FlexiForm Installation Manual

Simploud

Introduction

FlexiForm is a versatile and intuitive Salesforce-native solution for creating, managing, and deploying dynamic forms tailored to your business needs. Designed to simplify complex data collection and workflow processes, FlexiForm enables users to create fully customizable forms that adapt to various use cases, including compliance audits, validation tests, customer feedback, and operational checklists.

Key Features:

- **Customizable Templates:** Build templates with flexible logic, conditional paths, and branching to suit diverse workflows.
- **Dynamic Data Integration:** Seamlessly connect forms to Salesforce objects to fetch, calculate, and update field values in real-time.
- **Audit Trail & Compliance:** Ensure transparency and compliance with comprehensive logging of all interactions, edits, and approvals.
- **User-Friendly Interface:** Empower users with an intuitive, guided form experience that streamlines data capture and processing.
- **Scalable & Adaptable:** Use FlexiForm across teams and departments for various applications, from simple surveys to advanced decision trees.

With FlexiForm, businesses can achieve greater efficiency, accuracy, and compliance in managing workflows, all while staying within the secure Salesforce ecosystem.

The following sections describe in details how to install FlexiForm and to use its advanced features on your Salesforce based application.

Simploud

Installation Steps

Follow the steps below to install and configure the FlexiForm application. Each step introduces an essential part of the setup process. Detailed instructions, including screenshots, are provided in subsequent sections.

1. **Create a Connected App for OAuth Authentication**
    - Name the app **"Electronic Signature"**.
    - Set the **OAuth Policy** to **Relax IP Restrictions** (enables access to a unique ID).
    - Match the **Callback URL** to the Salesforce login or test environment (https://login.salesforce.com or https://test.salesforce.com).
2. **Configure Remote Site Settings**
    - Add the **Org Domain URL** to Salesforce's Remote Site Settings. This ensures external calls from the application are authorized.
3. **Enable OAuth Flows**
    - In **OAuth and OpenID Connect Settings**, enable the **Allow OAuth Username-Password Flows** option. This is required for user authentication.
4. **Install the Application**
    - Obtain the installation link for FlexiForm and replace the placeholder prefix with your **Org Domain** URL to install in your specific Salesforce environment.
5. **Set Custom Settings and Update Signature Connected App Settings**
    - Navigate to **Custom Settings** and open **Signature Connected App**.
    - Enter the **Consumer Key** and **Consumer Secret** from the **Electronic Signature** Connected App created earlier.
6. **Assign the FlexiForm Admin Permission Set**
    - Grant the **FlexiForm Admin** permission set to all users who require access to the app. This permission set provides administrative privileges.
7. **Enable Profile Access to FlexiForm**
    - Ensure all relevant profiles have access to the **FlexiForm App**. This guarantees that the necessary users can interact with the application.
8. **Embed the FormList Lightning Component**
    - Add the **FormList** Lightning component to each Lightning Record Page where forms are required. This integrates FlexiForm into your pages.
9. **Configure Object Triggers for Auto Form Creation**
    - Add a **Trigger line** to each object where auto form creation is required. This step ensures forms are generated automatically as needed.

Simploud

Connected App

Access the Setup Menu

- Click on the **Gear Icon** in the top-right corner and select **Setup** from the dropdown.

Navigate to the Connected Apps Section

- In the **Quick Find** search bar, type **App Manager** and select **App Manager** from the search results.

Create a New Connected App

- In the **App Manager**, click the **New Connected App** button in the upper-right corner.

Basic Information Setup

- Fill in the required fields under the **Basic Information** section:
  - **Connected App Name**: Electronic Signature
  - **API Name**: Automatically populated as Electronic_Signature. You can leave this as is.
  - **Contact Email**: Enter your email address or the administrator's email.

Enable OAuth Settings

- Scroll down to the **API (Enable OAuth Settings)** section and check the box labeled **Enable OAuth Settings**.
- Provide the following details:
  - **Callback URL**: Enter https://login.salesforce.com (for production) or https://test.salesforce.com (for sandbox).
  - **Selected OAuth Scopes**: Add the following scopes by moving them from the **Available OAuth Scopes** list to the **Selected OAuth Scopes** list:
    - Allow access to your unique identifier (openid)

Set IP Relaxation

- Under the **Policies** section you can access if you click on the **Manage** button:
  - For **IP Relaxation**, select **Relax IP Restrictions**. This allows access to unique identifiers without being blocked by IP constraints.

Save the Connected App

- After filling out the details, click **Save**.
- You will see a message indicating that the Connected App might take up to 10 minutes to become active.

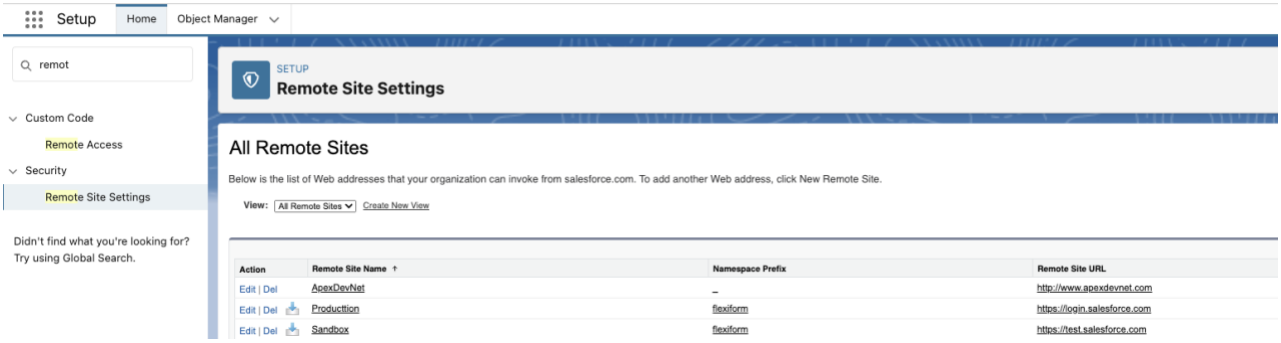
Retrieve Consumer Key and Secret

- Once the app is saved:
  - In the **Manage Connected Apps** section, locate the **Electronic Signature** app and click on its name.
  - Under the **API (Enable OAuth Settings)** section, copy the **Consumer Key** and **Consumer Secret**. These will be required later for the **FlexiForm installation**.

Remote Site Settings

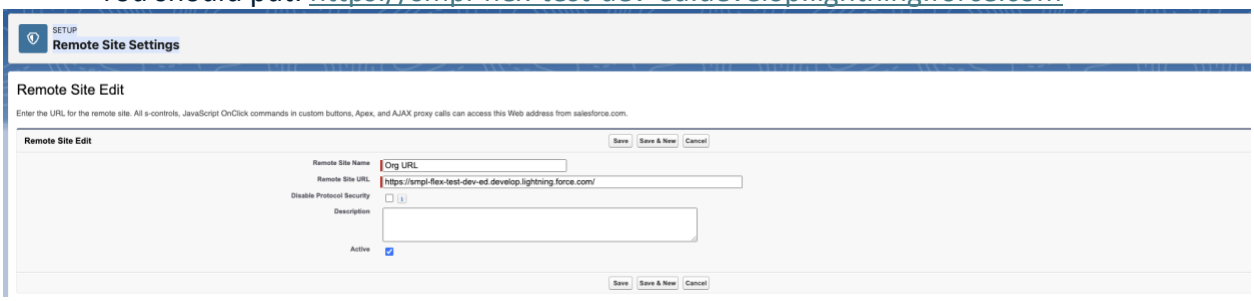
Navigate to the Connected Apps Section

- In the **Quick Find** search bar, type **Remote** and select **Remote Site Settings** from the search results.



Create a new Remote Site Settings

- Give the remote site settings a name like Org URL
- For the remote site URL copy and paste the URL of your domain until the first "/" sign
- For example if your URL on the browser is: https://smpl-flex-test-dev-ed.develop.lightning.force.com/lightning/page/home
  You should put: https://smpl-flex-test-dev-ed.develop.lightning.force.com



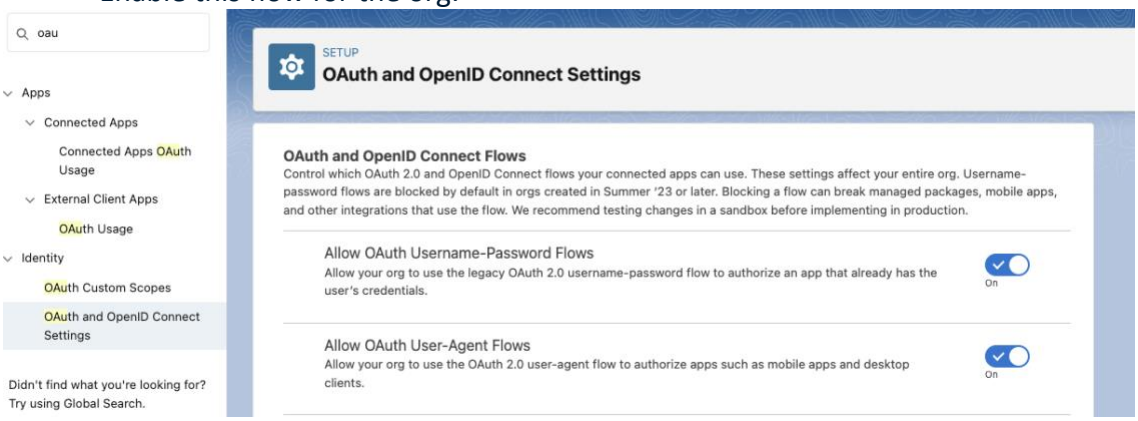
Enable Oauth Flows

Access the OAuth Settings

- In the **Quick Find** search bar, type **OAuth** and click **OAuth and OpenID Connect Settings**.

Locate and Edit the Setting

- In the **OAuth and OpenID Connect Settings**, look for the option **Allow OAuth Username-Password Flows**.
- Enable this flow for the org.



Application Installation

Obtain the Installation Link

- Locate the FlexiForm application installation link provided to you.
- Ensure the link corresponds to the correct Salesforce environment:
  - **Production**: https://login.salesforce.com
  - **Sandbox**: https://test.salesforce.com

Modify the Installation Link

- Replace the placeholder prefix in the installation link with your **Org Domain URL**.
  - Example: Replace https://your-domain.my.salesforce.com with your Salesforce org's unique domain.

Open the Installation Link

- Paste the modified link into your browser and press **Enter** to start the installation process.

Login to Salesforce

- If prompted, log in to your Salesforce org with an administrator account.

Choose the Installation Type

- You will be asked to select the installation scope. Choose **Install for All Users**

Acknowledge the Security Warnings

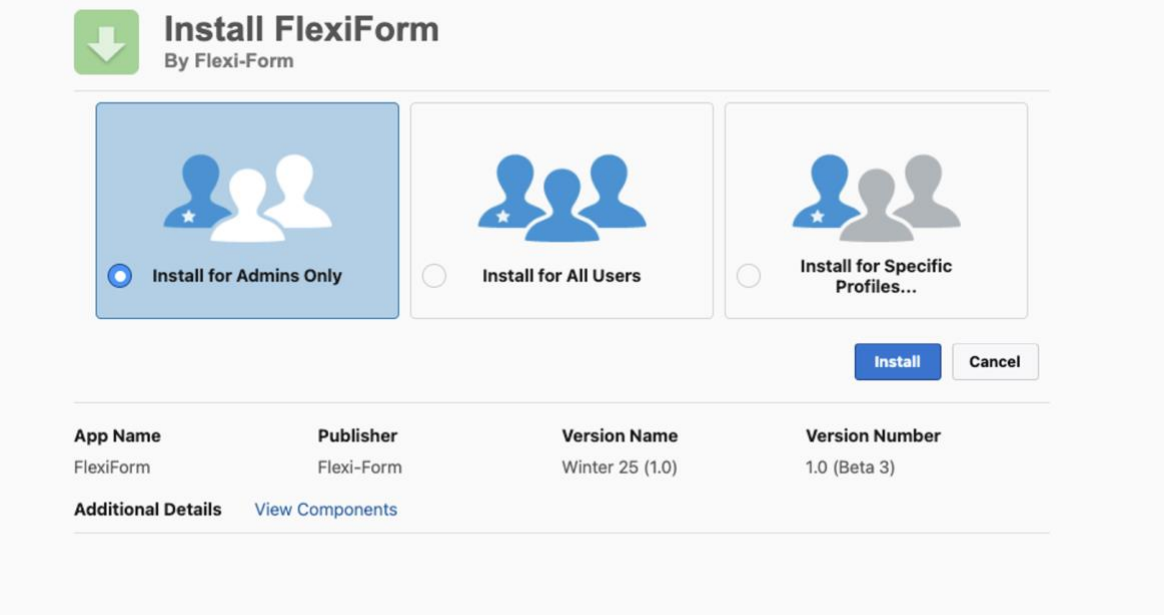
- Review the security warnings presented during the installation.
- Check the box to confirm that you acknowledge the warnings and trust the package source.

Start the Installation

- Click the **Install** button to begin. The installation process may take a few minutes.

Wait for Installation Completion

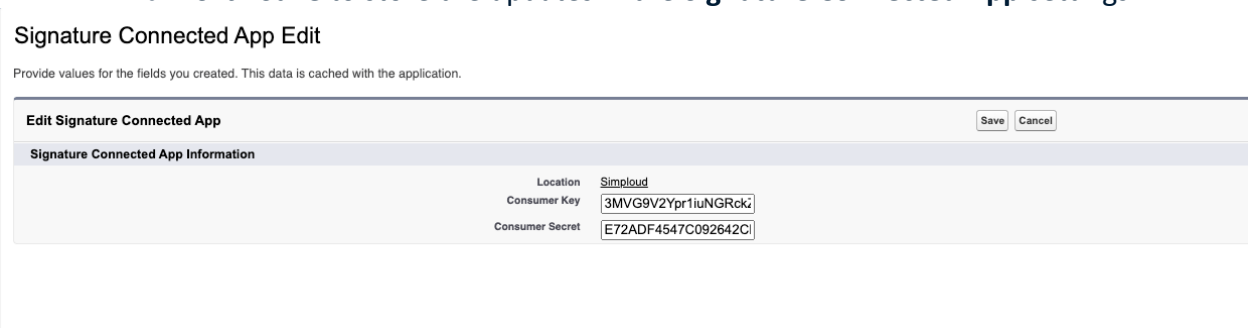
- Once the installation is complete, you'll see a confirmation message. If the process takes longer, you will be notified via email when it's done.



Custom Settings

1. **Locate the Custom Settings**
    - In the **Quick Find** search bar, type **Custom Settings** and select **Custom Settings** from the search results.
2. **Find FlexiForm Settings**
    - In the list of available custom settings, locate **FlexiForm Settings** and click **Manage** next to it.
3. **Create a New Top-Level Record**
    - Click the **New** button to create a new top-level record.
    - Click **Save** to finalize the record creation.

1. **Find Signature Connected App in Custom Settings**
    - In the **Custom Settings** list, locate **Signature Connected App** and click **Manage** next to it.
2. **Create a New Top-Level Record**
    - Click the **New** button to create a new top-level record for **Signature Connected App**.
    - Click **Save** to create the record.
    - **Edit the Signature Connected App Record**
    - After creating the top-level record for **Signature Connected App**, click the **Edit** link next to the record you just created.
    - Retrieve the **Consumer Key** and **Consumer Secret** from the connected app configuration for **Electronic Signature** (as created earlier).
    - Copy the **Consumer Key** and **Consumer Secret** from the connected app and paste them into their respective fields in the **Signature Connected App** record.
2. **Save the Record**
    - Click **Save** to store the updates in the **Signature Connected App** settings.



Permission Set Assignments and Profile Access

Search for Permission Sets

- In the **Quick Find** search bar, type **Permission Sets** and select **Permission Sets** from the results.

Locate "FlexiForm Admin"

- In the list of available permission sets, confirm the presence of the **FlexiForm Admin** permission set.

Open the Permission Set

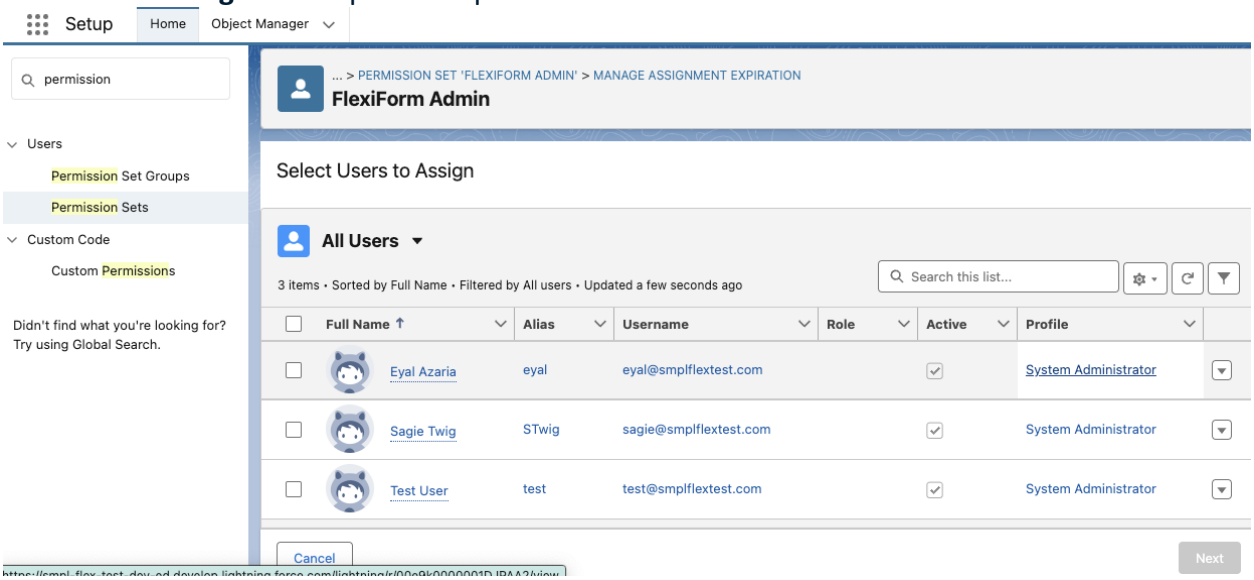
- In the **Permission Sets** list, click on **FlexiForm Admin**.

Assign Users

- On the **Permission Set Detail** page, click the **Manage Assignments** button.

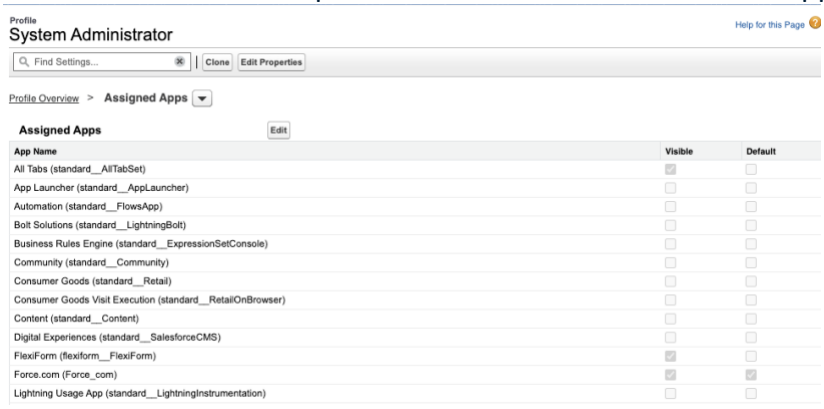
Add Users

- Click **Add Assignments**.
- Search for and select the users who require access to the app.
- Click **Assign** to complete the process.



Profile access to FlexiForm app

- In the **Quick Find** search bar, type **Profile** and select **Profiles** from the results.
- For the relevant profiles add access to the FlexiForm app



Embed FormList Component

Navigate to the Object Manager

- In **Setup**, type **Object Manager** in the **Quick Find** search bar and click **Object Manager**.

Select the Relevant Object

- From the list of objects, find and click the name of the object where you want to embed the **FormList** component (e.g., Account, Opportunity, or a custom object).

Open the Lightning Record Page

- On the object's detail page, click **Lightning Record Pages** in the left-hand menu.
- Select the Lightning Record Page you want to edit. If no page exists, create a new one by clicking **New**.

Edit the Lightning Record Page

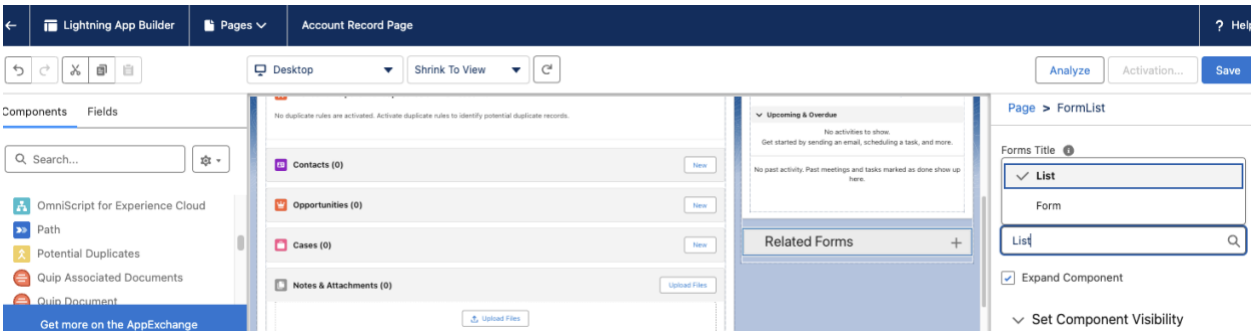
- Click **Edit** next to the Lightning Record Page you intend to modify.
- This will open the **Lightning App Builder** interface.

Add the FormList Component

- In the Lightning App Builder, locate the **Components** panel on the left side.
- Search for the **FormList** component.
- Drag and drop the **FormList** component onto the desired section of the page layout.

Configure the Component (if necessary)

- Select the **FormList** component on the page layout.
- Configure its properties in the right-hand **Component Properties** panel
- The FormList can appear as a list by default or as the full single form view
- Click **Save** to save the changes to the Lightning Record Page.
- If the page has not been activated yet, click **Activation** in the top-right corner.
- Assign it as the **Org Default** for all profiles.



Simploud

Object Triggers

The following detailed instructions will guide you through adding the provided trigger snippet to an existing trigger. This assumes the organization already has a trigger on the target object. The instructions also cover using the namespace prefix for external access to TemplateTriggerHandler.autoCreateForms.

Part 1: Understand the Trigger Requirements

- **Trigger Scope**: The snippet is designed for the **after update** scope.
- **Namespace Prefix**: The namespace prefix (e.g., flexiform) is required to access TemplateTriggerHandler.autoCreateForms from outside the FlexiForm package.
- **Global Method**: The method TemplateTriggerHandler.autoCreateForms requires:
  - Trigger.new: A list of updated records.
  - Trigger.oldMap: A map of old records with their IDs to track changes.

Part 2: Adding the Snippet to an Existing Trigger

1. **Locate the Relevant Trigger**
   - Navigate to **Setup** > **Object Manager** and select the object where the trigger exists (e.g., Account, Opportunity, or a custom object).
   - Click **Triggers** in the left-hand menu to view existing triggers.
   - Locate the trigger handling the **after update** event.
2. **Edit the Trigger**
   - Click **Edit** next to the trigger to modify its code.
   - Review the existing logic to determine the best placement for the new snippet.
3. **Add the Trigger Snippet**
   - Insert the following line of code within the **after update** scope:

```
trigger AccountCustomTrigger on Account (after update) {
   // Existing logic for other operations

   // Add the snippet for after update
   if (Trigger.isAfter && Trigger.isUpdate) {
      flexiform.TemplateTriggerHandler.autoCreateForms(Trigger.new, Trigger.oldMap);
   }
}
```

4. **Save the Trigger**
   - Save your changes. Salesforce will compile the trigger to check for syntax errors.

Part 3: Creating a New Trigger (If None Exists)

If no trigger exists for the object, create one for the after update event:

1. **Navigate to Apex Triggers**
   - In the **Object Manager**, select the target object.
   - Click **Triggers** > **New**.
2. **Write the Trigger**
   - Use the following code as a template:

```
trigger ObjectCustomTrigger on ObjectName (after update) {
    flexiform.TemplateTriggerHandler.autoCreateForms(Trigger.new, Trigger.oldMap);
}
```

- Replace ObjectName with the API name of your target object (e.g., Account, Opportunity).

3. **Save the Trigger**
   - Click **Save** to create and compile the trigger.