



**CODEYLABS**

## **DRTE - ADD ON**

Automatic Save Final Signed  
Document With Certificate Of  
Completion on document  
record

**User Guide**



DRTE

CoC feature is an add-on designed to ensure legal compliance and provide irrefutable proof of execution for every document signed using DRTE.

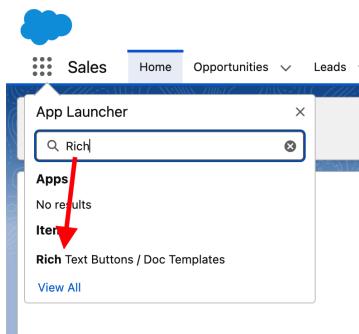
Once all parties have successfully signed a document, the CoC feature allows you to combine the final, executed document with a detailed, legally-compliant **Audit Log** into a single, secure PDF file.

# Configuration - Certificate of Completion

Following installation, the module requires specific configuration within DRTE and Salesforce Setup.

## Step 1: Create the Certificate of Completion Document Template

1. Navigate to the Rich Text Button/Document Templates Tab.



2. Create a **New** Document Template record and name it "**Certificate of Completion.**"

A screenshot of the 'New Rich Text Button / Doc Template' creation page. The page has a header 'New Rich Text Button / Doc Template' and a note '\*= Required Information'. The main section is titled 'Information'. It contains the following fields:

- Rich Text Button Number (dropdown menu)
- Template Name: **Certificate of Completion** (highlighted in yellow)
- SLDS icon-name: **SLDS icon-name** (highlighted in yellow)
- In Use: **In Use** (highlighted in yellow)
- Owner: **Rahul Jain**
- Category: **Category** (highlighted in yellow)
- Deactivated: **Deactivated** (highlighted in yellow) with a checked checkbox
- Display Order: **0** (highlighted in yellow)
- Test SFDC Record Id: **Test SFDC Record Id** (highlighted in yellow)

At the bottom, there is a 'System Information' section with 'Cancel', 'Save & New', and 'Save' buttons.

3. Save the record.

#### 4. Import the JSON Template:

- Open **Setup** in a new tab, search for **Static Resources**, and locate the **CertificateOfCompletionDocumentSectionsJSON** JSON file (this file is bundled with the managed package). **Download this file.**

- Return to the newly created Document Template record.
- Utilize the **Import** functionality to upload the downloaded JSON file.

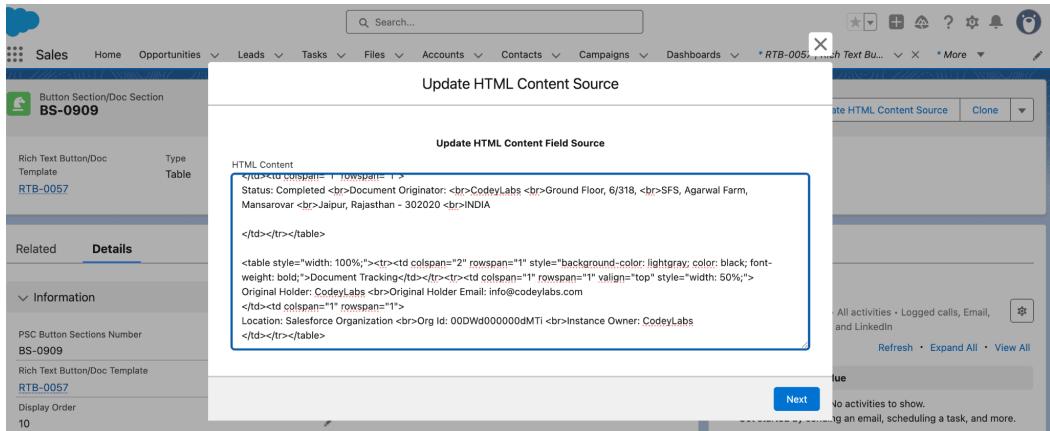
The screenshot shows the 'Rich Text Button / Doc Template' page for 'RTB-0060'. The main area displays the 'Details' tab with fields like 'Rich Text Button Number' (RTB-0060), 'Owner' (Rahul Jain), 'Category' (1), 'Deactivated' (unchecked), and 'Display Order' (0). To the right, a sidebar titled 'Document Section Export/Import' has two buttons: 'Export Sections' and 'Import Sections' (highlighted with a red arrow). Below these buttons is a message 'Current: 0 document sections'.

5. Upon successful import, refresh the page. The pre-defined document sections will now be visible.

The screenshot shows the 'Rich Text Button / Doc Template' page for 'RTB-0060'. The main area displays the 'Details' tab with a table titled 'Button Sections/Doc Sections (6+)'. The table lists six sections with columns for 'PSC Button Sections Number', 'Display Order', 'HTML Content', and 'Is Active'. To the right, the 'Document Section Export/Import' sidebar shows 'Current: 8 document sections' and an 'Activity' section with a message 'Filters: Within 2 months - All activities - Logged calls, Email, Events, List email, Tasks, and LinkedIn'.

6. **Customize Placeholders:** Review each section and modify the static text placeholders as necessary to meet organizational branding and requirements.

The screenshot shows the 'Rich Text Button / Doc Template' page for 'RTB-0060'. The main area displays the 'HTML' tab, which contains a 'Certificate of Completion' section with static text placeholders such as 'Document Id: {{date\_ext\_GENERIC\_DOC\_ID\_c}}', 'Status: Completed', and 'Document Originator: [YOUR COMPANY ADDRESS]'. A red arrow points to the placeholder 'Document Id: {{date\_ext\_GENERIC\_DOC\_ID\_c}}'. To the right, there is a large blue placeholder area.



## Step 2: Configure DRTE Allowed Objects

This step ensures the DRTE query engine can access the relevant e-signature log data for certificate generation.

1. In **Setup**, search for **Custom Metadata Types**.
2. Locate and manage the records for the **DRTE Allowed Object** custom metadata type.

Action	Label	Installed Package	Namespace Prefix	Visibility	API Name	Record Size	Description
Manage Records	DRTE Allowed Object	DRTE	pscdnryrichtext	Public	pscdnryrichtext__DRTE_Allowed_Object__mdt	396	DRTE Allowed Object contains the list of the name of standard and custom object that DRTE component will allow to query. If there's a query defined in the document sections, please ensure that the object queried is listed here

3. Create two new records to include the following objects:
  - `pscdnryrichtext__ESignature_Log__c`
  - `pscdnryrichtext__ESignature_Request__c`

DRTE Allowed Object

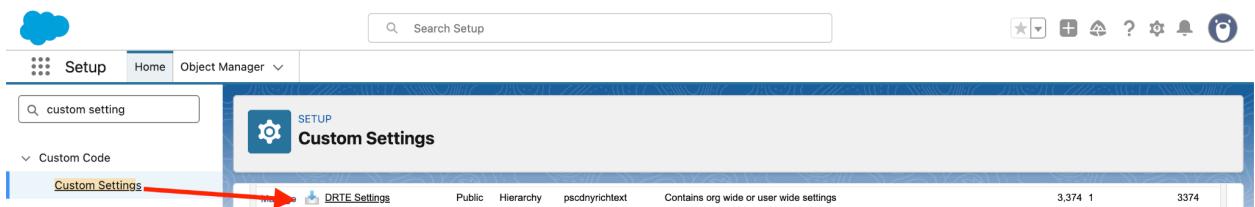
DRTE Allowed Object Detail		<a href="#">Edit</a>	<a href="#">Delete</a>	<a href="#">Clone</a>
Label	pscdnyrichtext__E_Signature_Request__c	Protected Component <input type="checkbox"/>		
DRTE Allowed Object Name	pscdnyrichtext_E_Signature_Request_c	Namespace Prefix		
Object API Name	pscdnyrichtext__E_Signature_Request__c			

DRTE Allowed Object

DRTE Allowed Object Detail		<a href="#">Edit</a>	<a href="#">Delete</a>	<a href="#">Clone</a>
Label	pscdnyrichtext__ESignature_Log_c	Protected Component <input type="checkbox"/>		
DRTE Allowed Object Name	pscdnyrichtext_ESignature_Log_c	Namespace Prefix		
Object API Name	pscdnyrichtext__ESignature_Log__c			

### Step 3: Configure DRTE Settings - Custom Setting

1. Navigate to **Setup**, search for **Custom Settings**, and manage the **DRTE Settings** record.



2. Populate the following fields with the required values:
  - **CertificateOfCompletionTemplateId**: Enter the **Salesforce ID** (e.g., `a01XXXXXXXXXXXXXX`) of the "Certificate of Completion" Document Template record created in '**Configuration**'.
  - **CertificateOfCompletionDocumentPrefix**: Provide a unique text prefix (e.g., `COC-`) for the generated document's generic ID.

Custom Setting  
**DRTE Settings**

[Help for this Page](#) 

If the custom setting is a list, click **New** to add a new set of data. For example, if your application had a setting for country codes, each set might include the country's name and dialing code.

If the custom setting is a hierarchy, you can add data for the user, profile, or organization level. For example, you may want different values to display depending on whether a specific user is running the app, a specific profile, or just a general user.

[Edit](#) [Delete](#)

▼ Default Organization Level Value

Location  
gptversion (Deprecated) - No Use  
Esign Guest Site URL  
Company Footer Text For Email Template  
Company Support Email for Email Template  
Send Signature Request Email Template Id  
Share PIN Email template Id  
doctemplateid\_certcompletion  
CertificateOfCompletionTemplateId  
a1rT10000JMNPhIA

endpoint (Deprecated) - No Use  
key (Deprecated) - No Use  
Auto Send eSign 2Signer Email\_Template\_Id  
Company Header Image For Email Template  
Org Wide Email To Send Signature Request  
Send Signed Document Email Template Id  
User Story Sign Off Email Template Id  
Signature Reminder Email\_Template\_Id  
GenericDocId Prefix

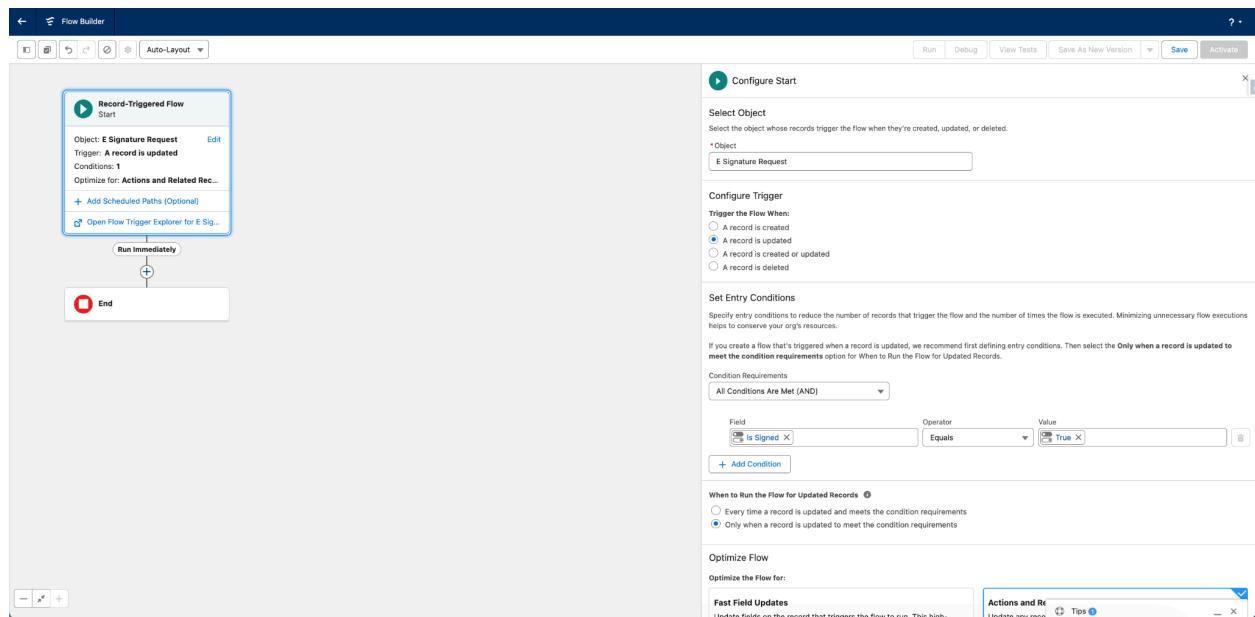
DOC-

[View: All](#) [Create New View](#)

# Configuration - Automatic Signed Document (w/CoC) Creation after Signature

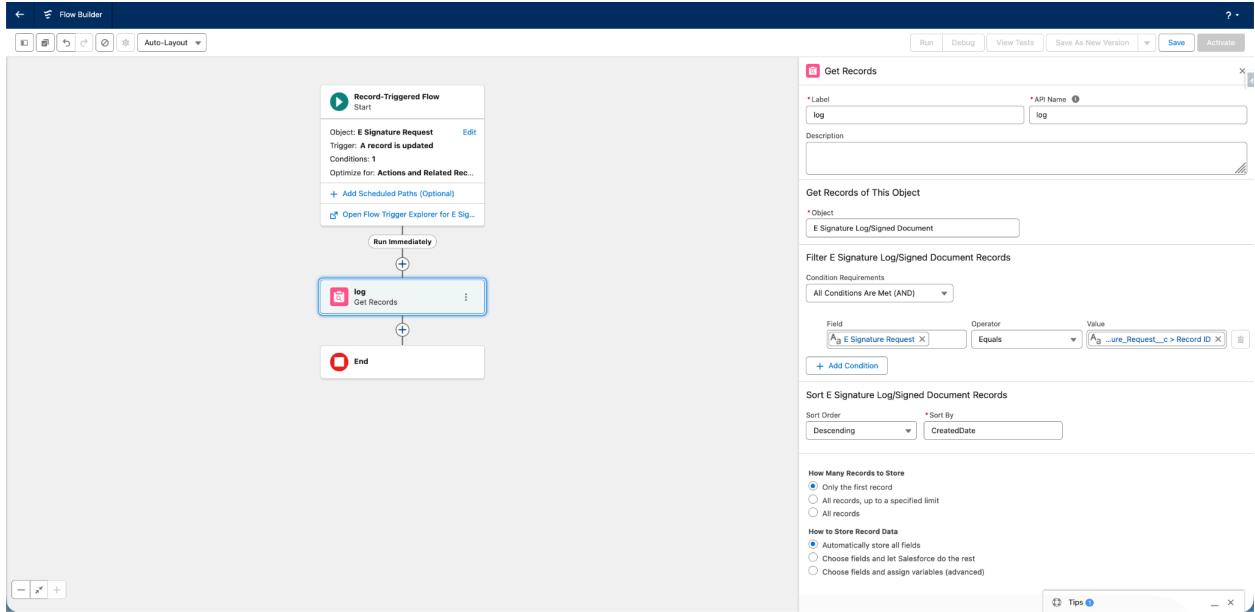
## Step 1: Create a new Trigger Flow on E Signature Request Object.

Trigger the Flow when record is updated, Set Entry Condition “isSigned” equal to “TRUE”. Only When Record Is Updated to Meet the conditional requirements



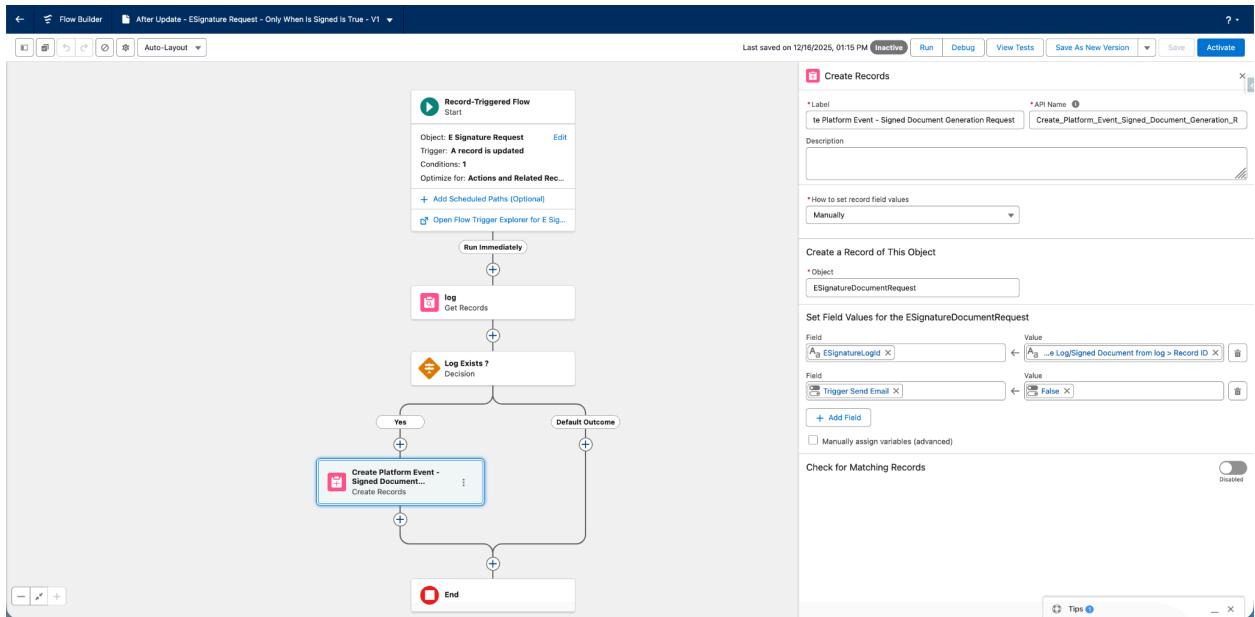
## Step 2: Query Related E Signature Log (Child Record)

Use Get Records element and query as shown in the screen shot below



## Step 3: Create a packaged platform event record.

To trigger the functionality to generate Signed Document with CoC and attach to related document record. (**Imp:** You may set Trigger Send Email equal to True, to send automatic email to related contacts/signees with signed document)



## Step 4: Configure the scheduled path for this functionality.

This object holds critical functionality for e-signature request generation and handling the response after signature. To make sure your custom code doesn't hinder e-signature native functionality, move the logic to Async Processing (as scheduled path shown below)

