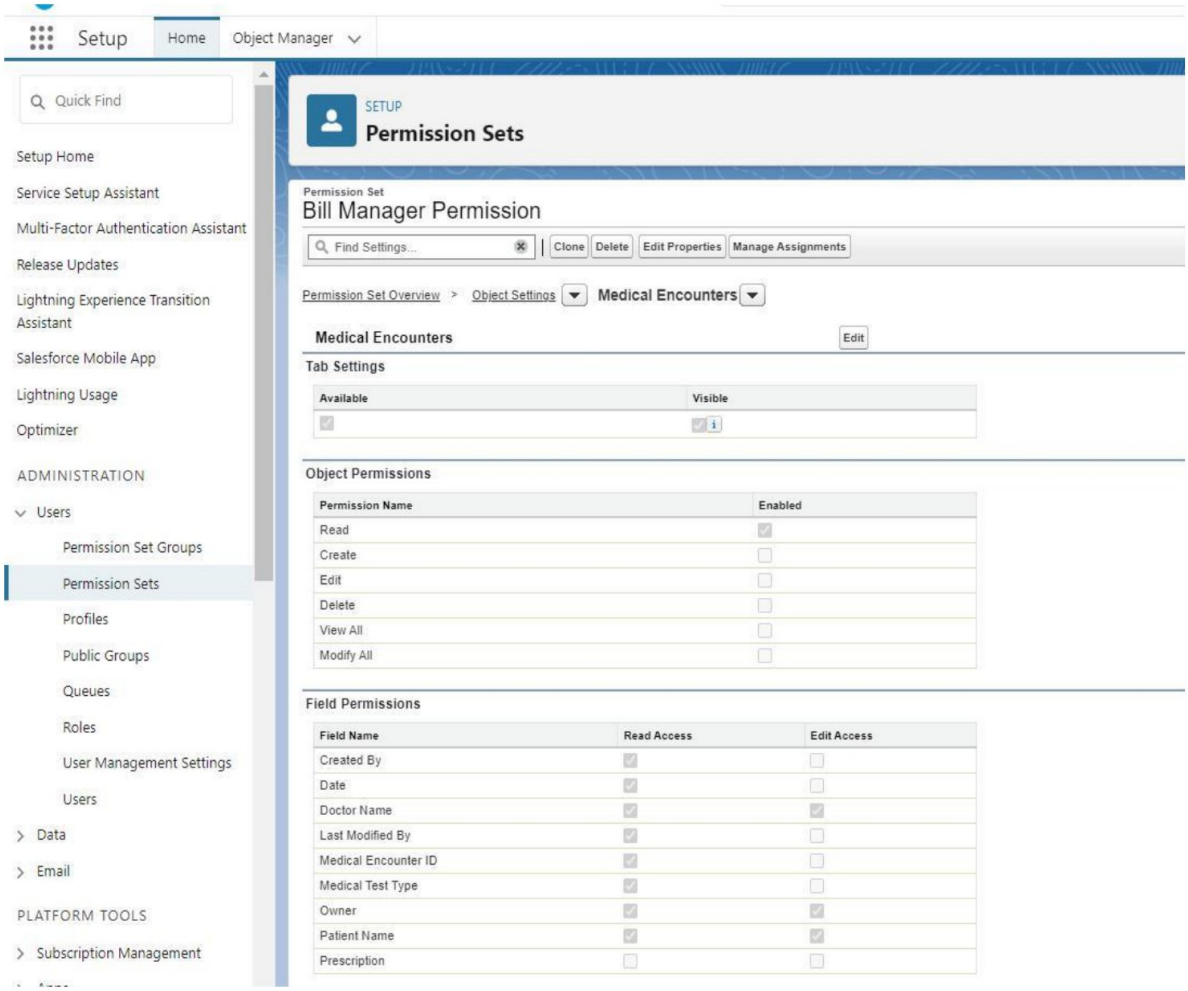
NO ACCESS

Patient Object

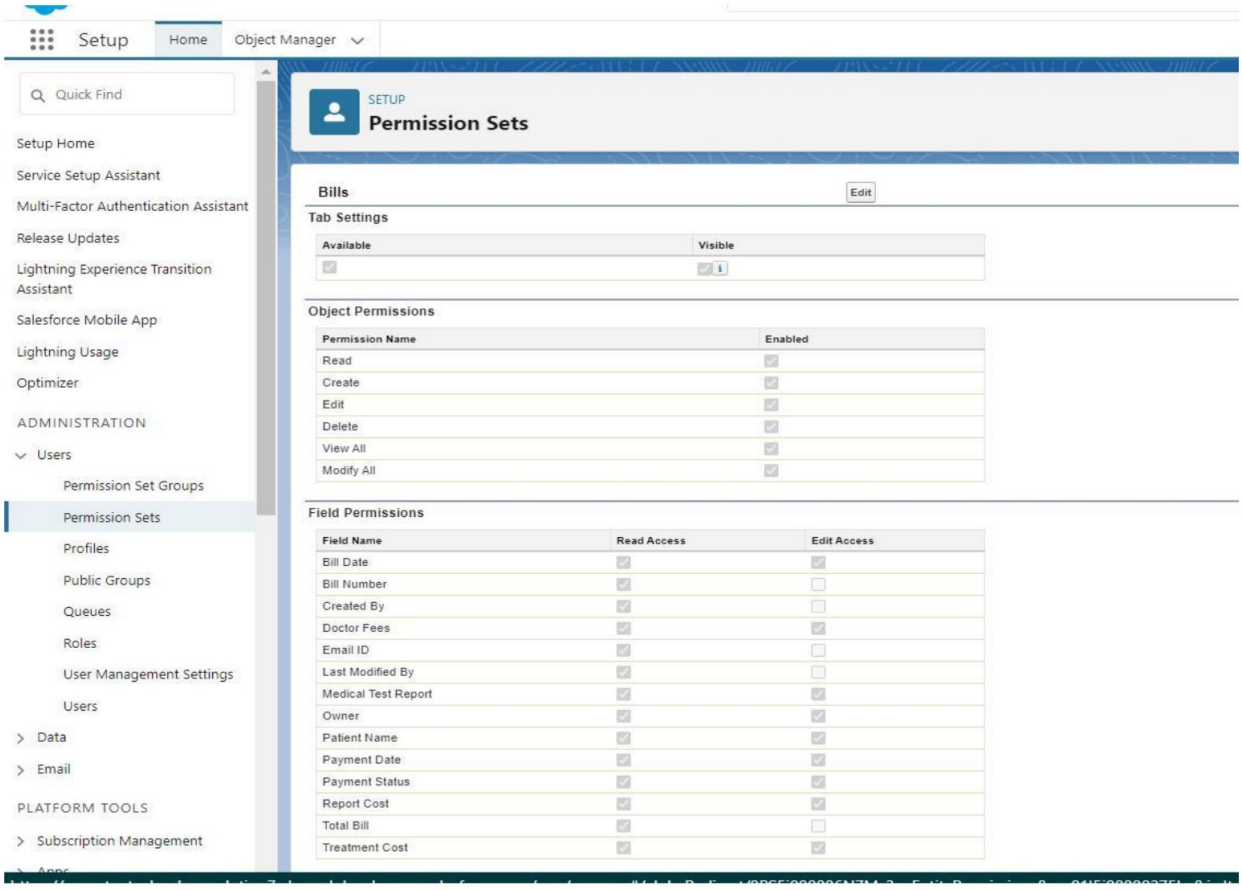


Special Note – Bill Manager can read patient records.

Medical Encounter



Bill Object
s



Scenario 10- create an App in classic ,

Add a logo in a tab called Documents

Logo size should not be more than 20 kb

Step 1: Click on avatar and select Switch to Salesforce Classic first.

Step 2: Click on Setup.

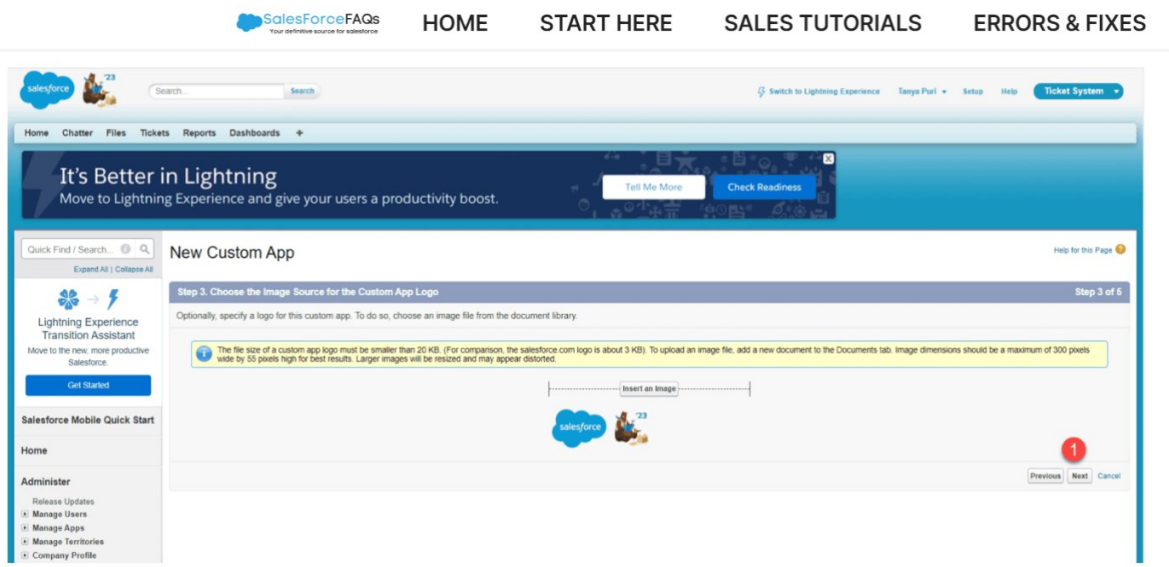
Step 3: , enter the keyword apps in the Quick Find / Search Box.

Step 4: Click on Apps located under Create < Build.

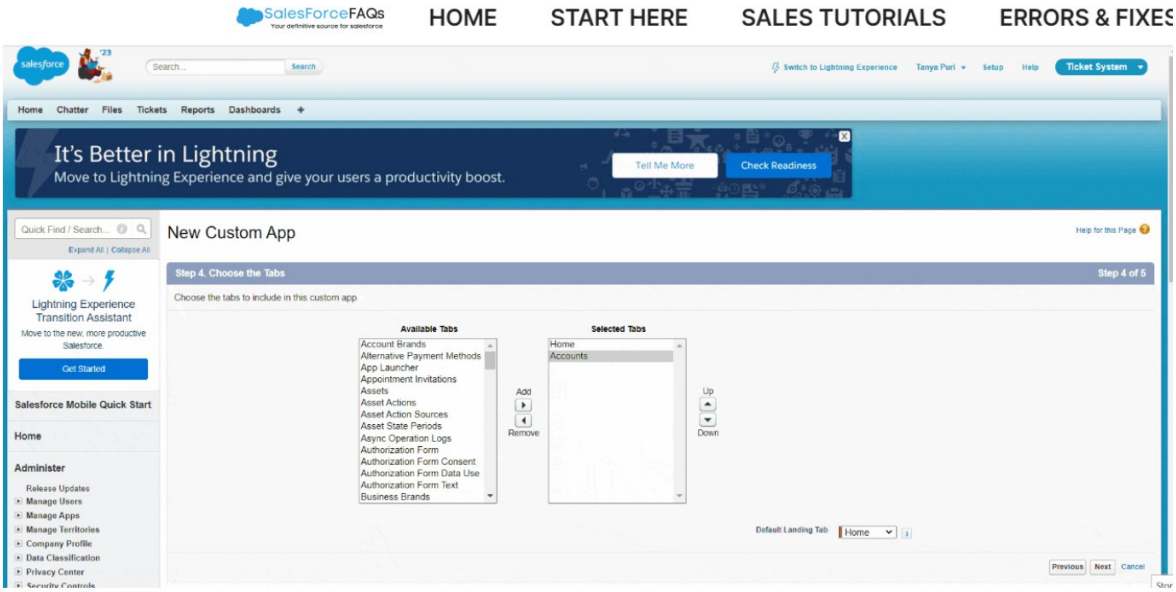
Step 5: Select the type of app you want to create. Here, I select the custom app radio button. Then, click on Next.

Step 6: Provide the label of the app and the app name is automatically populated. If you want to give more description about the application, add to the Description. Then, click on Next.

Step 7: add the log to the application and do the branding, Insert an Image. But make sure, the file size of the custom app logo must be smaller than 20 KB. Then, click on Next.



Step 8: Now, choose the tabs that you want to include in your custom app. Select the specific tabs from the Available Tabs and move them to the Selected Tabs. Then, click on Next.



Step 9: Click on the visible checkbox of the profile, in which you want to visible your new custom app. Then, click save.

Scenario 11: upgrade your app in lightining

From the Home tab in Setup, enter App in the Quick Find box, then select App Manager. Find the Classic app that you want to upgrade in the apps list. Click. , and select Upgrade.

FLOWS

Scenario 12: APPOINTMENT BOOKING FLOW

When an appointment will be booked both doctor and patient will receive an email with appointment id

1.create a new flow

2.Click record trigger flow

3.1) Object - Appointment

2) Trigger - A record is created

3) optimize for - Actions and Related Records

4. click the plus icon

5. Add action

6. In search all actions search for send email and choose it
7. Label – Patient Appointment Booking.

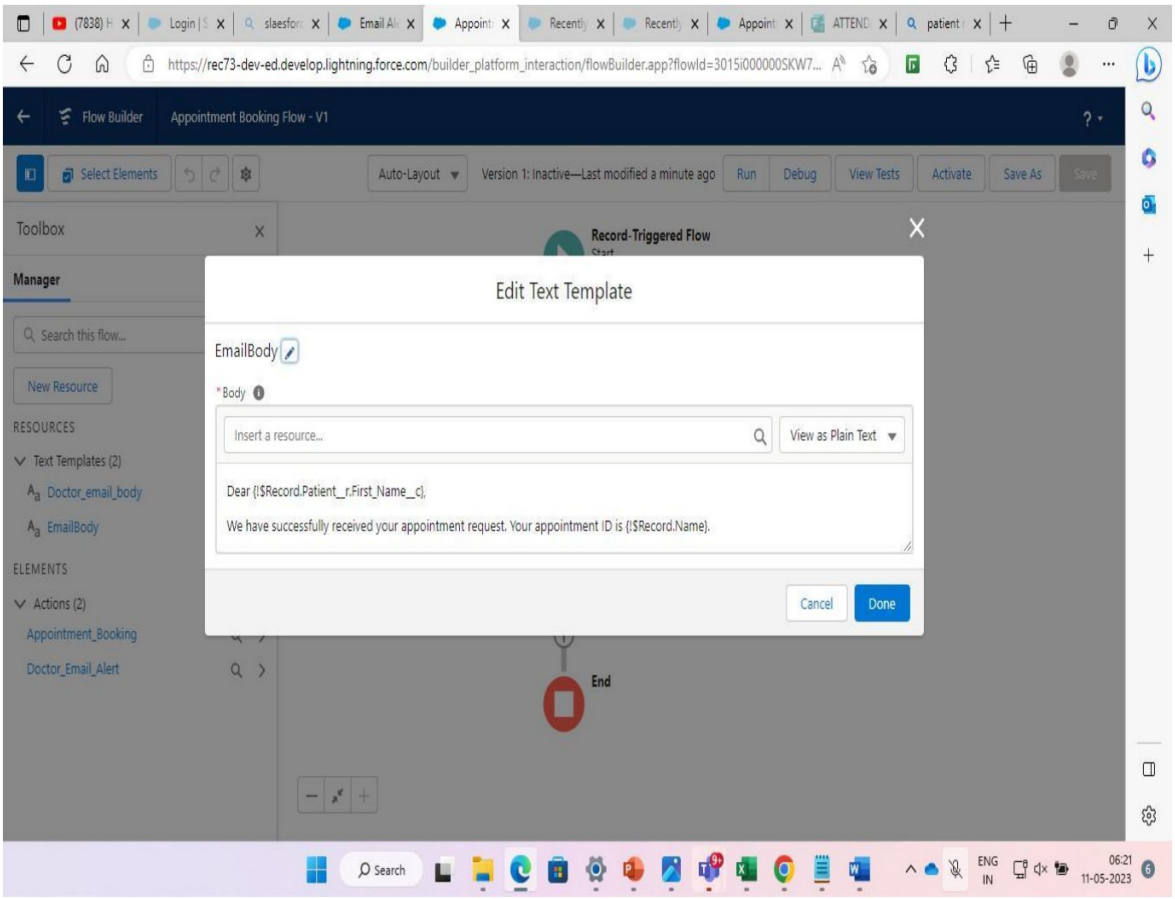
8.Body-New resource-Text Template-Email Body(api name).

In Body:

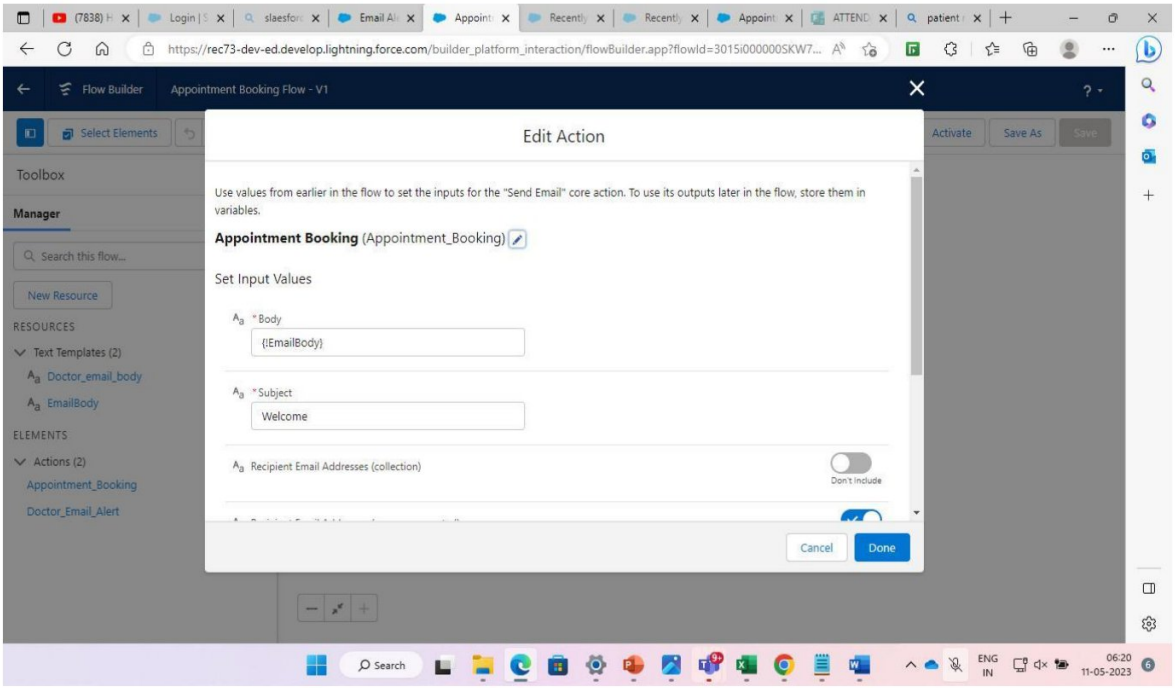
Dear {!$Record.Patient__r.First_Name__c},

We have successfully received your appointment request.

Your appointment ID is {!$Record.Name



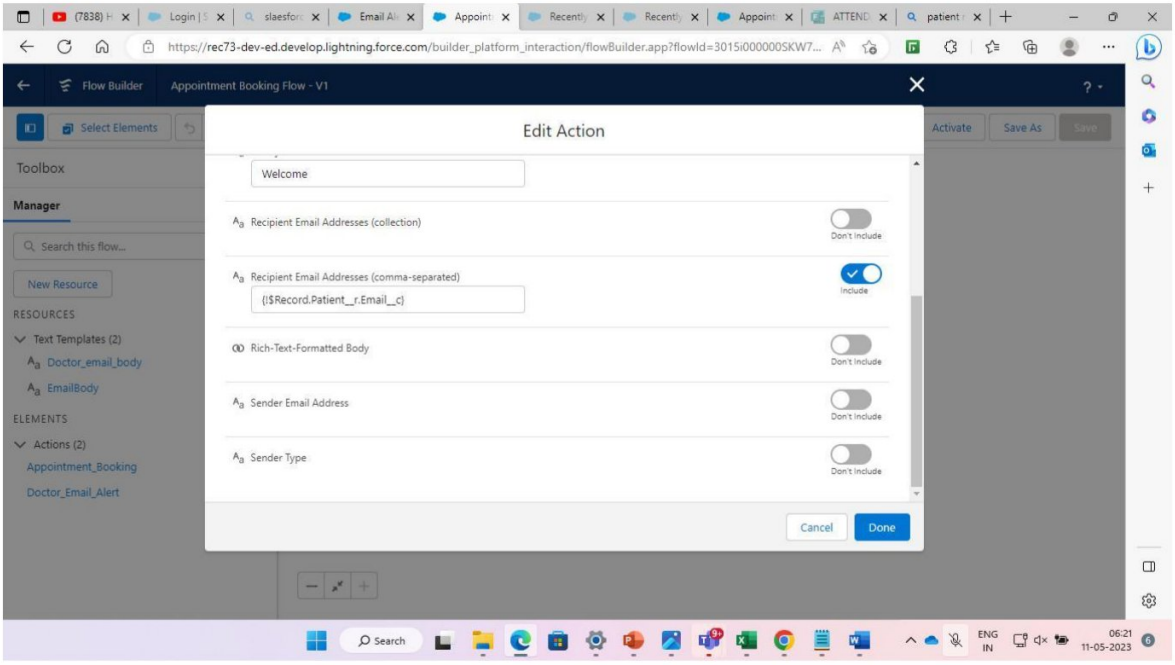
9. In subject type "Welcome".



10.In recipient mail address click include.

11. In recipient email addressses click Record- Appointment-patient-email.

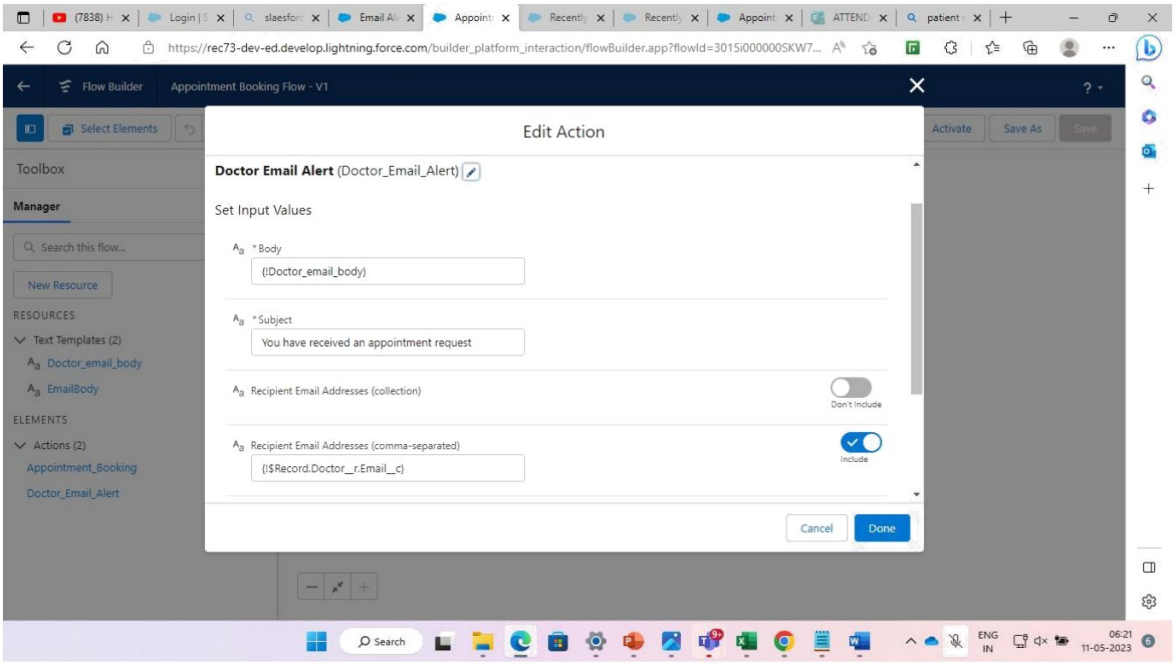
({!$Record.Patient__r.Email__c}).



12. click the plus icon and add action again

13. In search all actions search for send email and choose it.

14. Label – Doctor Email Alert



15.Body-New resource-Text Template-Doctor Email Body(api name).

In Body:

Hello

Dr.{!$Record.Doctor__r.Name}, You have received an appointment from {!$Record.Patient__r.Name} on {!$Record.Date_Time__c}.

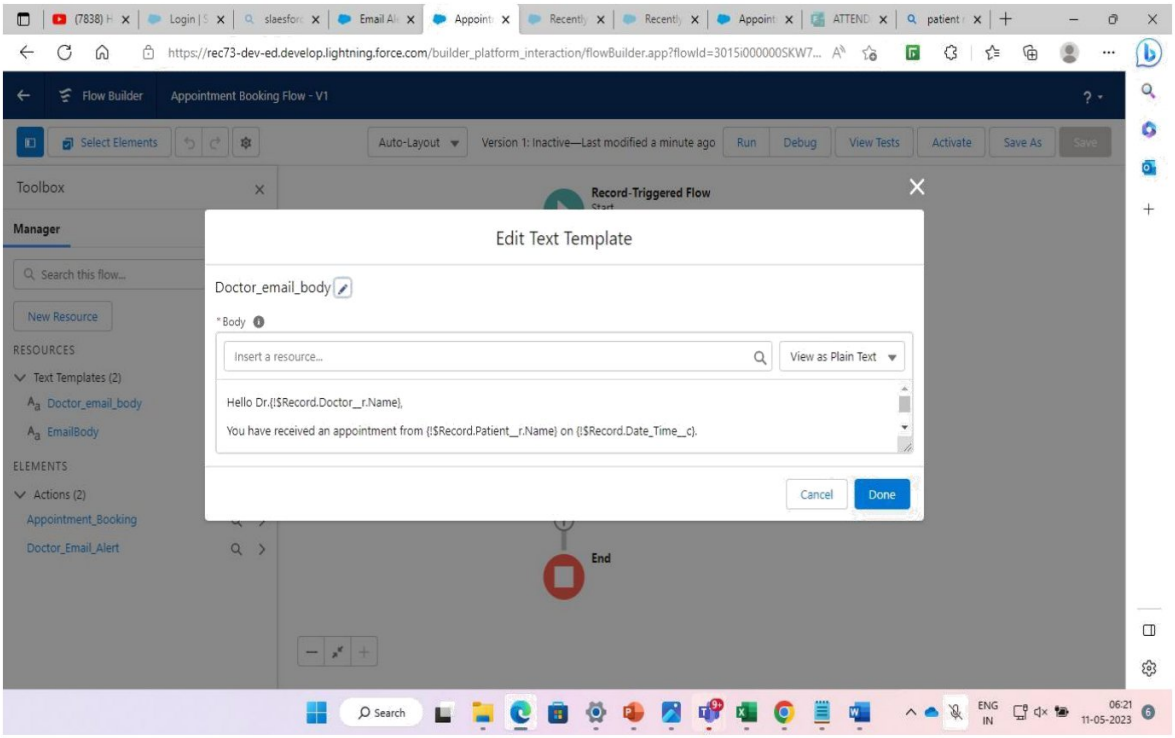
Appointment Details:

Appointment ID: {!$Record.Name}

Reason : {!$Record.Reason__c}

Thanks and Regards

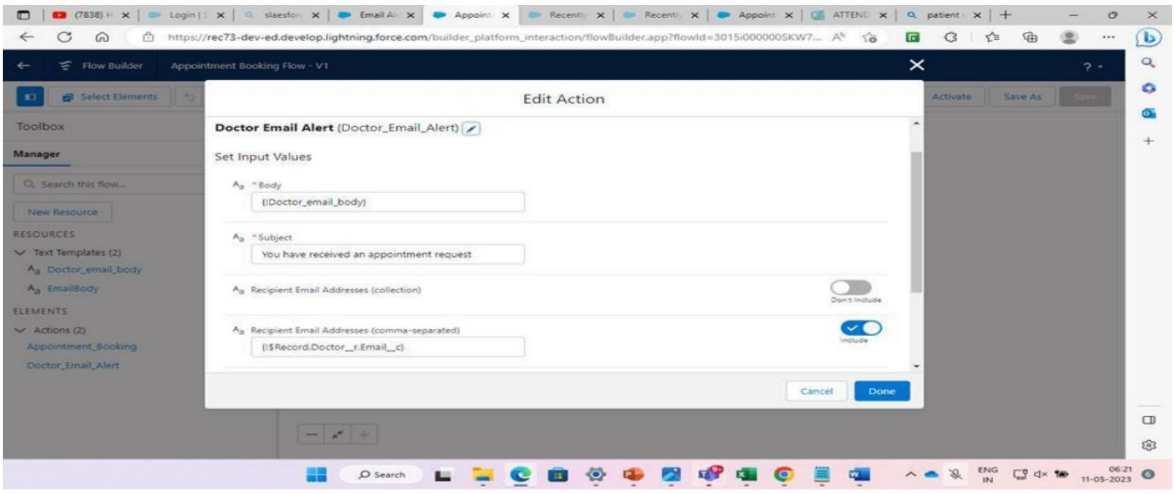
ABC Hospital



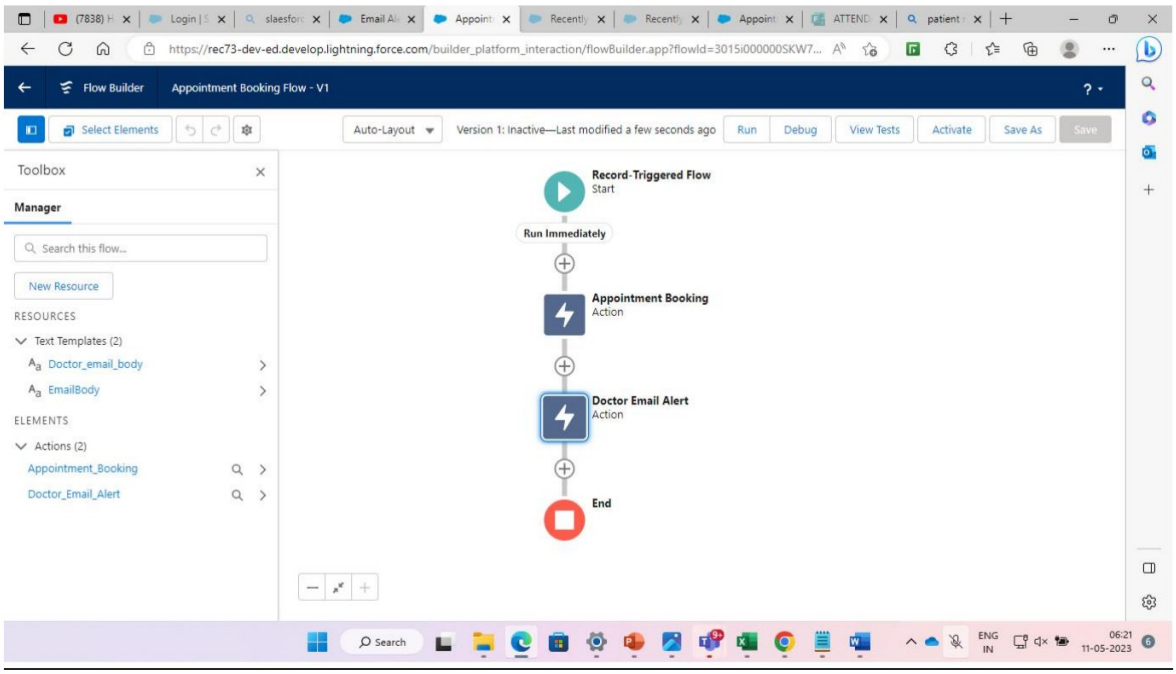
16. In subject type – You have received an appointment request.

17. In recipient email addressses click Record- Appointment-patient-email.

({!$Record.Patient__r.Email__c}).



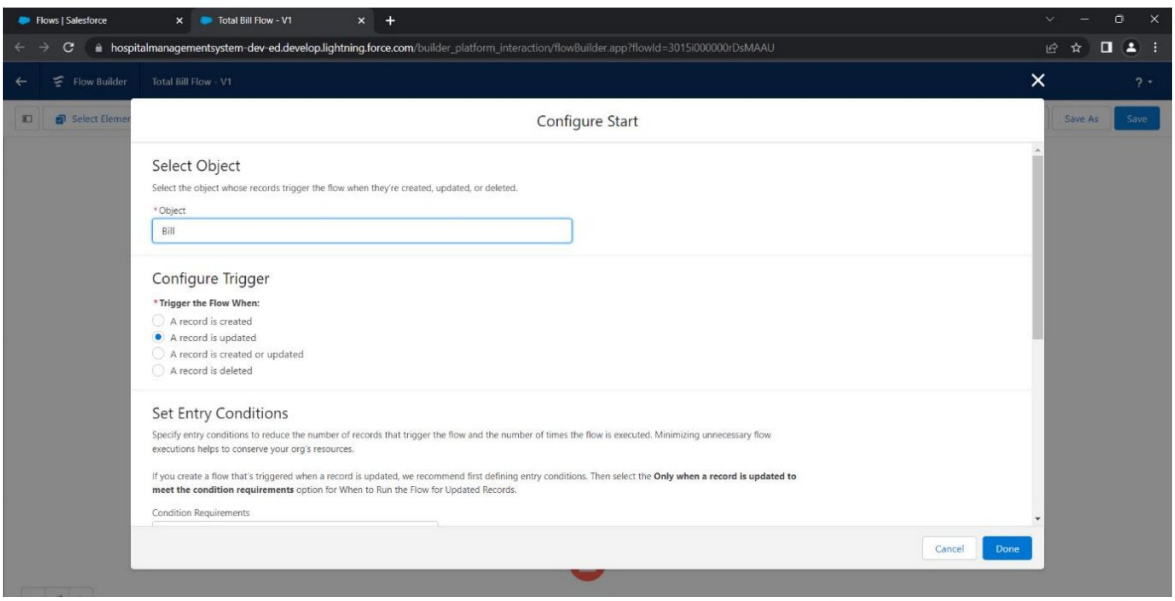
Screenshot of Final Flow:

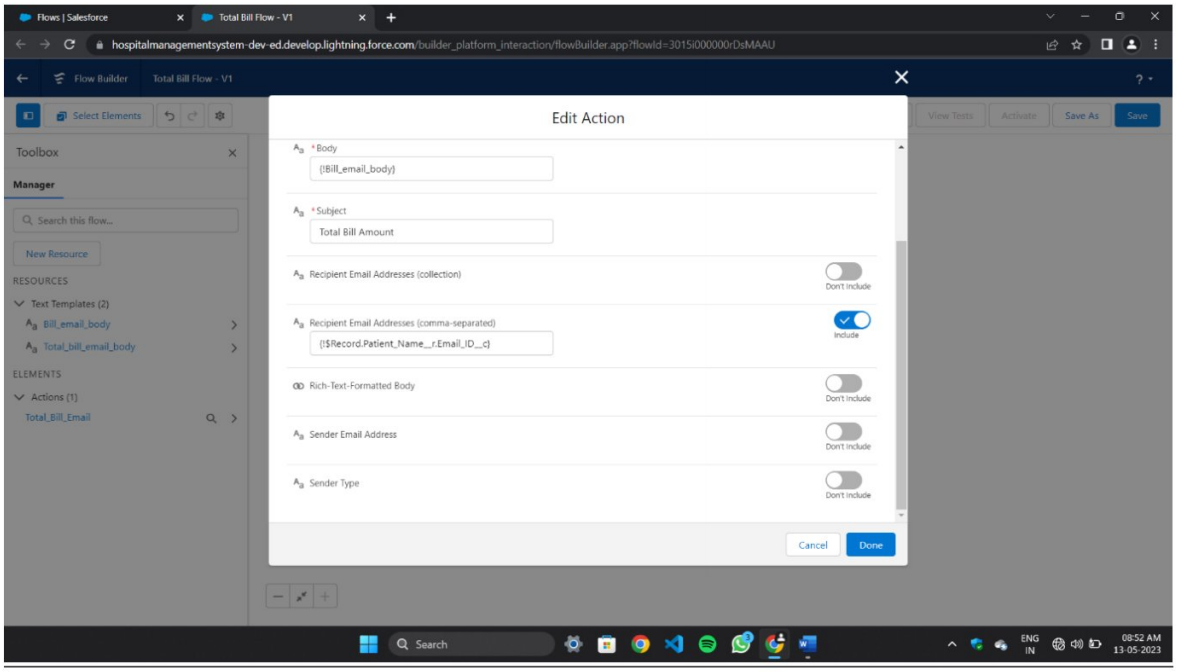


Scenario 13: TOTAL BILL AMOUNT FLOW

When the total bill amount is generated that trigger mail to patient will flowing details

1.Create Trigger Flow





Email Body :

Dear {!$Record.Patient_Name__r.Name},

We hope this email finds you in good health. As per your recent appointment at ABC Hospital, we would like to provide you bill details

Patient Name : {!$Record.Patient_Name__r.Name}

Appoinment Id : {!$Record.Appointment__r.Name}

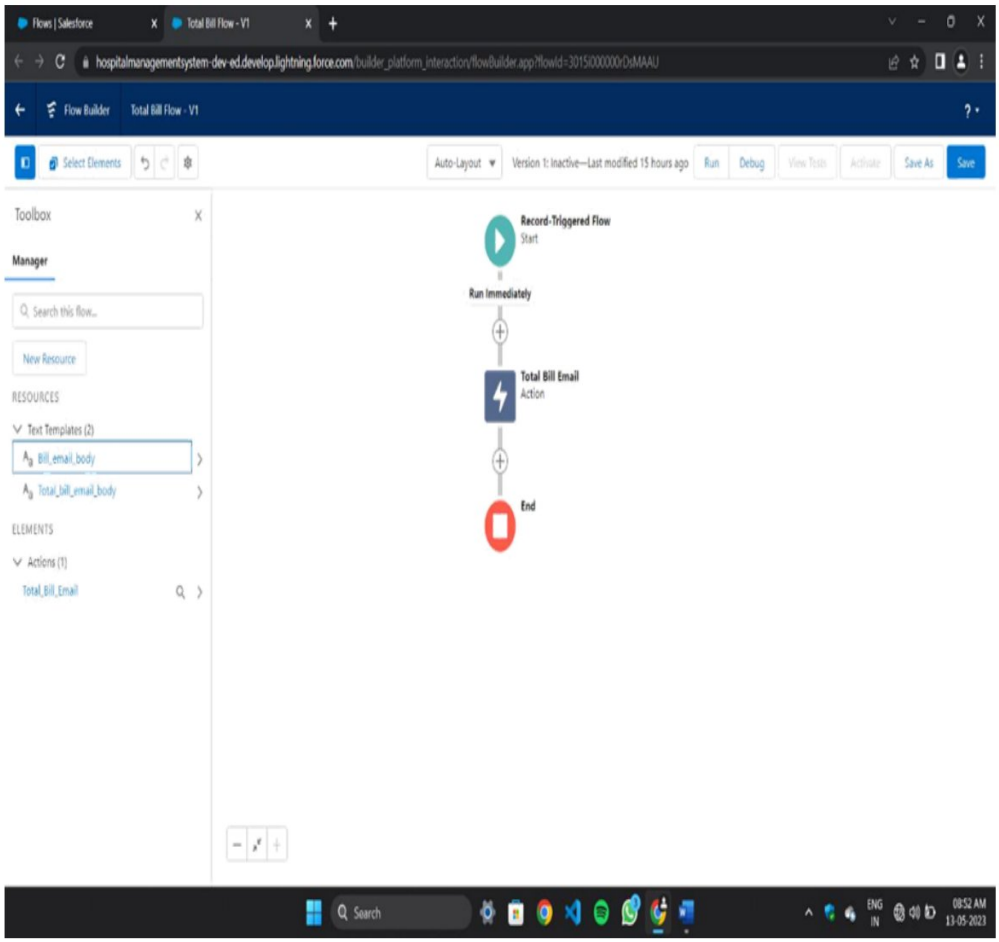
Doctor fee :
{!$Record.Appointment__r.Doctor_Fees__c}

Report Cost :
{!$Record.Medical_Test_Report__r.Report_Cost__c}

Total Bill Amount : {!$Record.Total_Bill_Amount__c}

Thanks and Regards

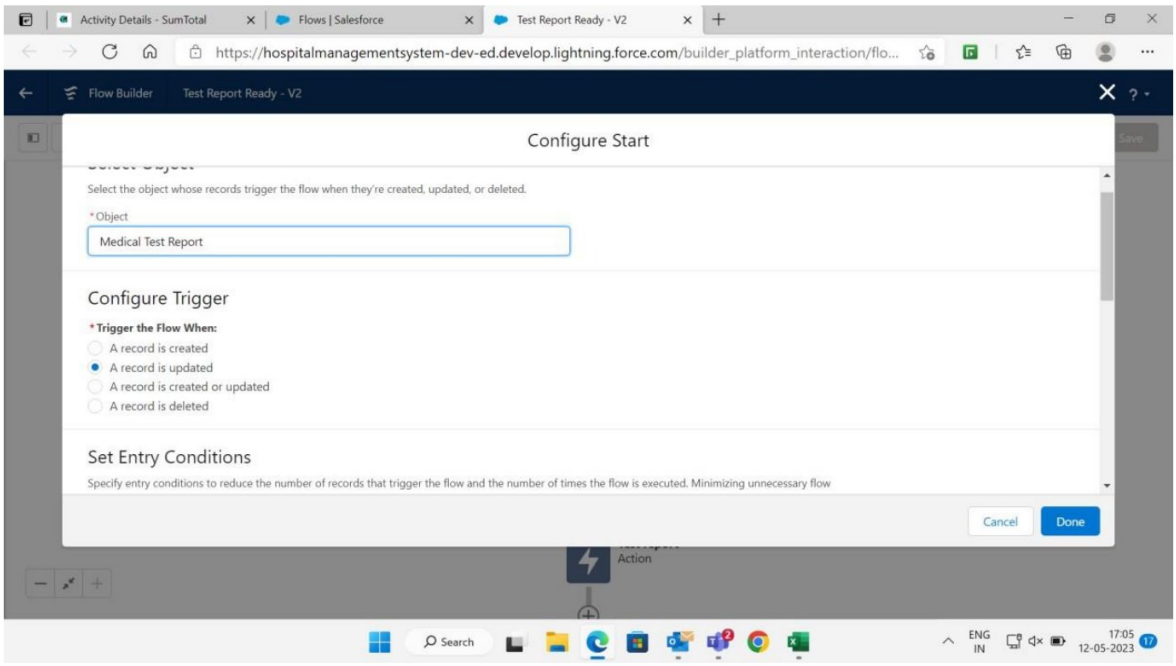
ABC hospital



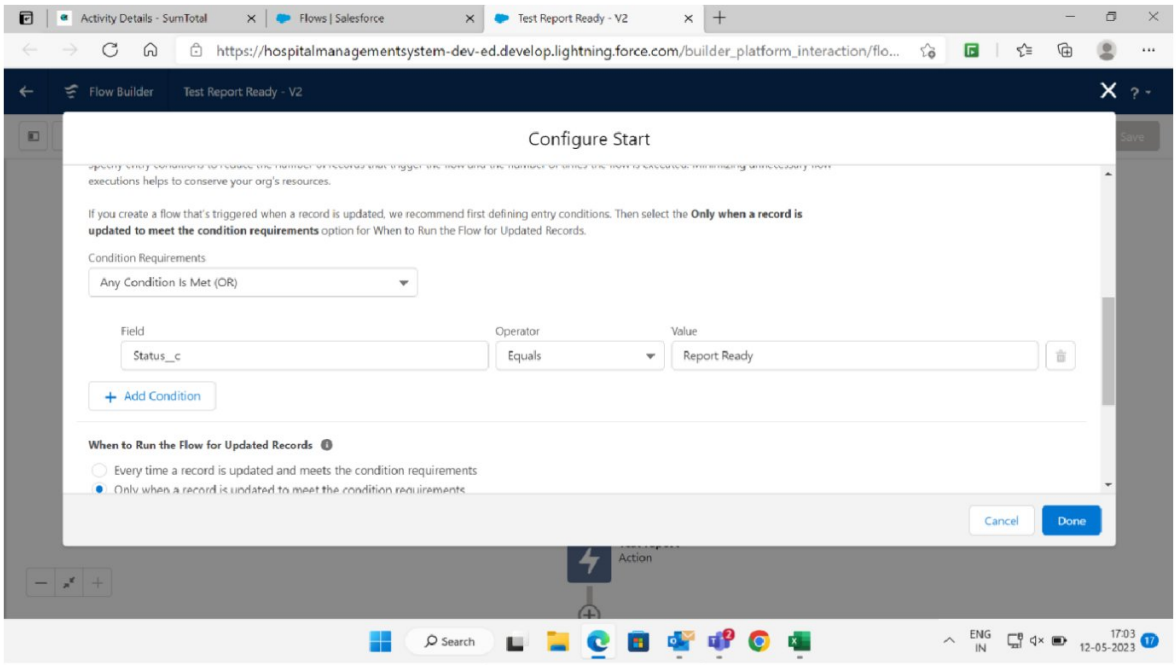
Scenario 14: TEST REPORT READY FLOW

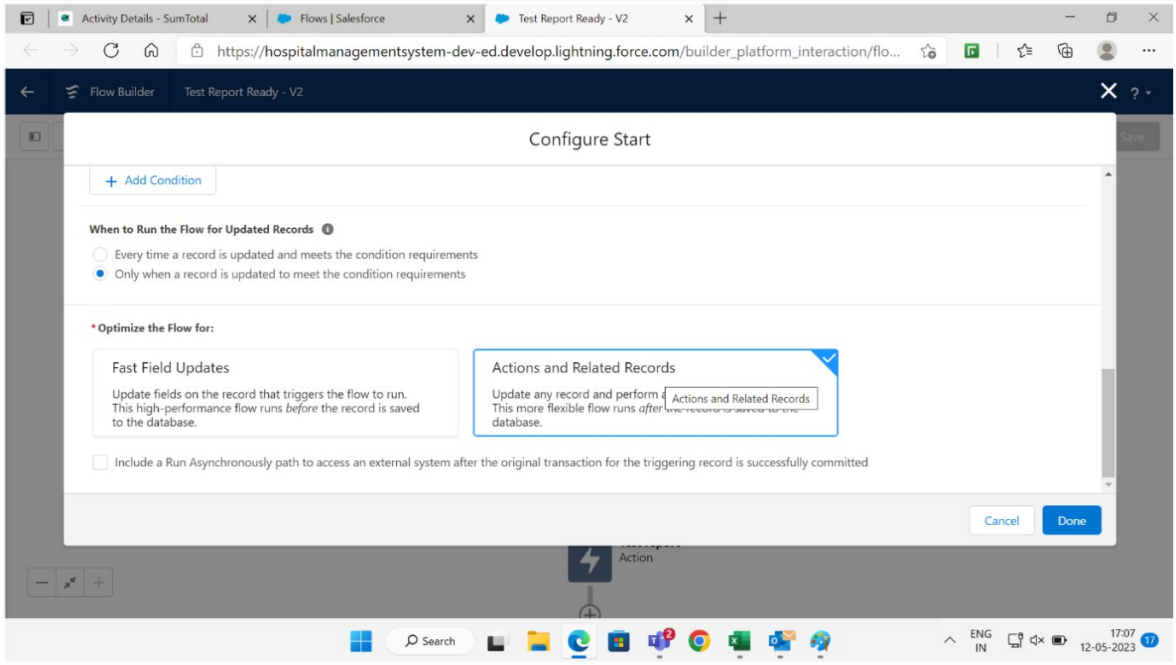
When the Lab staff upload the report file and update the status field in the medical test report object that trigger the email to patient

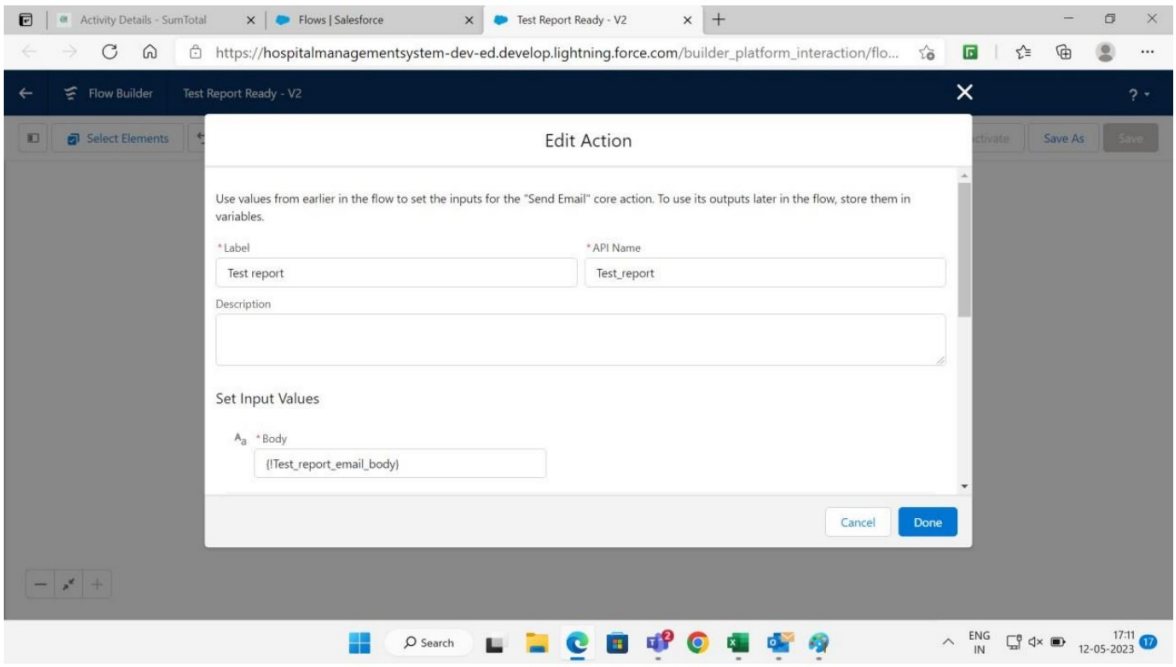
1. Create a new flow
2. Click record trigger flow
3. Object – Medical Test report
4. Trigger – A record is updated

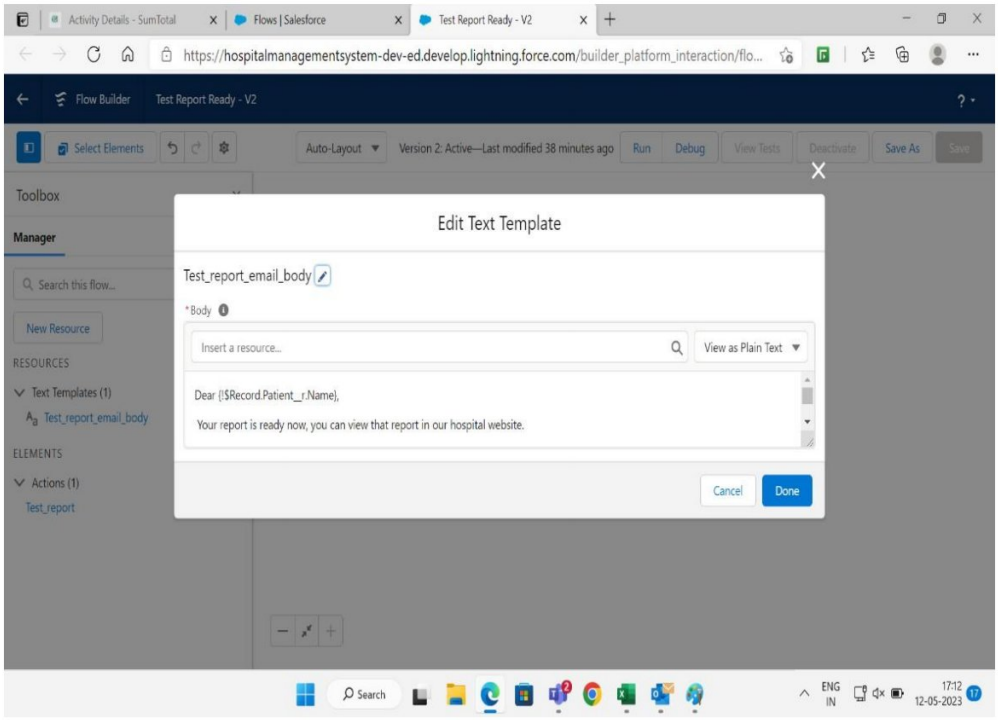


5.









Email body :

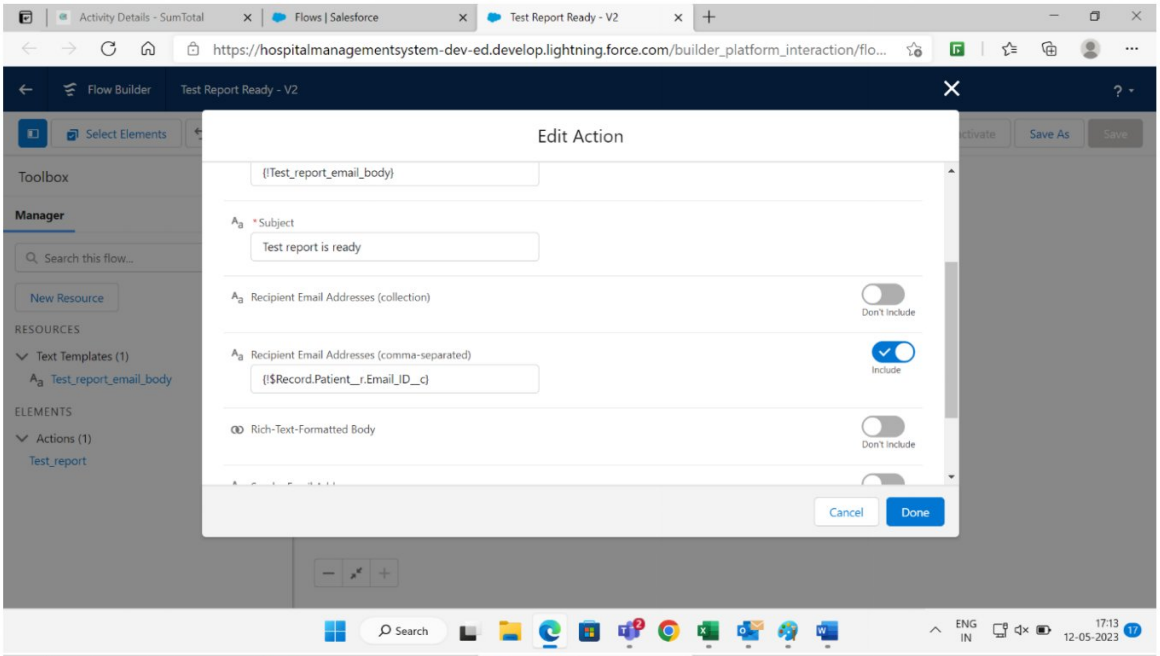
Dear {!$Record.Patient_r.Name},

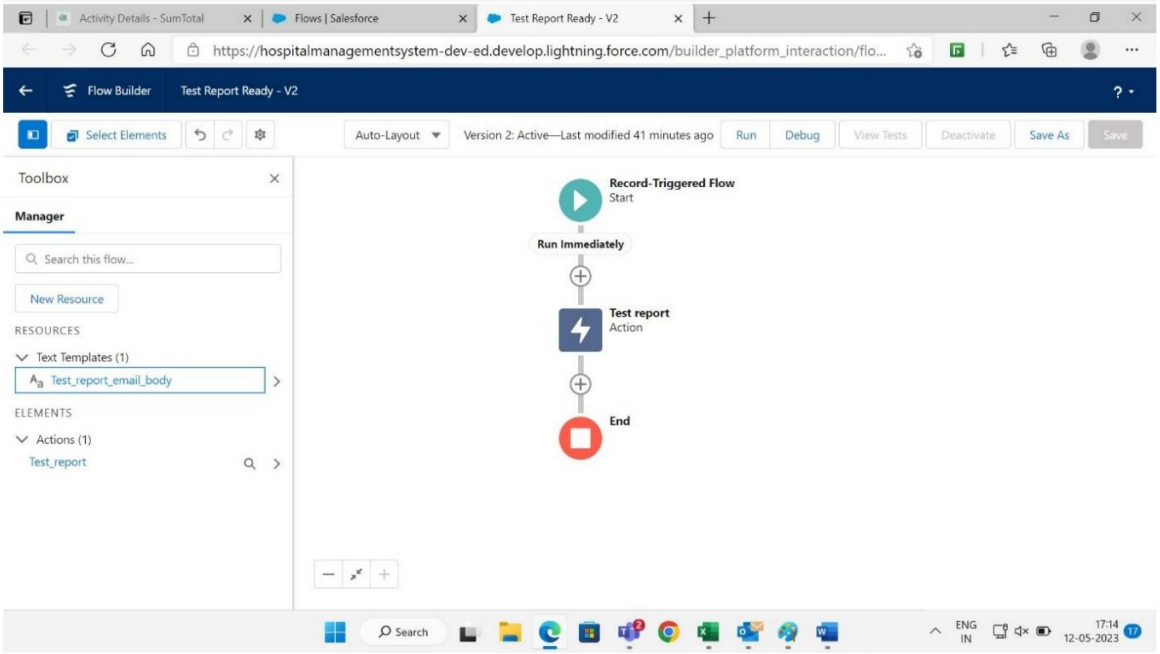
Your report is ready now, you can view

that report in our hospital website.

Report id : {!$Record.Name}

Thanks and Regards
ABC HospitaL





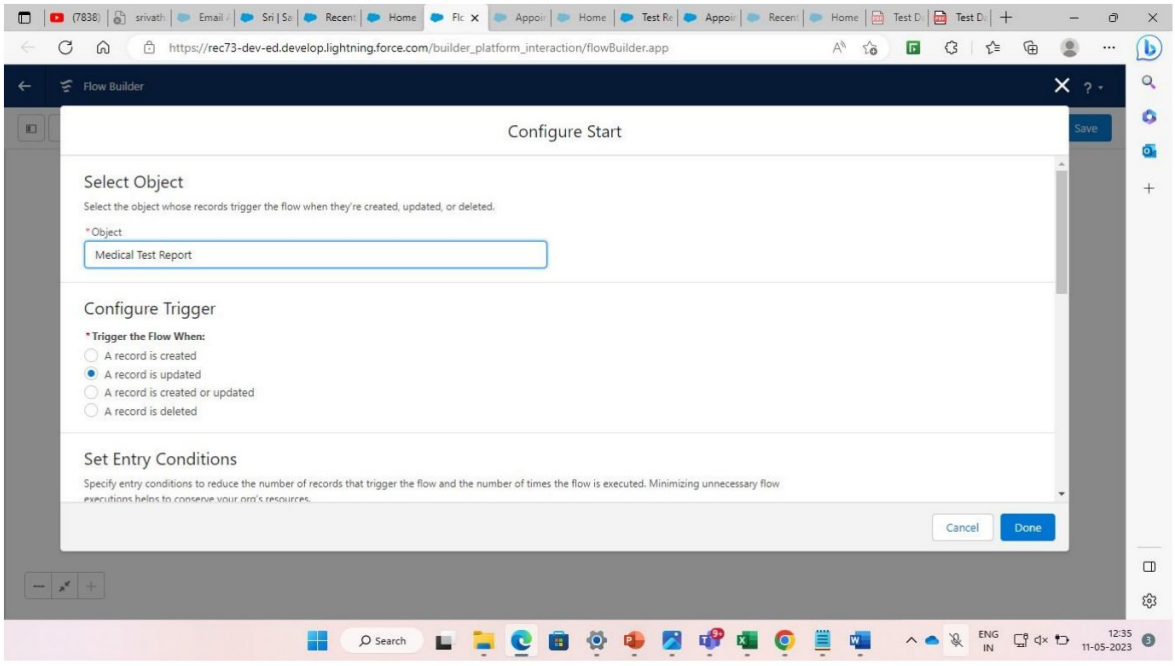
Scenario 15- TEST DONE FLOW

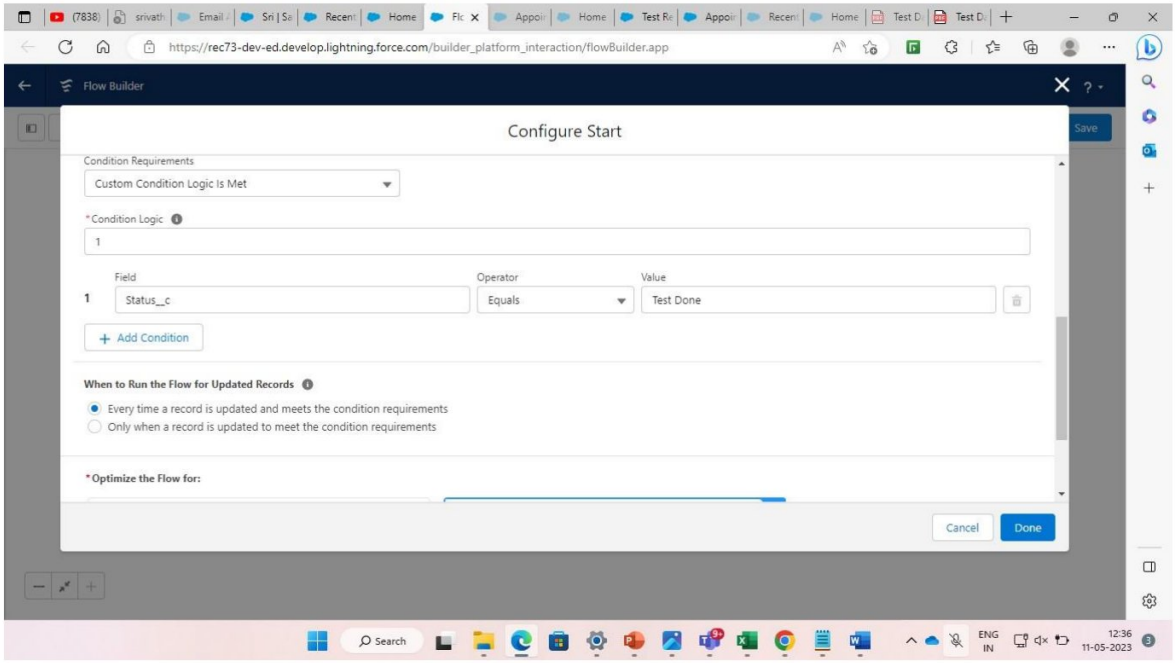
When medical test will be completed ,patient will receive a mail with test report

1. create a new flow

2. Click record trigger flow

- Object – Medical Test Report
- Trigger - A record is updated
- optimize for - Actions and Related Records



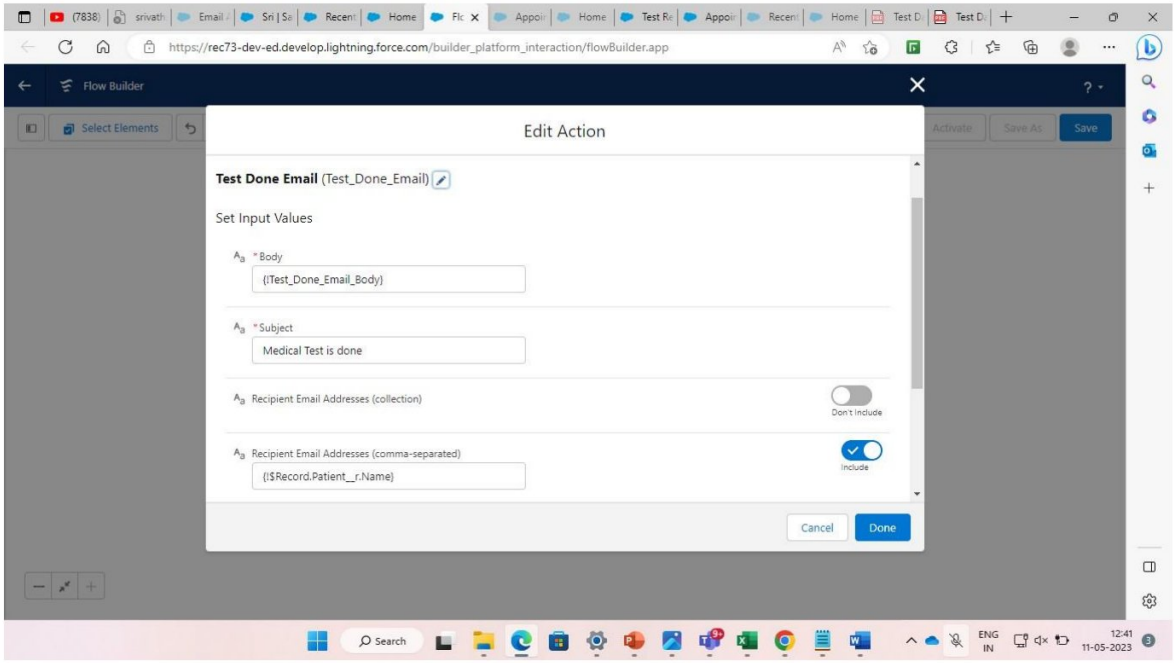


3. click the plus icon

4. Add action

5. In search all actions search for send email and choose it.

6. Label – Test Report Done.

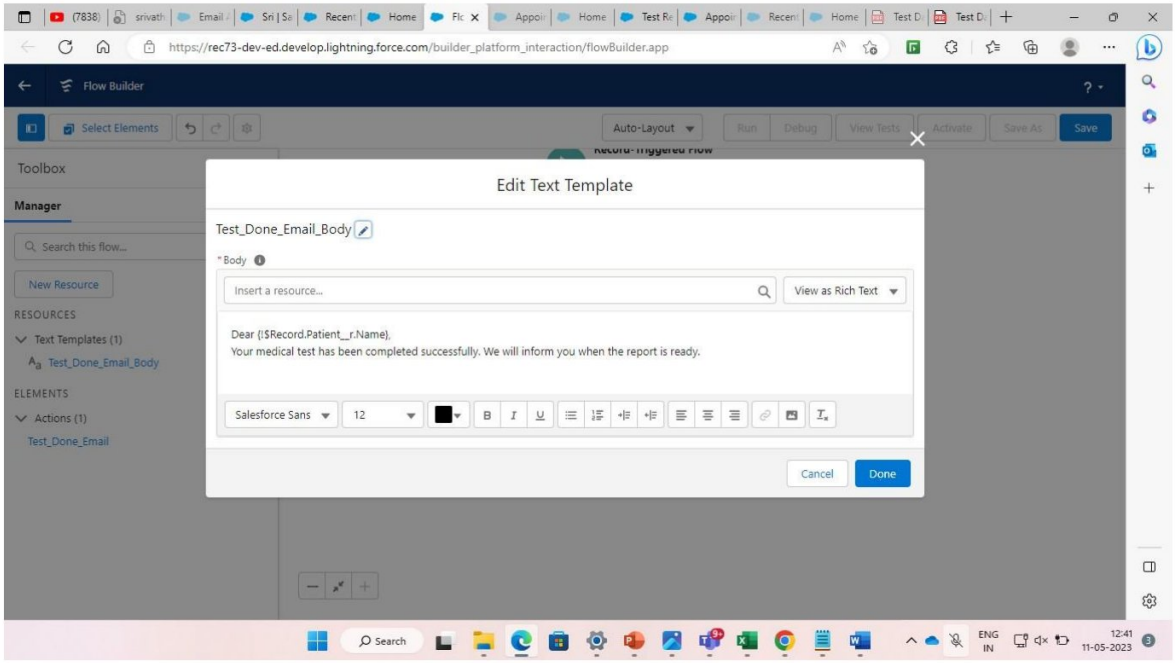


7. Body-New resource-Text Template-Test_Done_Email_Body(api name).

Body:

Dear {!$Record.Patient___r.Name},

Your medical test has been completed successfully.
We will inform you when the report is ready.

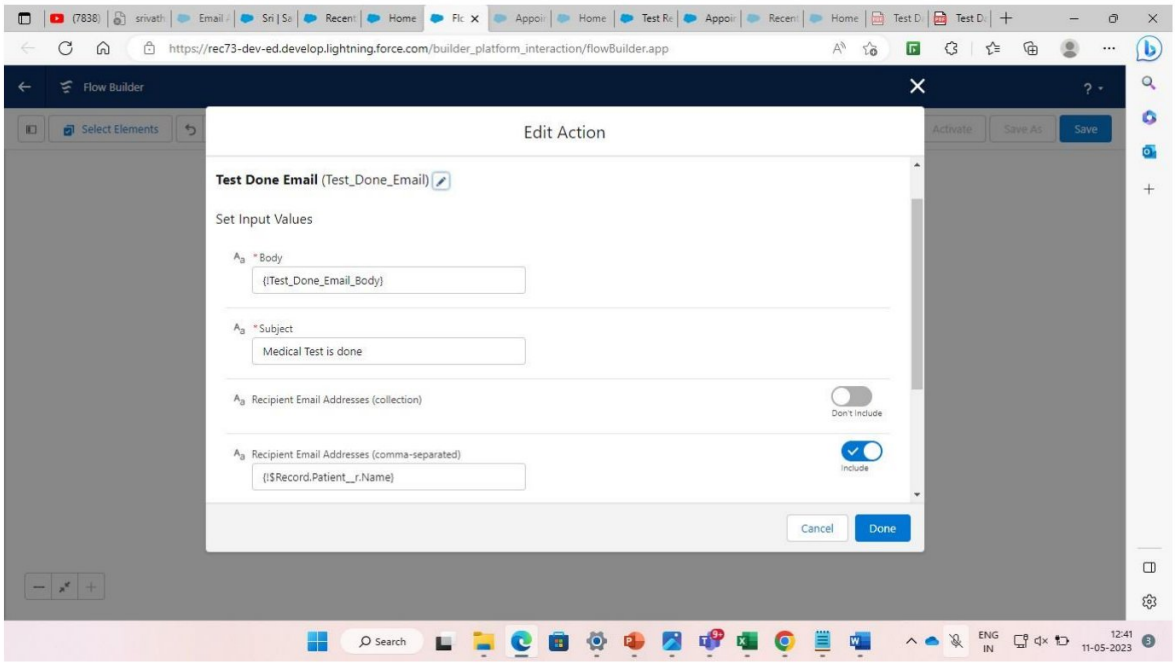


8. In subject type "Medical Test is done".

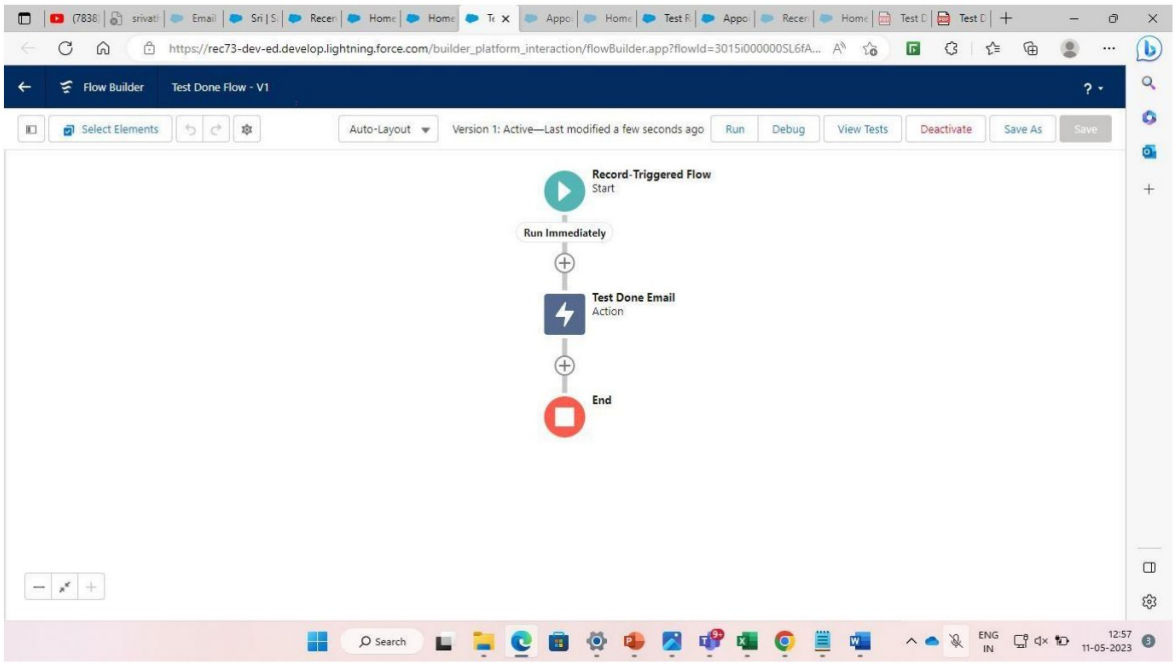
9. In recipient email address click include.

10. In recipient email addressses click Record-Medical Test Report-patient-email.

({!$Record.Patient_____r.Email____c}).



Final Flow :



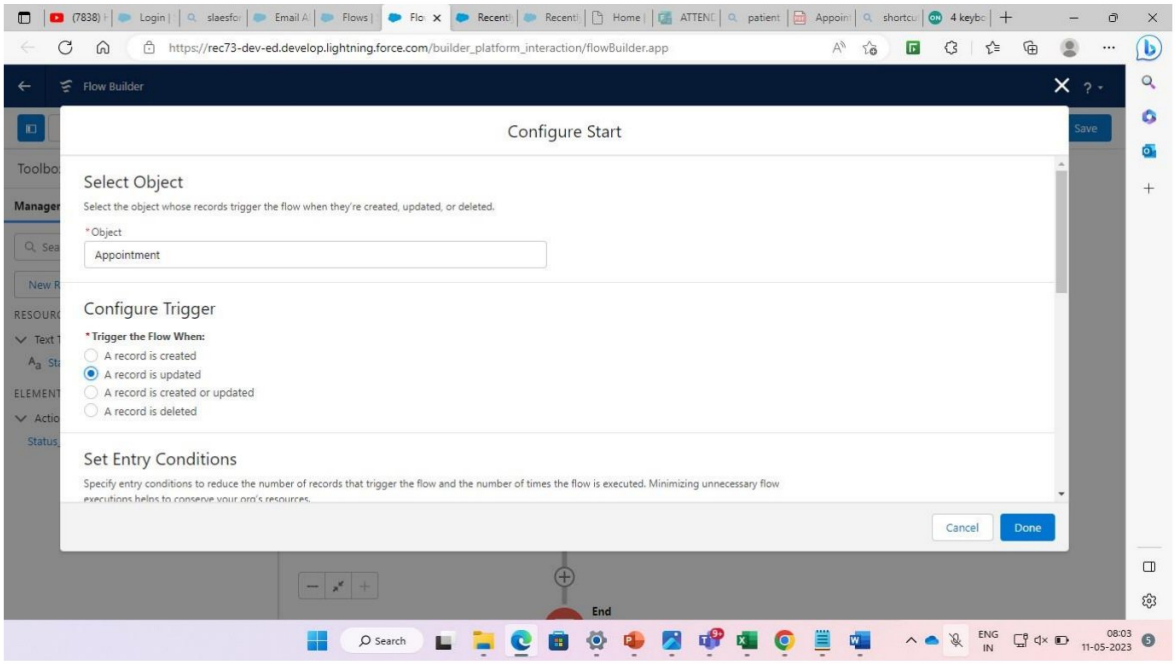
Scenario 16: APPOINTMENT STATUS FLOW

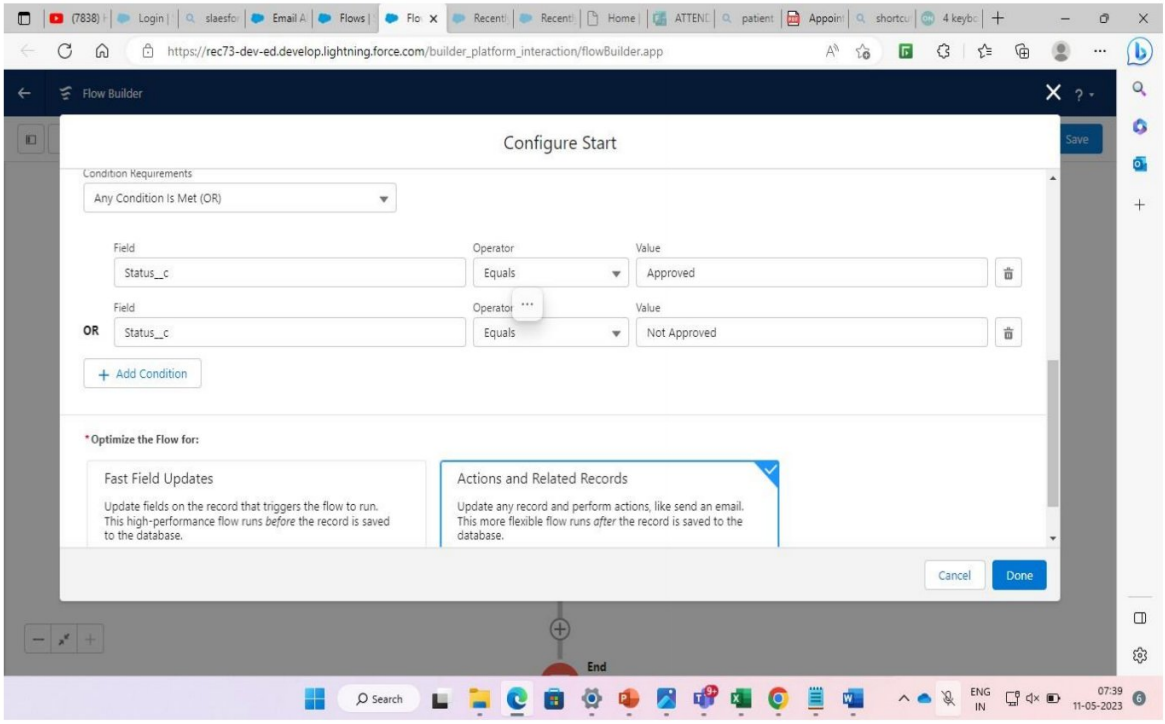
When doctor will approve or decline the appointment ,patient will get a mail accordingly.

1. create a new flow

2. Click record trigger flow

- Object – Appointment
- Trigger - A record is updated
- optimize for - Actions and Related Records



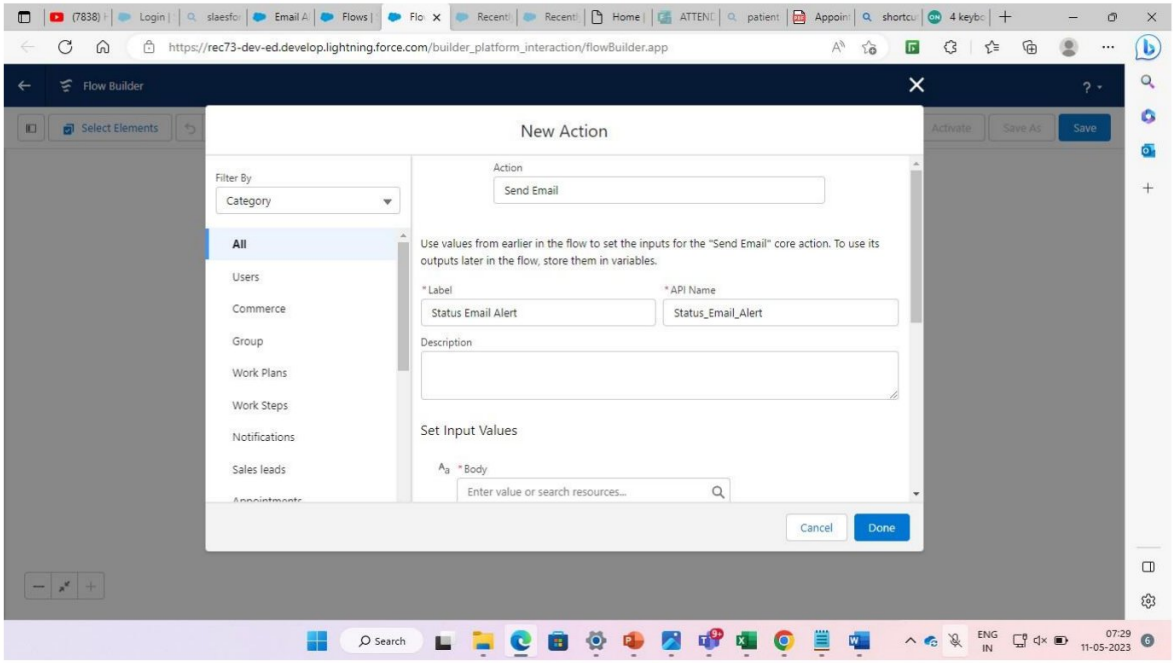


3. click the plus icon

4. Add action

5. In search all actions search for send email and choose it.

6 Label – Status Email Alert.



6. Body-New resource-Text Template-Status Email Body(api name).

<u>Body:</u>

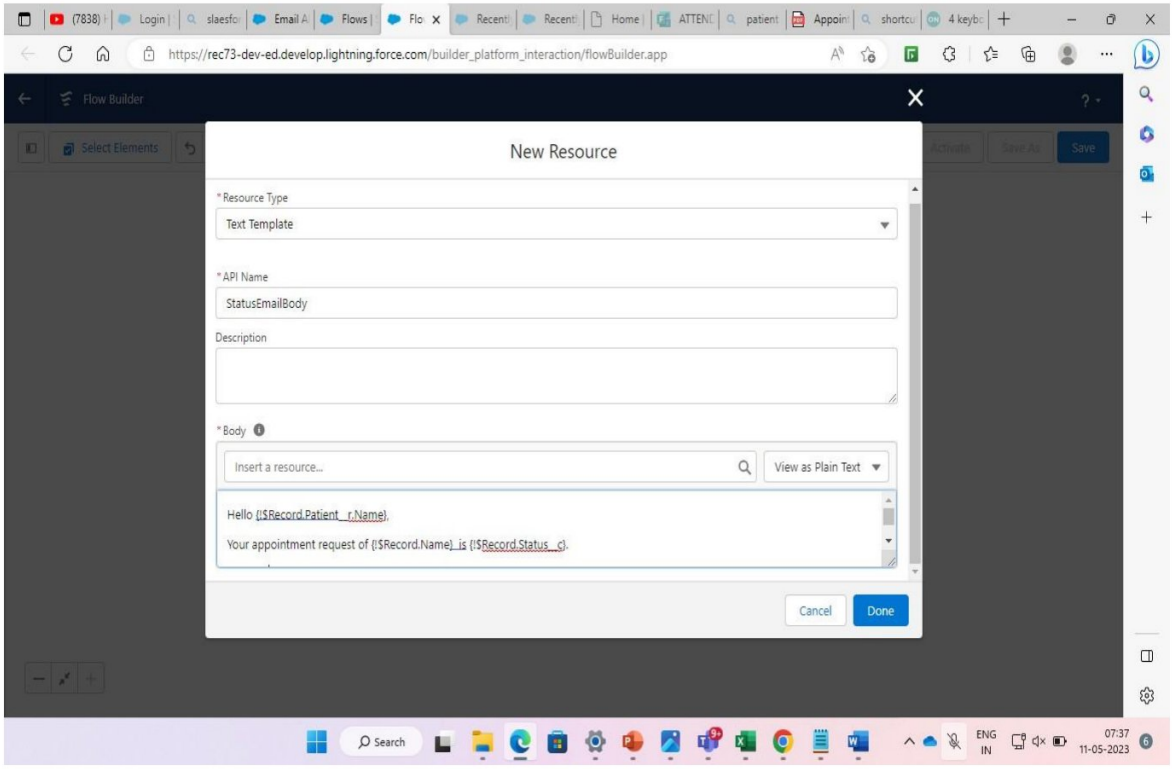
Hello {!$Record.Patient__r.Name},

Your appointment request of {!$Record.Name} is {!$Record.Status_c}.

Regards,

ABC

Hospital

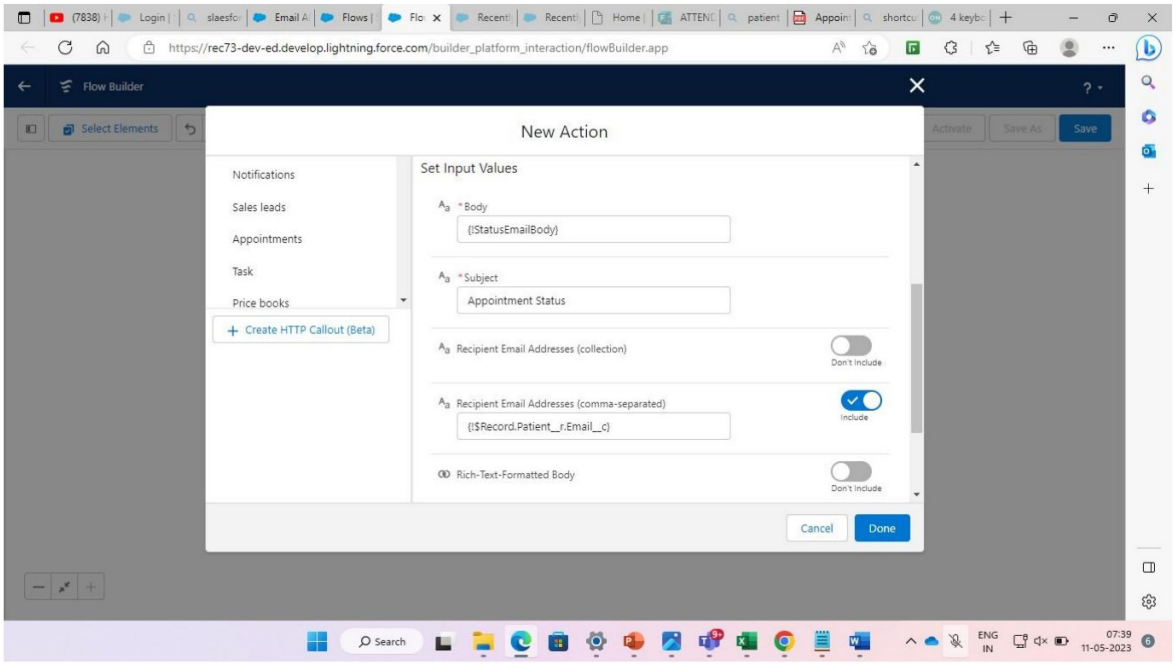


7. In subject type "Appointment Status".

8. In recipient email address click include.

9. In recipient email addressses click Record-Appointment-patient-email.

({!$Record.Patient_____r.Email____c}).



Final Flow :