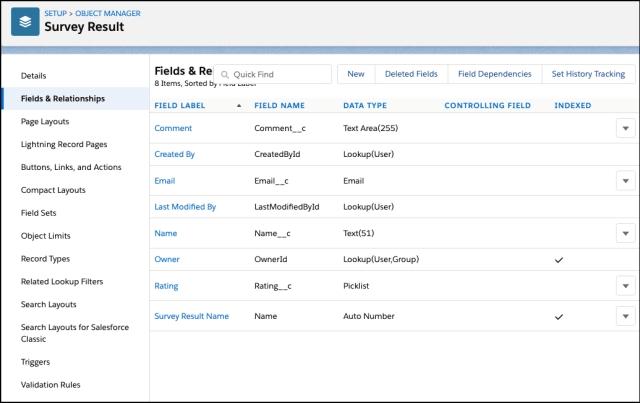
4.Create a screen flow for a basic survey to fill in the details for any form.

Solution:

Step 1: Create a Custom Object

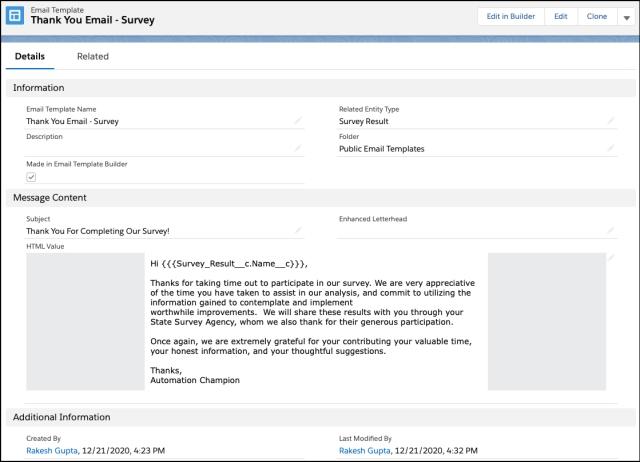
The next step is to create a custom object Survey Result and a few custom fields to store survey responses.

1. Click Setup.
2. In the Object Manager, click Create | Custom Object.
3. Now create a custom object Survey Result and fields as shown in the screenshot below:
4. Click Save.

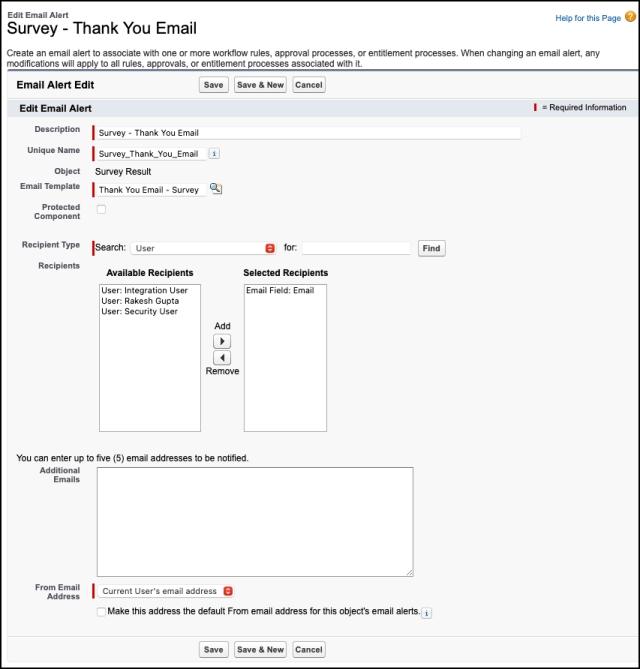


Step 2: Create a Thank You For Survey Lightning Email Template

1. Click App Launcher.
2. In the Quick Find box, type Email Templates.
3. Clicks on the New Email template button.
4. Name the Lightning Email Template and make sure to store it in the Public Email Templates folder.
5. Create a template like the following screenshot.

Step 3: Create an Email Alert

1. Click Setup.
2. In the Quick Find box, type Email Alerts.
3. Select Email Alerts, click on the New Email Alert button.
4. Name the Email Alert and click the Tab button. The Unique Name will populate.
5. For Object select Survey Result.
6. For the Email Template chooses Lightning Email Template Thank You Email – Survey.
7. For Recipient Type select Email Field: Email.
8. Click Save.



Step 4.1: Salesforce Flow — Create a Screen that Allow Users to Fill Survey

1. Click Setup.
2. In the Quick Find box, type Flows.
3. Select Flows then click on the New Flow.
4. Select the Screen Flow option and click on Next and configure the flow as follows:
5. **How do you want to start building: Freeform**
6. We will use the Screen element to capture a Survey response form. Drag and drop a Screen element onto the canvas.

Step 4.2: Salesforce Flow — Add a Record Creates Element to Save Survey Respons

1. Drag-and-drop the Create Records element onto the Flow designer.
2. Enter a name in the Label (Save Response) field; the API Name will auto-populate.
3. For How Many Records to Create – select One.
4. For How to Set the Record Fields – select Use separate resources, and literal values.
5. Select the Survey\_Result\_\_c object from the dropdown list.
6. Set Field Values for the Survey Result
7. Row 1:
8. Field: Comment\_\_c
9. Value: {!Comment}

2. Click Add Row

3. Row 2:

1. Field: Email\_\_c

2. Value: {!Email.value}

4. Click Add Row

5. Row 3:

1. Field: Name\_\_c

2. Value: {!Name.firstName} {!Name.lastName}

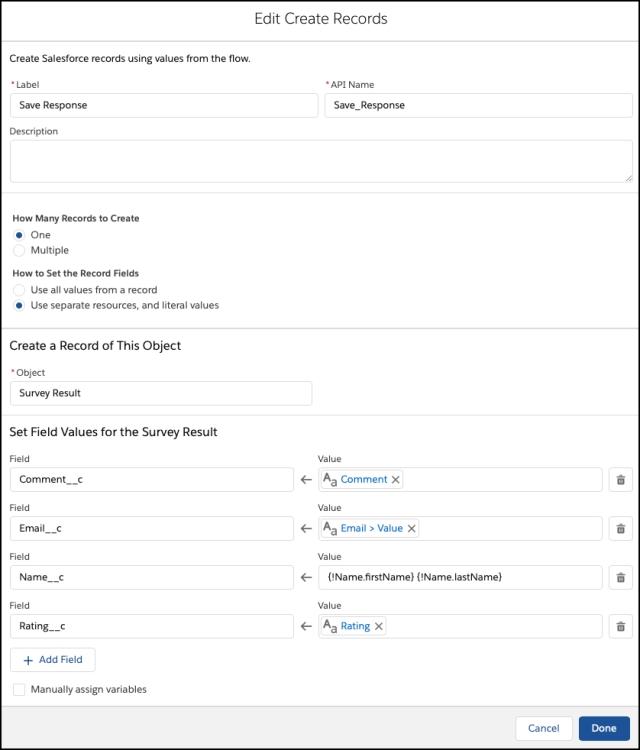
6. Click Add Row

7. Row 3:

1. Field: Rating\_\_c

2. Value: {!Rating}

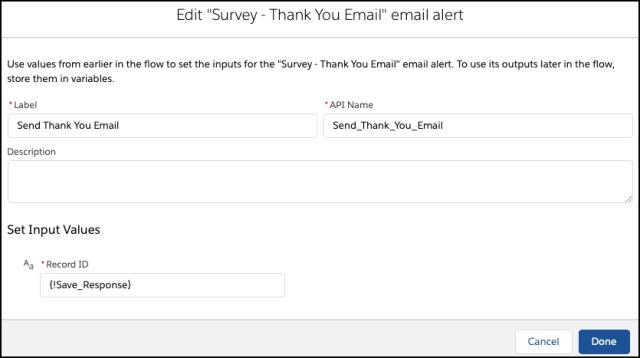
7. Click Done.

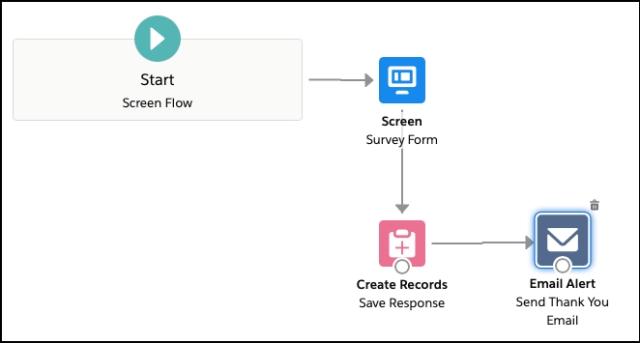


Step 4.3: Salesforce Flow — Call an Acton — Email Alert to Send Out Thank You Email

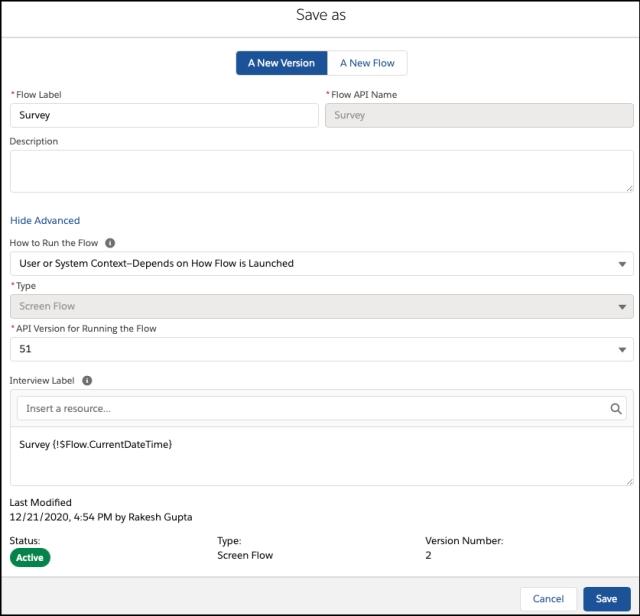
The next step is to call the Survey – Thank You Email email alert from flow so that when flow fires it triggers the thank you email to survey participants.

1. Under Toolbox, select Element.
2. Drag-and-drop Action element onto the Flow designer.
3. In the Action box, type Survey – Thank You Email.
4. Clicks on the Survey – Thank You Email email alert.
5. Click Done.



In the end, Sergio’s Flow will look like the following screenshot:

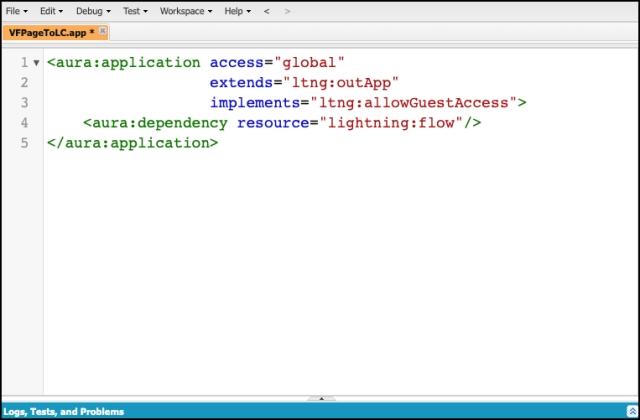
1. Click Save.
2. Enter Flow Label the API Name will auto-populate.
3. Click Show Advanced.
4. How to Run the Flow: User or System Context—Depends on How Flow is Launched
5. Type: Screen Flow
6. API Version for Running the Flow: 51
7. Interview Label: Survey {!$Flow.CurrentDateTime}
8. Click Save



Step 5: Create a Lightning Application to Render Lightning Runtime for Flow in a Visualforce Page

Now we will create a Lightning Application that declares a dependency on the lightning:flow component.

1. Click Setup | Developer Console
2. Navigate to File | New | Lightning Application
3. Enter a Name (VFPageToLC) field, make sure to select the Lightning Out Dependency App checkbox.
4. Click Submit.
5. Copy code from GitHub and paste it into your Lightning Application.
6. Save your code.

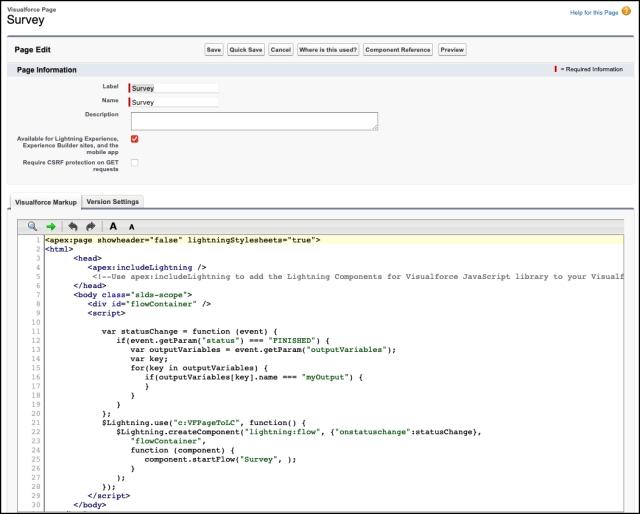


Step 6: Create a Visualforce Page and Embed Your Flow Into It

Now we will create a Lightning Application that declares a dependency on the lightning:flow component.

Add the Lightning Components for Visualforce JavaScript library to your Visualforce page using the <apex:includeLightning/> component. In the Visualforce page, reference the dependency app. Then write a JavaScript function that creates the component on the page using $Lightning.createComponent()Click Setup.

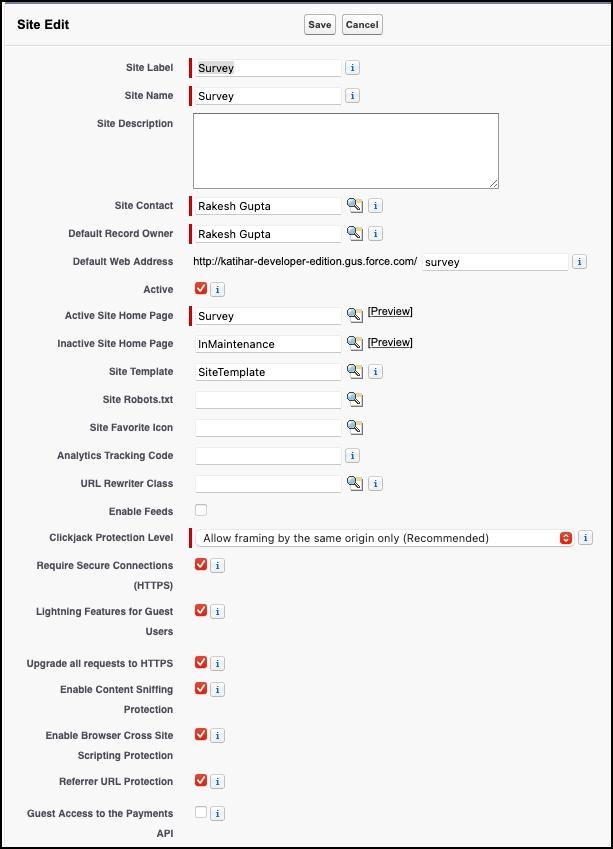
* 1. In the Quick Find box, type Visualforce Pages.
  2. Clicks on the New button.
  3. Copy code from GitHub and paste it into your visualforce page
  4. Click Save.



Step 7: Create a Force.com Site to Open Your Flow for Unauthenticated Access

Now we will create a site to open the flow for unauthenticated access.

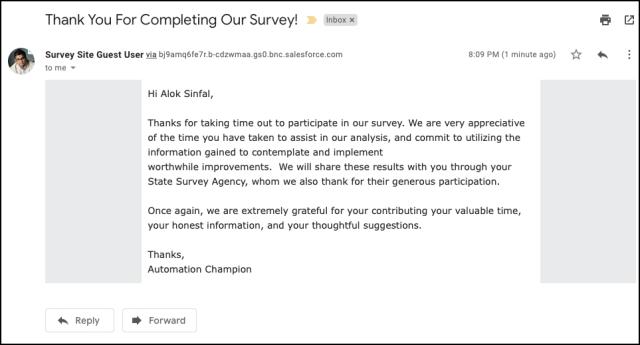
1. Click Setup.
2. In the Quick Find box, type Sites.
3. Clicks on the New button.
4. Fill the details as per the screenshot below:
5. Click Save.



Under site, Public Access Settings make sure that guest users have Create access on Survey Result object and Edit on the fields.

Proof of Concept

Now onward, if someone opens the site url and fills the form:





After successful submission, he/she will receive an email

