VolunteerMatters Salesforce App Technical Guide

Overview

This app connects your Salesforce account with VolunteerMatters to keep contact records in both systems perfectly in sync—automatically and in real time. You can easily choose which fields from your Salesforce Contacts should match the corresponding fields in VolunteerMatters.

Once set up, the app continuously monitors updates in either system. Any time a contact is added or changed in Salesforce or VolunteerMatters, those updates are instantly reflected in the other, ensuring your contact information stays accurate, consistent, and up to date everywhere.

Purpose

This app is designed to save you time and prevent errors by removing the need to manually update contact information between Salesforce and VolunteerMatters. It automatically syncs contact records change, the two

systems, so no matter where you make a change, Salesforce or VolunteerMatters, your data stays consistent and up to date.

With fewer manual updates and no conflicting records, your team can work more efficiently, trust the data they see, and spend less time managing contacts across systems.

Key features

Dynamic Field Mapping

- → Easily map fields between Salesforce Contacts and VolunteerMatters Contacts using an intuitive interface.
- → Supports custom and standard fields, allowing flexible, bidirectional synchronization.

Simple Configuration

- → No-code setup for admins and users.
- → Guided configuration steps make it easy to connect your Salesforce org with VolunteerMatters system.

Quick Installation

- → Lightweight managed package with minimal setup requirements.
- → Designed for quick deployment in both sandbox and production environments.

Real-Time Sync

- → Automatically updates records in either system when changes occur.
- → Reduces manual effort and prevents data inconsistencies.

volunteermatters
Let's do something extraordinary

salesforce

Supported Salesforce editions

- Professional
- Enterprise
- Unlimited
- Developer

Step-by-Step Installation and Configuration instructions

Follow the steps below to configure your Salesforce org to sync with VolunteerMatters database.

volunteermatters
Let's do something extraordinary

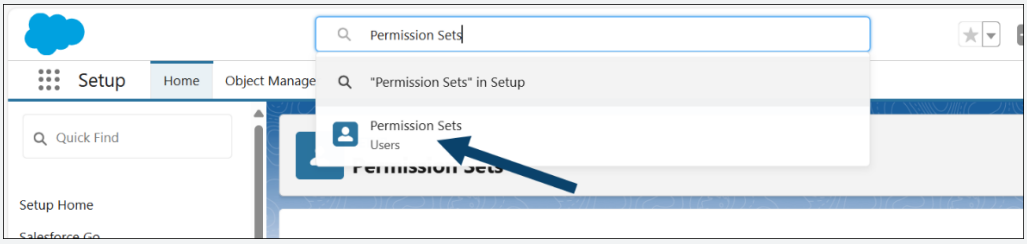
salesforce

Install APP (AppExchange)

- → Visit **AppExchange**
- → Type **VolunteerMatters** Connector in the Search AppExchange field and select Enter
- → Select the app and click 'Get It Now'
- → Use your Salesforce credentials to login to the org in which you want to install the app
- → Select Install for All Users and click Install
- → In case of a first-time installation, you will have to approve third-party access
- → In Approve Third-Party Access, select Yes, grant access to these third-party web sites
- → In some cases, it is possible that you get a message that the app is taking longer to install. You will receive an email when the installation is done

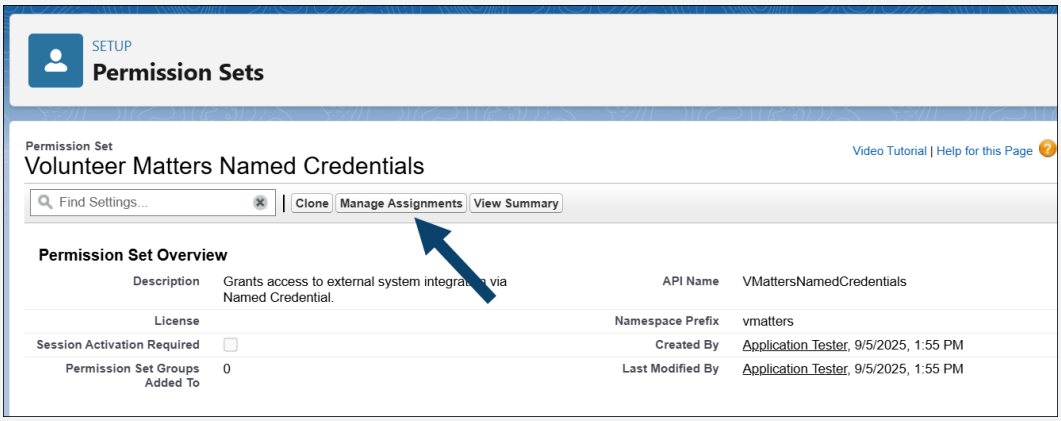
Assign permission set to users

- → Within Setup, enter **"Permission Sets"** in the search box and click on it.

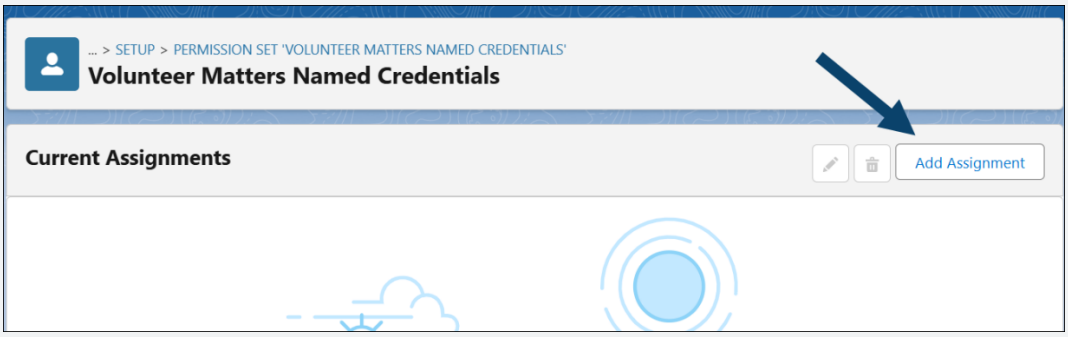


- → First you will need to Assign the '**Volunteer Matters Named Credentials**' permission set to the **Automated Process** user and the **User** who will perform the initial configuration

  When on Permission Sets, click on '**Volunteer Matters Named Credentials**', then click the **Manage Assignments** button.



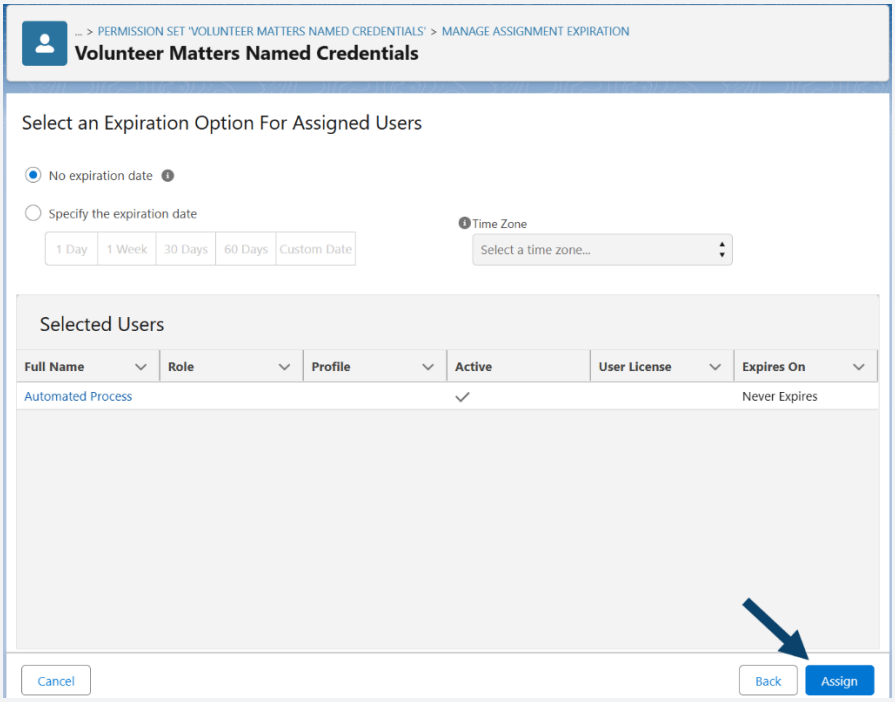
Then click **Add Assignment.**



Note: For the Automated Process User, you may need to enter “Automated” in the search box and click enter for it to be returned.

*In some Salesforce orgs, the **Automated Process** user may not appear in the user search when assigning permission sets via the UI. If this occurs, you can assign the permission set using **Execute Anonymous Apex**. See the workaround instructions below.*

→ Choose the **Users** you want to have this permission set and click next, then click the Assign button and then Done. You will have to repeat this process for the user who is running the initial configuration, if you had to search for the automated process user.



Next you will need assign the '**Volunteer Matters Permission set**' to the **Users** you want to configure the mapping between Contact records and Volunteer Matters database.

→ When on Permission Sets, click on '**Volunteer Matters Permission set**, then click the **Manage Assignments** button. Then click **Add Assignment.**

Choose the **Users** you want to have this permission set and click next, then click the Assign button and then Done.

Workaround: Assign Permission Set to Automated Process User

Prerequisites

Ensure you have the following permissions:

- **API Enabled**
- **Author Apex**

volunteermatters
Let's do something extraordinary

salesforce

Steps

1. Navigate to **Setup** within your Salesforce system.
2. Click the **gear icon** (top right) and select **Developer Console**
3. In the Developer Console, go to:
    - **Debug → Open Execute Anonymous Window**

Execute the Following Code

Paste the code below into the Execute Anonymous window:

```
User u = [SELECT Id FROM User WHERE Name = 'Automated Process' LIMIT 1];

PermissionSet ps = [SELECT Id FROM PermissionSet WHERE Name = 'VMattersNamedCredentials'
LIMIT 1];

insert new PermissionSetAssignment(
    AssigneeId = u.Id,
    PermissionSetId = ps.Id
);
```

Then click **Execute**.

Notes

- VMattersNamedCredentials is the **API Name** of the permission set.
    - If needed, verify this in **Setup → Permission Sets**
- This method directly assigns the permission set when the Automated Process user is not available in the UI
- No confirmation message is shown, but the assignment can be verified under:
    - **Permission Set → Volunteer Matters Named Credentials → Manage Assignments**

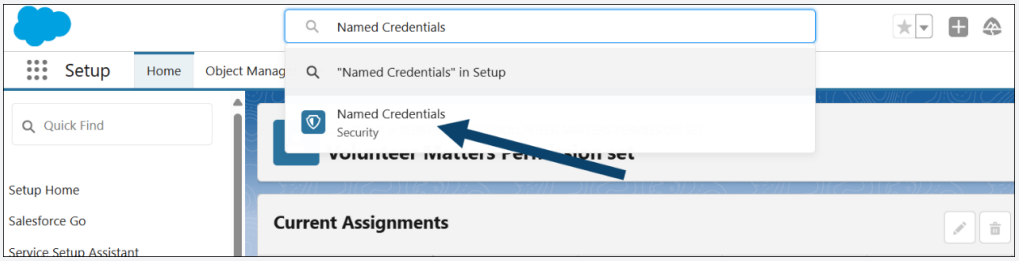
When to Use This

Use this workaround only if:

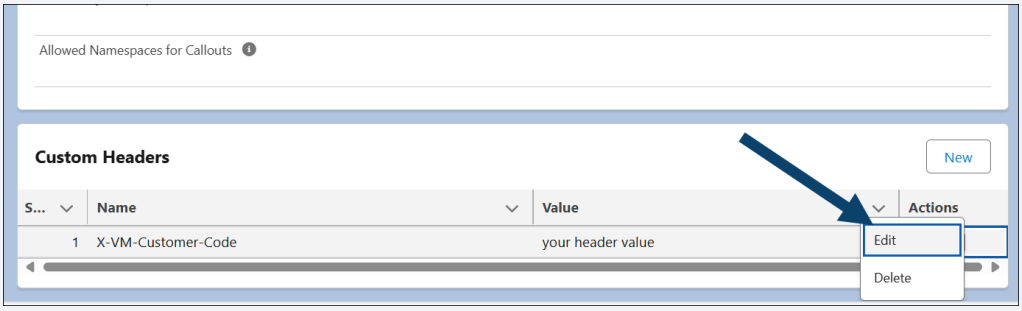
- Searching for **"Automated Process"** does not return a result during permission set assignment

Configure Named Credentials

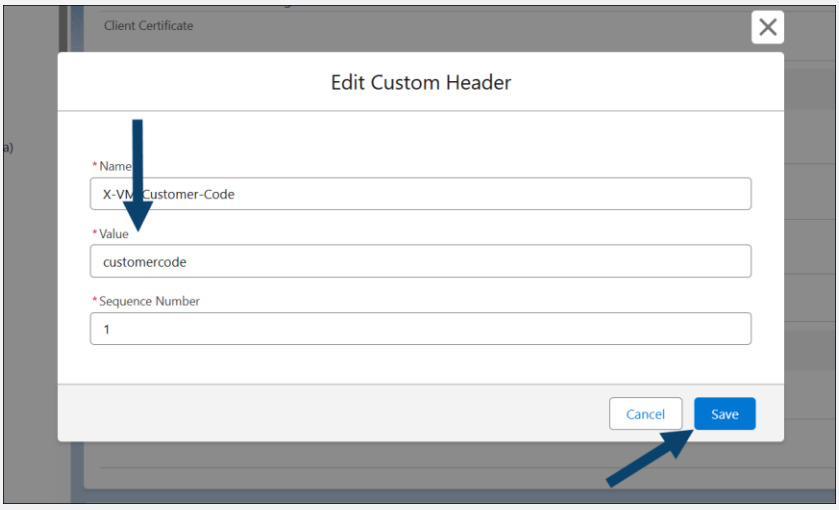
→ Within Setup, enter “**Named Credentials**” in the search box and click on it.



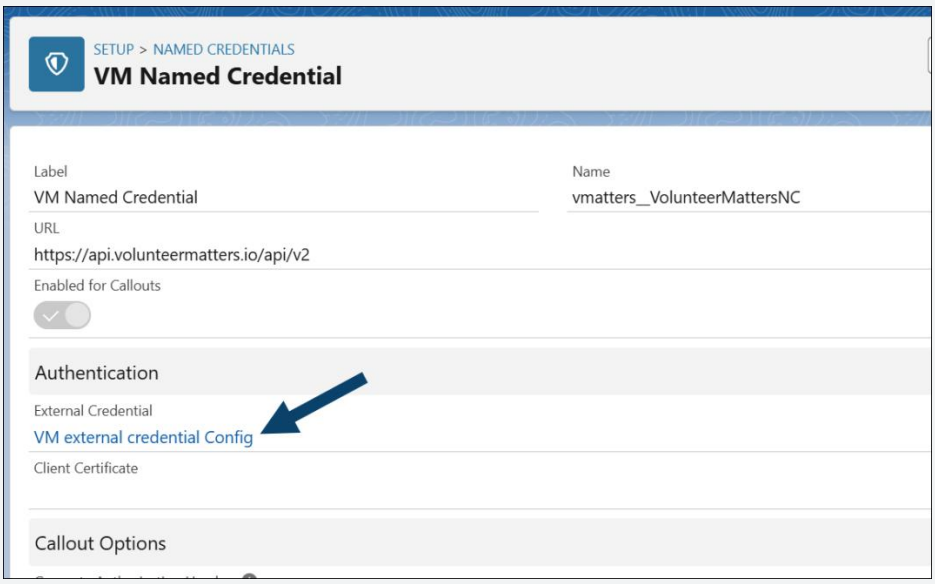
→ Click on the ‘**VM Named Credential**’ named credential. Scroll to the bottom and edit the ‘**X-VM-Customer-Code**’ in **Custom Headers** section.



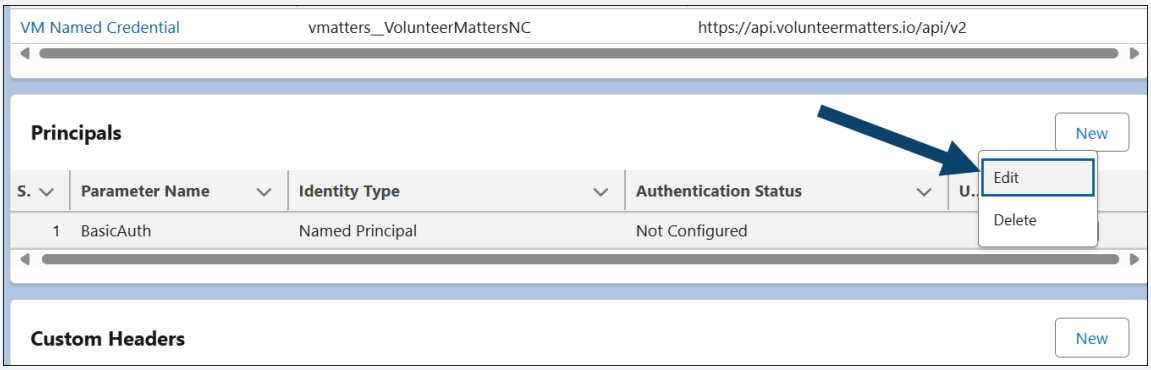
Then enter your VolunteerMatters customer code in the “Value” field which should have ‘**your header value**’ placeholder text and click the “Save” button



→ Once returned to the Named Credential page for the “**VM Named Credential**”, click on the “**VM external credential Config**” **External credential** in the ‘**Authentication**’ section



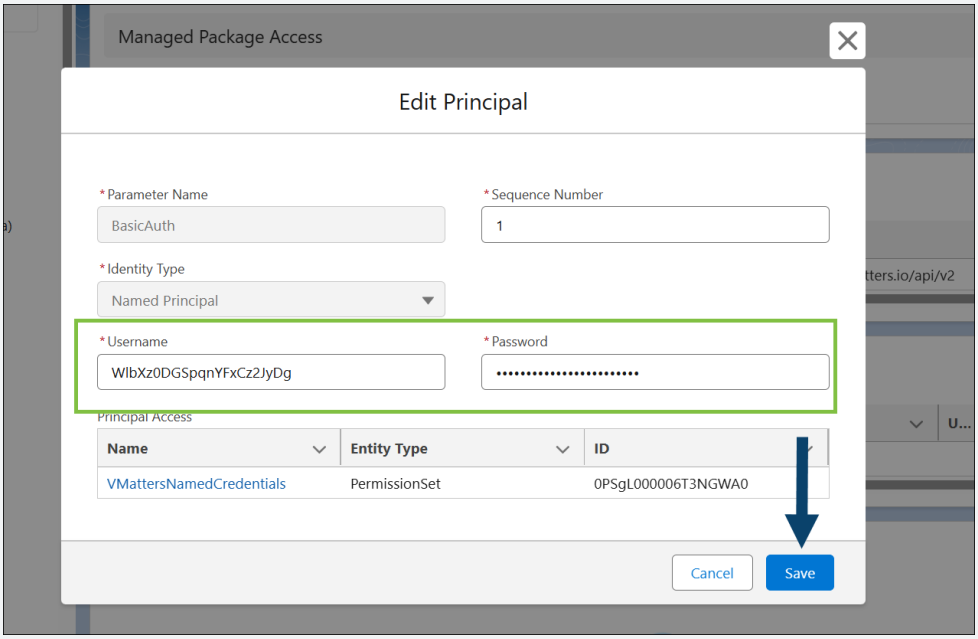
→ Then in the ‘**Principals**’ section click dropdown and select “Edit” for the ‘**BasicAuth**’ row.



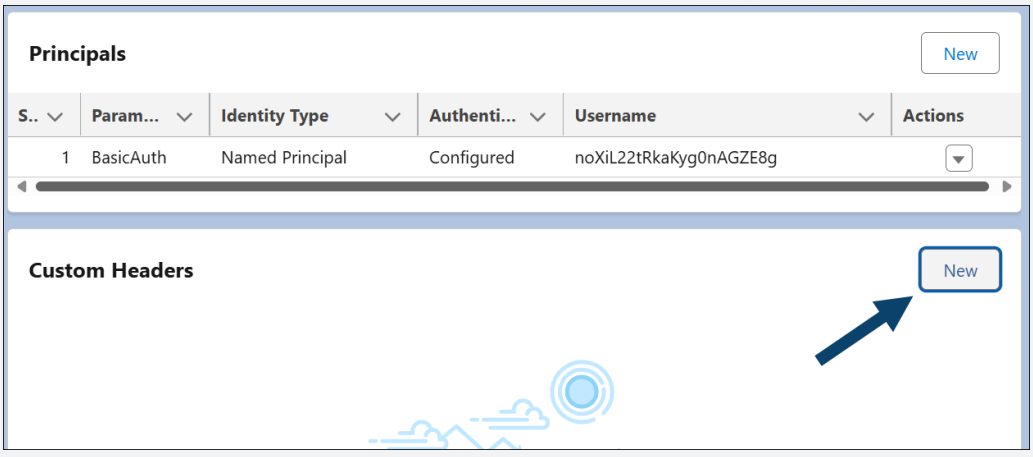
Enter your VolunteerMatters API Key and Secret into the **Username** and **Password** fields respectively and click the “Save” button. For more information on creating API Keys in your VolunteerMatters system click here. The “System API Version 2 - All Access” Authority is required for the VolunteerMatters Salesforce Connector.

Note: Before saving, verify that there is a principal access assignment for the

permission set



→ Under the Prinicipals section, there will be another Custom Headers section. Click “New” to create a new custom header.



Configuration Settings:

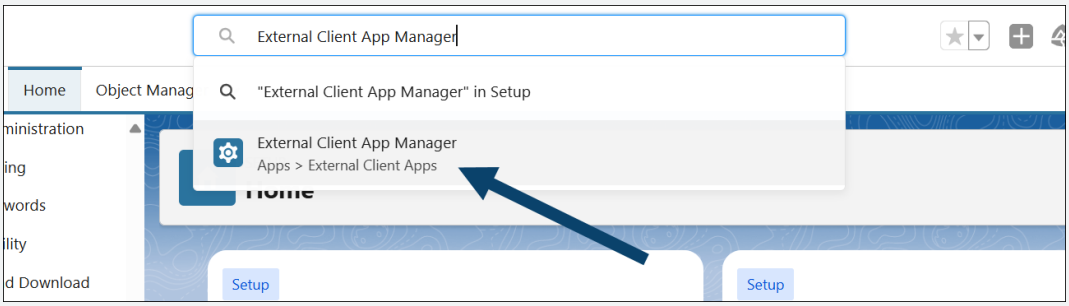
Name: X-VM-Customer-Code
Value: VolunteerMatters Customer Code

Then click “Save”.

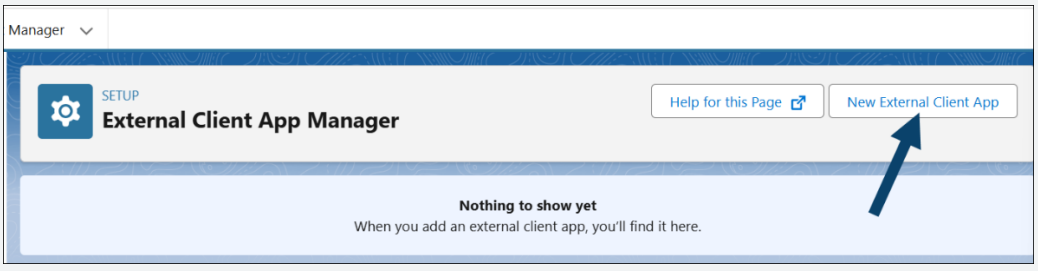
→ Navigate back to the initial “**Named Credentials**” area, click on the **VM Standard fields Name Credentials**
→ Repeat the step with the **Custom Headers** section from the previously configured named credential using the VolunteerMatters Customer Code and save.

Create Connected app/ External client app

→ Within Setup, enter “**External Client App Manager**” in the search box and click on it.



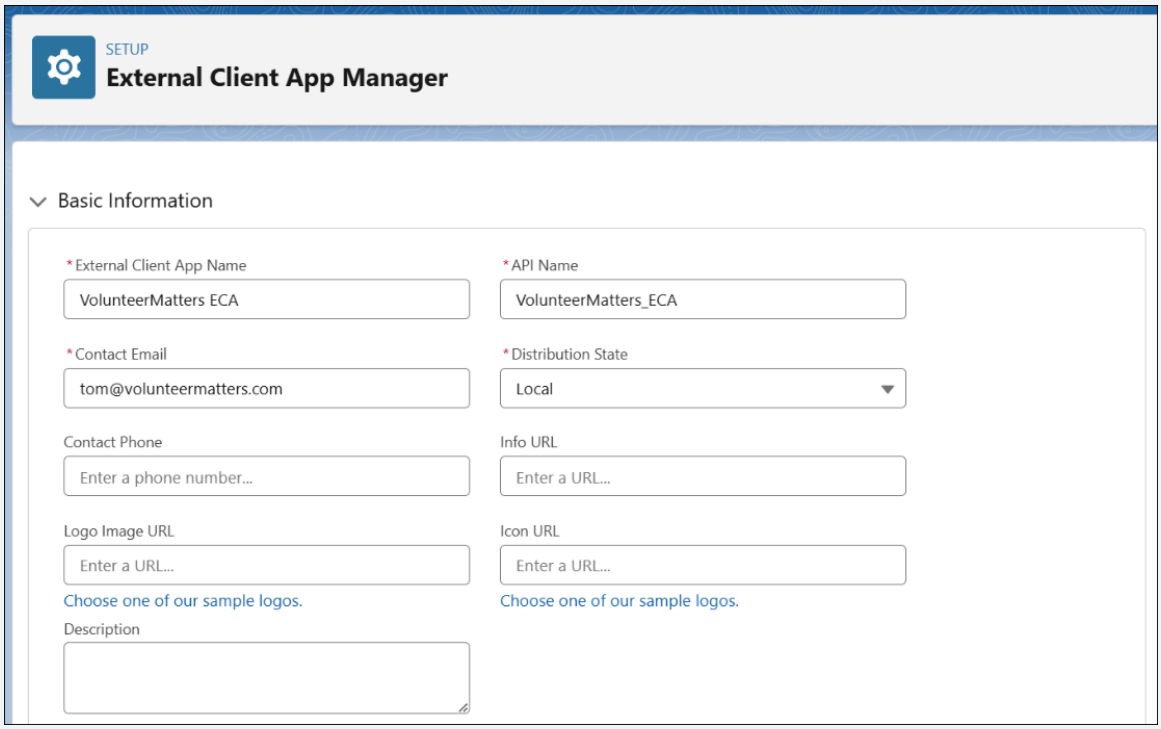
→ Then click on the “**New External Client App”** button.



→ Here you will create an app with the following settings

- **External Client App Name**: Any name you like (For Example VolunteerMatters ECA)

- **Distribution State**: Local
- **Contact Email**: One you have access to.

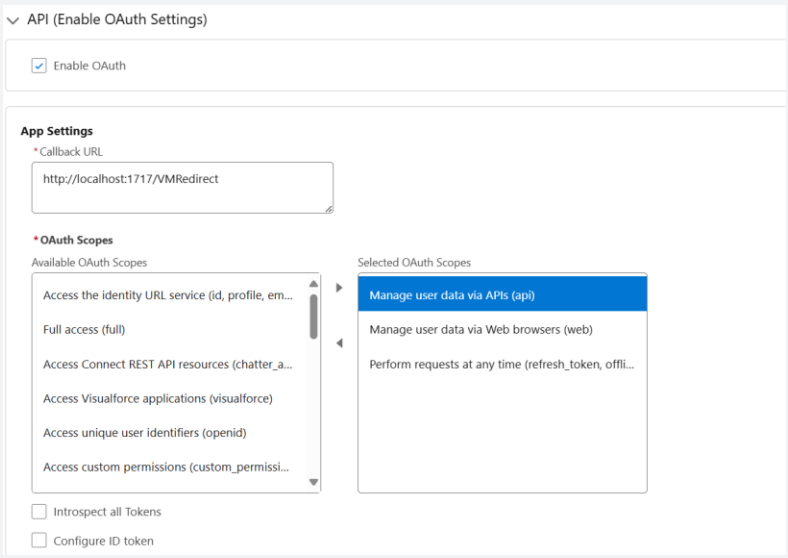


→ Check the box to **Enable OAuth** under **API (Enable OAuth Settings)** and enter the following:

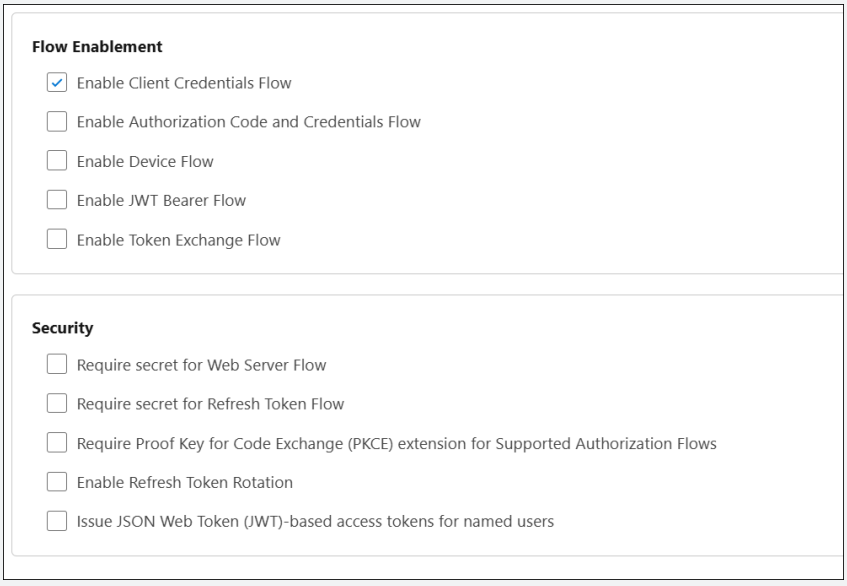
→ **Callback URL**: http://localhost:1717/VMRedirect

→ **OAuth Scopes:** – select and add the following scopes:

- Manage user data via APIs (api)
- Manage user data via Web browsers (web)
- Perform requests at any time (refresh_token, offline_access)

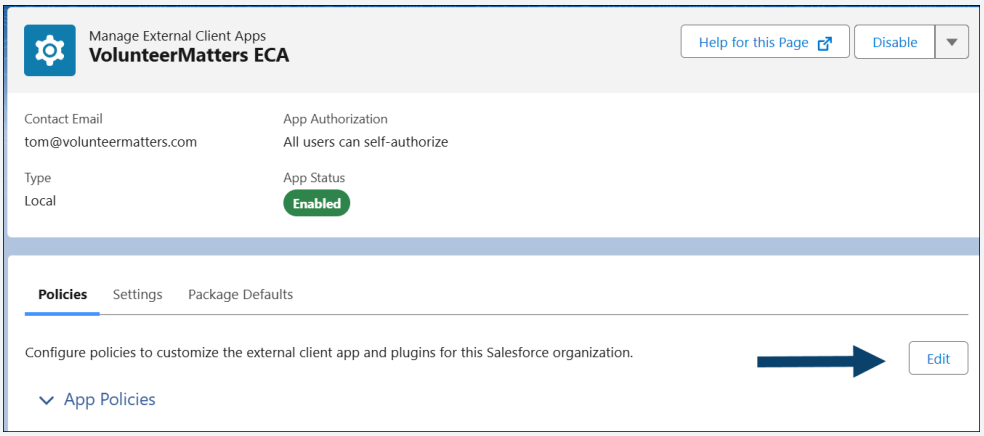


→ Check the box to **Enable Client Credentials Flow** under the **Flow Enablement** section and uncheck any **Security** Boxes that may have been checked by default.



→ Then click the **Create** button.

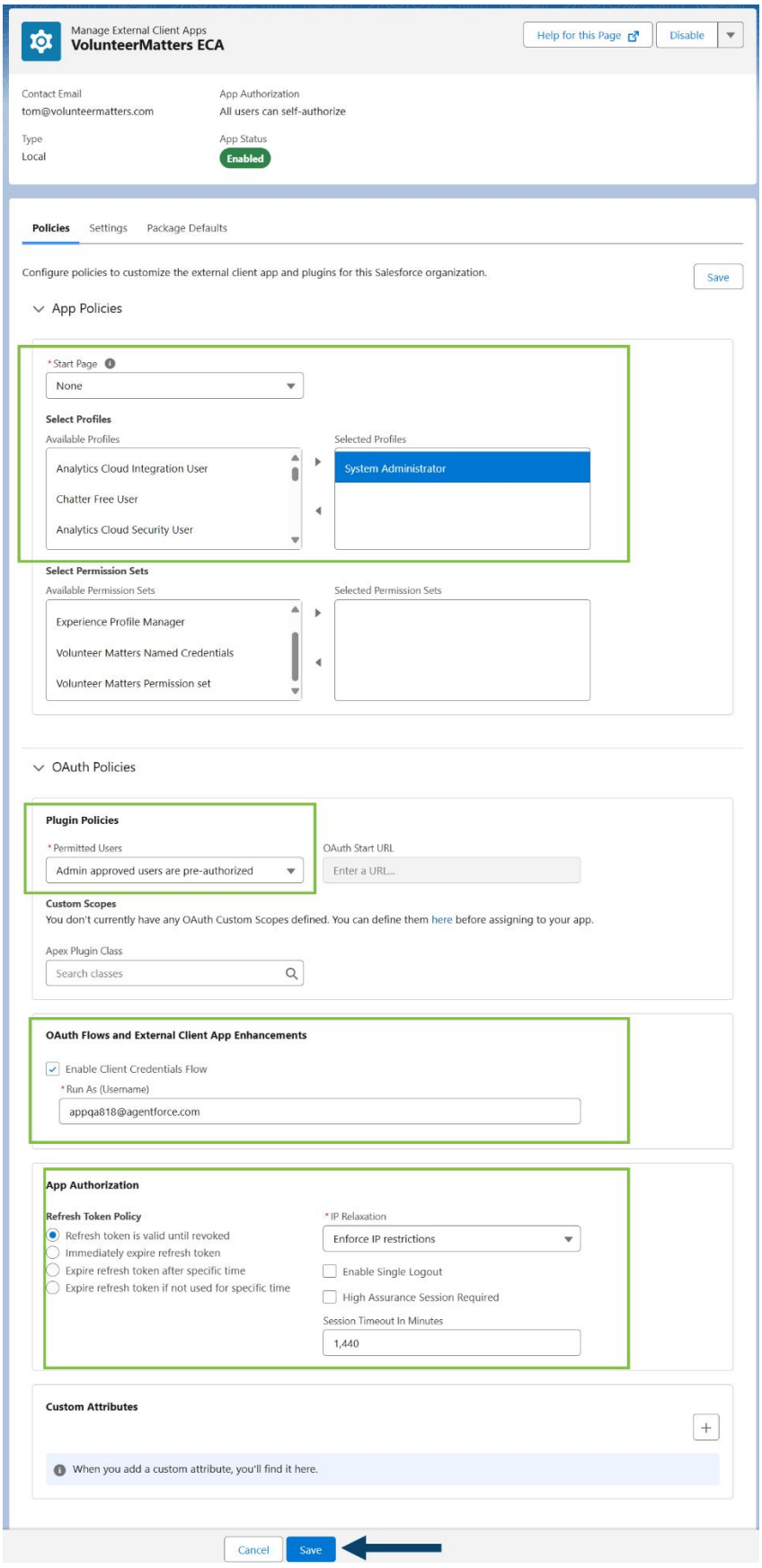
→ After creating the app, go to the **Policies** tab of the created app and edit the policies as follows.



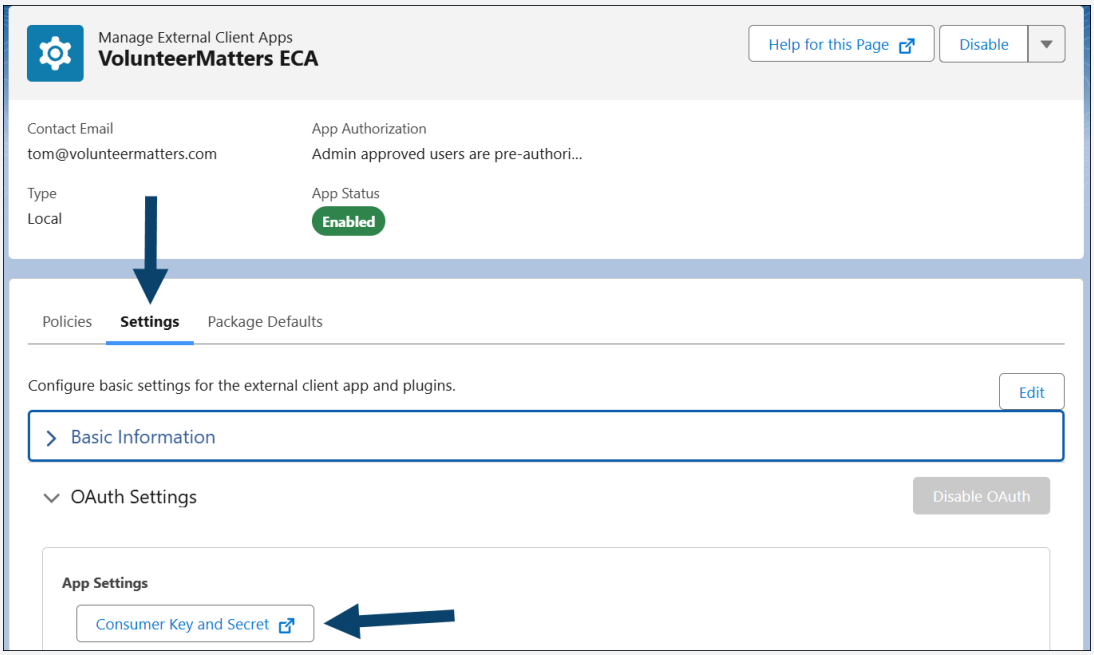
- → **Start Page** - None
- → For the **Permitted Users** drop down in the **OAuth Policies** section, choose **Admin approved users are pre-authorized.** You will need to confirm the policy by clicking OK. In the new **Selected Profiles** section displayed, choose the profile of the user you choose for 'Run As' (typically will be System Administrator).
- → **OAuth Flows and External Client App Enhancements:** Check **Enable Client Credentials Flow** and choose a user with a profile as chosen above. This will typically be the username of the primary setup account but must share the selected profile.
- → In **App Authorization** choose '**Refresh token is valid until revoked**' and input 'Session Timeout In Minutes' – **1440.**
- → Then click **Save**

volunteermatters
Let's do something extraordinary

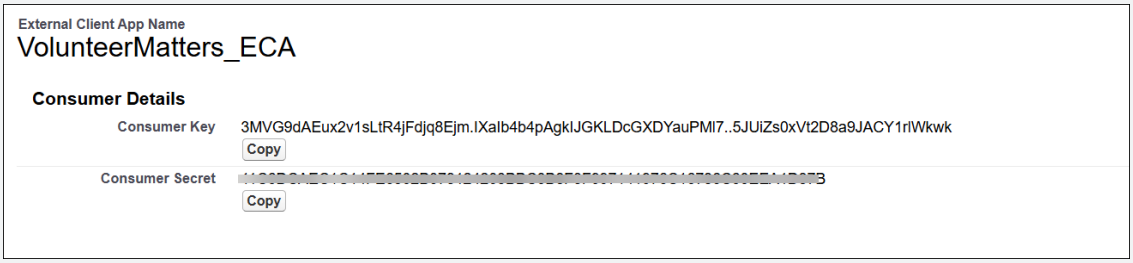
salesforce



→ After saving, go back to the **Settings** tab of the External client app => **OAuth settings** and click on '**Consumer Key and Secret'**.



→ A verification email will be sent to the account used. Once you enter the code, you will see the Consumer Key and Secret. Please record these as you will need them later in the configuration to enable sending data from VolunteerMatters to Salesforce.

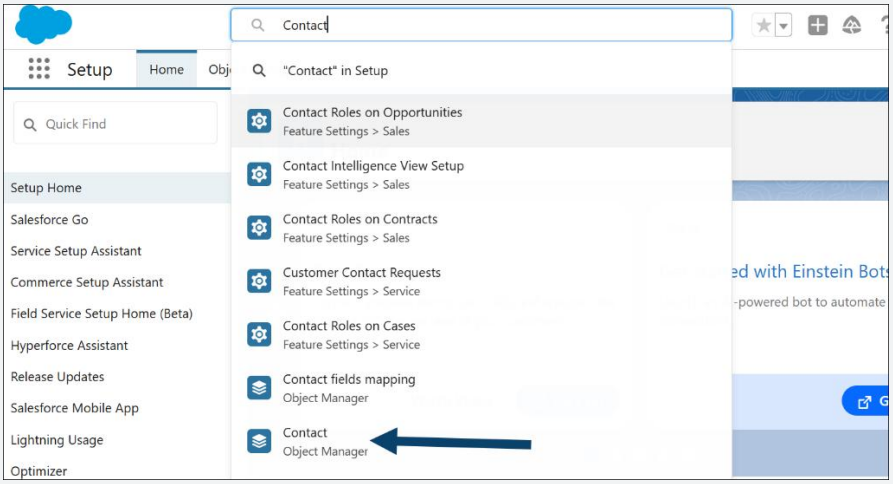


volunteermatters
Let's do something extraordinary

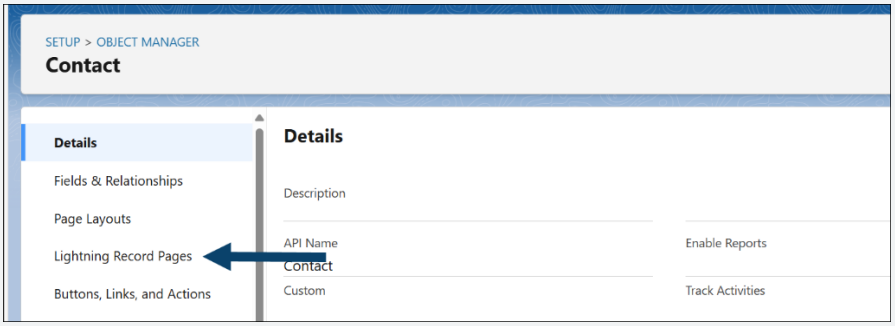
salesforce

Clone Automatically Created Lightning Contact Page Layout to be able to customize

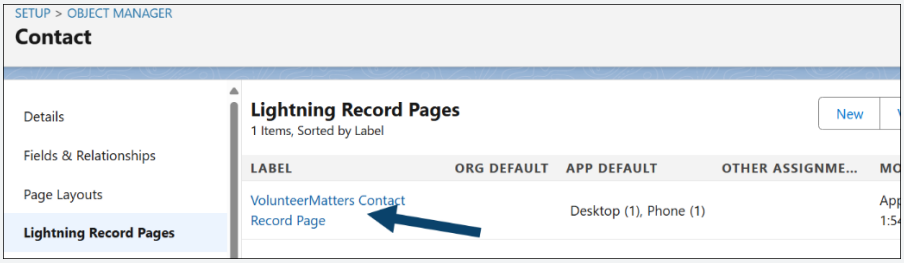
→ Within Setup, enter "**Contact**" in the search box and click on it.



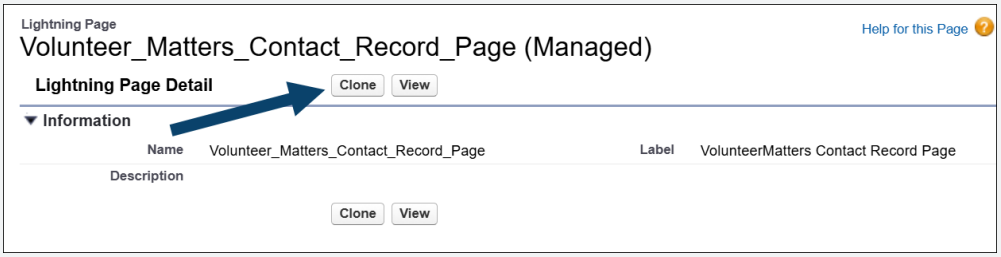
→ Then click on the "**Lightning Record Pages**" link.



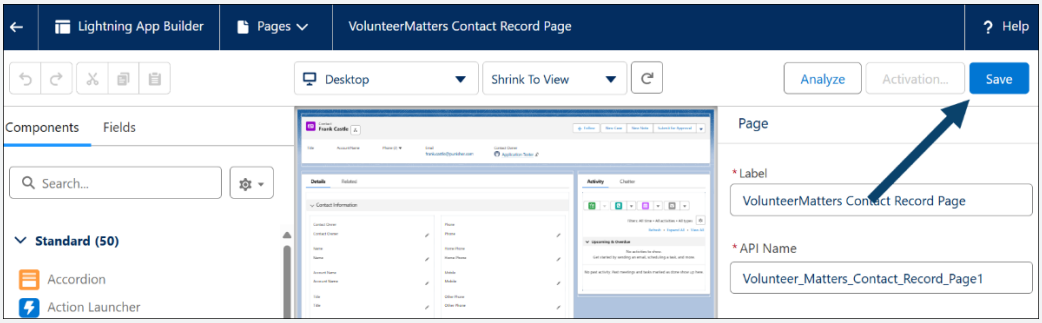
→ Click on the "VolunteerMatters Contact Record Page" link.



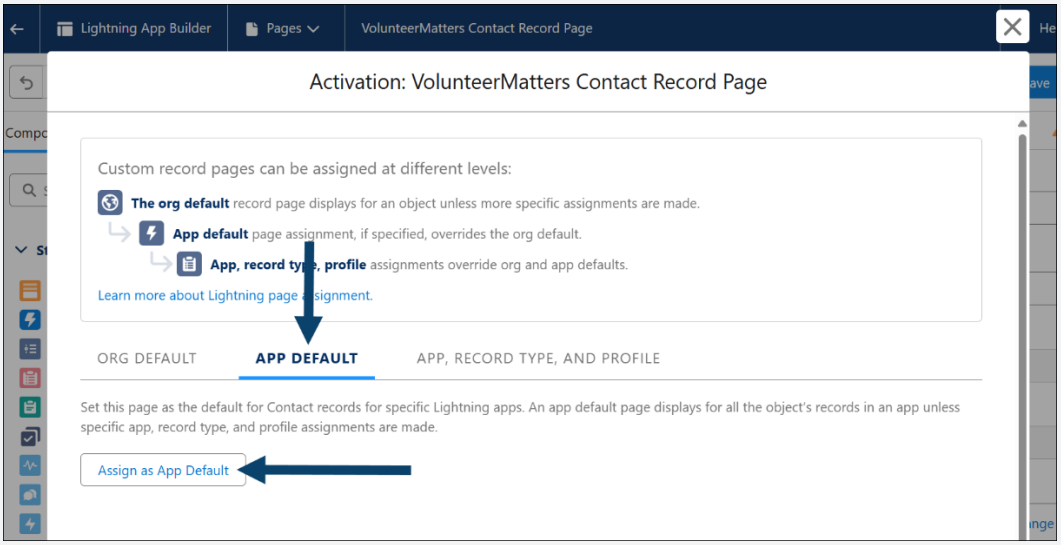
→ Click the “Clone” button.



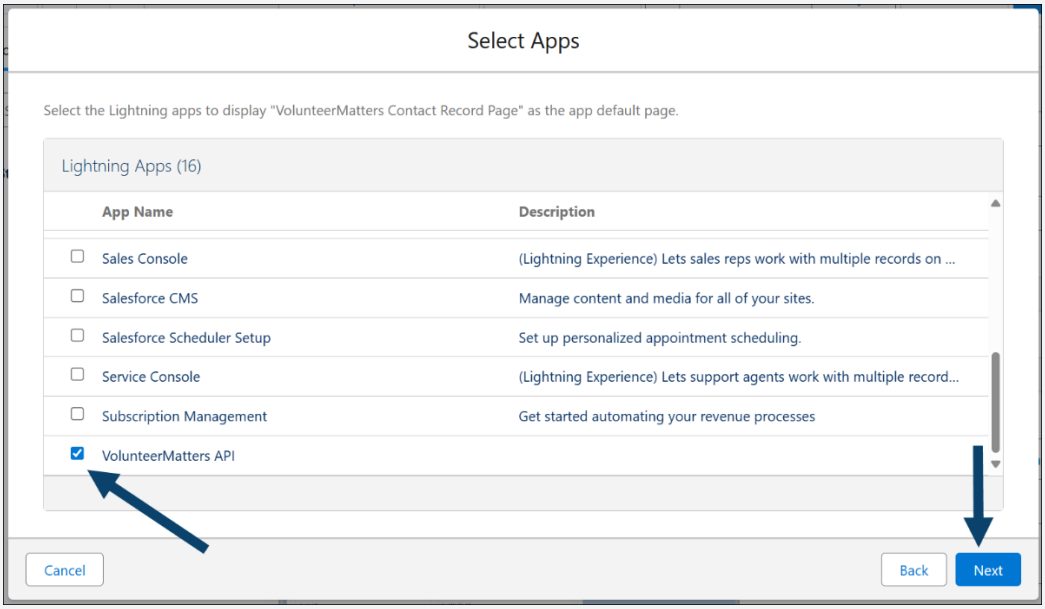
→ Add “New” to the end of the label. Other than that, you do not need to update at this juncture. Just click “Save” and “Activate”.



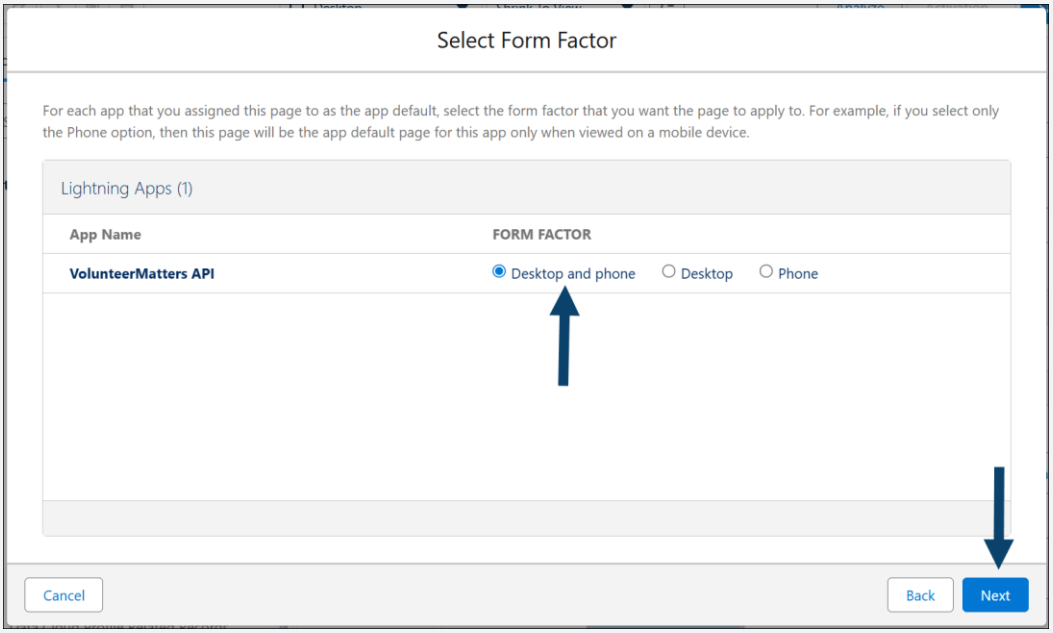
→ Next click on APP Default section and click the “Assign as App Default” button.



→ Then scroll down and check the “VolunteerMatters API” App and click “Next”.



→ If not already preselected, choose “Desktop and phone” as the Form Factor and click “Next”.

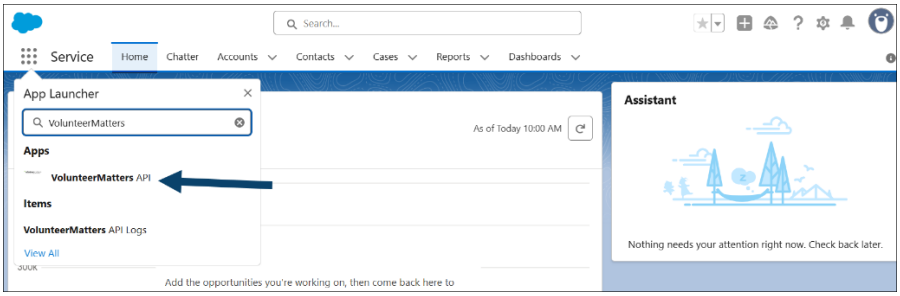


→ Review and Save. If needed, you can now update the page (instructions below).

VolunteerMatters API Configuration Steps

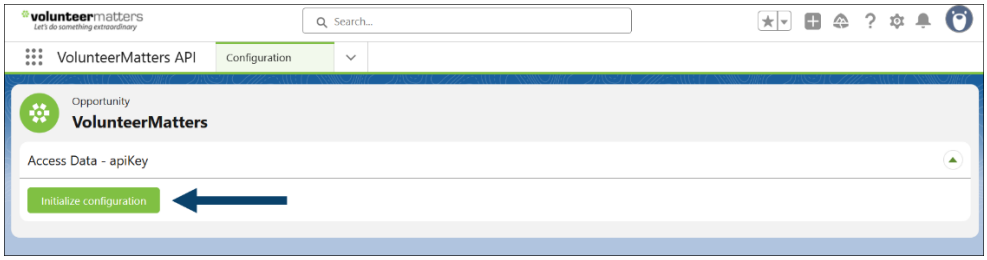
Initialize the APP and enter VolunteerMatters API Key/Secret and Customer Code

→ Go to the VolunteerMatters API app from the app launcher

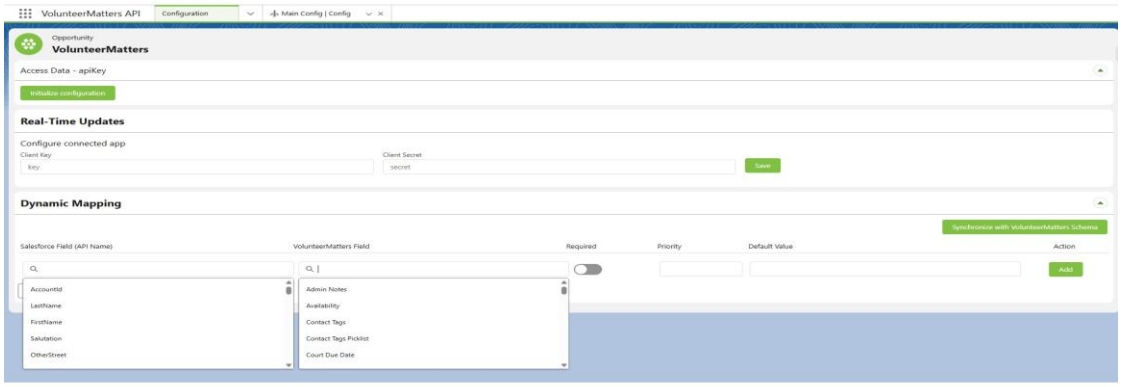


→ Go to the **Configuration** tab (you may be defaulted to it)- the first visit should create a Main Config record. A pop up should appear letting you know.

→ Click **Initialize configuration** button it should retrieve the fields from Volunteer Matters

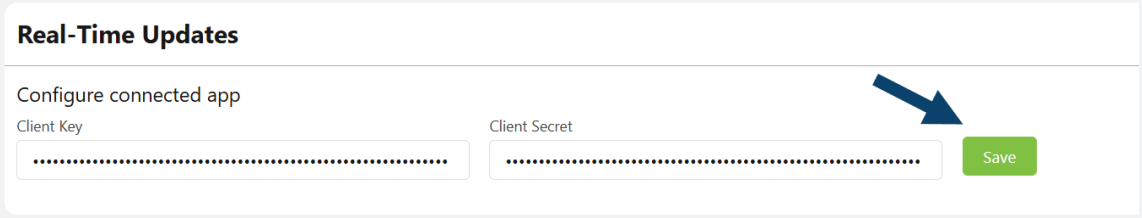


You should see the sections below and the fields in the VolunteerMatters field column should be available

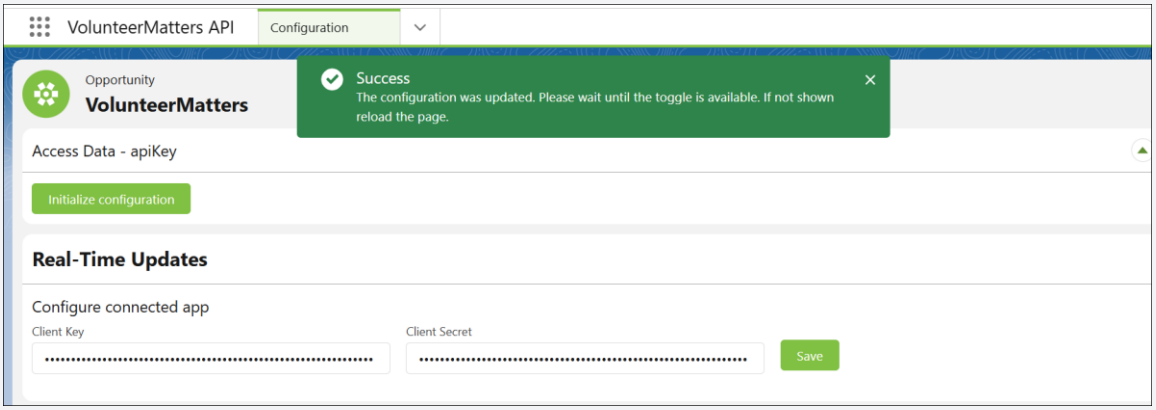


Enable inbound connection

→ In the **Real-Time Updates** section, enter the Consumer Key and Consumer secret of the External App that you noted above and click “Save”.



→ You will see a pop up if successful. In some cases, you might need to refresh the page to see the toggle switch to enable the updates.



→ After you refresh the page, the **Real-Time Updates Activate** toggle should be off, and the “current status connection” will be **Inactive**. We would

recommend completing the next dynamic mapping section before toggling it on.

Create mappings for records

Under **Dynamic Mapping**, you can map your fields in Salesforce to the corresponding VolunteerMatters fields. As well as mark the fields required, set priority points on fields to break a potential tie when there is a potential matching record in Salesforce.

Priority points work in conjunction with the **Threshold** setting on the configuration page. The threshold represents the minimum number of priority points required to match an existing Salesforce contact.

When **tiebreaking** is needed, the record must first meet or exceed the threshold configured. If multiple Salesforce contacts meet that threshold, the one with the highest priority points total will be selected. If multiple records have the same total, the first record found will be matched.

The matching logic when a record is created or updated in VolunteerMatters is as follows:

- If a Salesforce Contact does not have an external ID from VolunteerMatters:
    - If Priority Points Sum < Threshold, no match occurs, and a new Contact is created in Salesforce, even if another record shares some matching fields.
    - If Priority Points Sum ≥ Threshold and a similar Contact exists, that record will be matched and updated.
        - If multiple Contacts qualify, the one with the highest priority points is matched.

    - If there's a tie, the first record is matched.

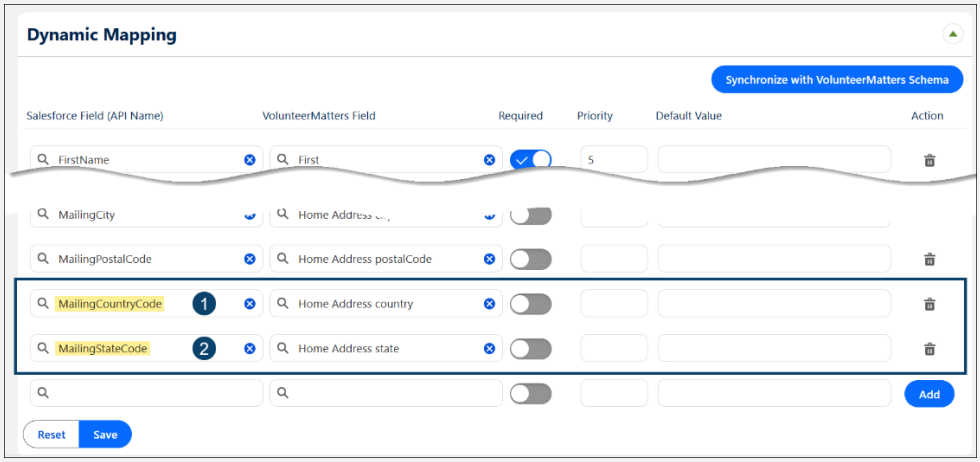
- If a Salesforce Contact does have an external ID from VolunteerMatters:
    - It is considered already matched and will be updated with any mapped information.

ADDITIONAL NOTES:

- **IMPORTANT: Address setup and configuration information.**
    - With Addresses in Salesforce, if you are using a picklist the values submitted from VolunteerMatters MUST match or the synch will fail. You will need to complete the following:
    - Contact VolunteerMatters support and have them configure a choose from a list state value for all address editing pages (Registration, My Profile, etc.).

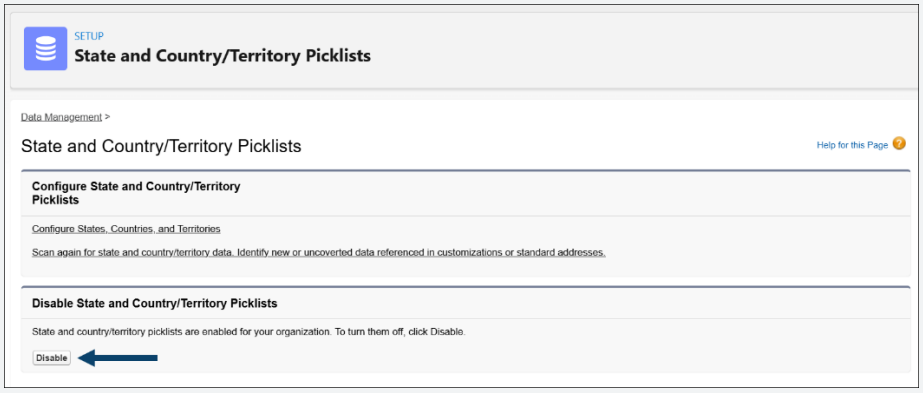
    - Within the VolunteerMatters Salesforce APP configuration mapping, you must configure the Country field above the State field. This is because with Salesforce State/Country picklist, the state values are only displayed after the country is selected.
    - Within the VolunteerMatters Salesforce APP configuration mapping, you need to select the CountryCode and StateCode fields within

Salesforce as that is the information sent from VolunteerMatters.



- For more information on Salesforce State and Country/Territory Picklists, click here.

- NOTE: You can always disable the State and Country/Territory Picklists within Salesforce via the Setup area, and it will not affect any synch and you can use the standard Country and State fields within Salesforce in the dynamic mapping.

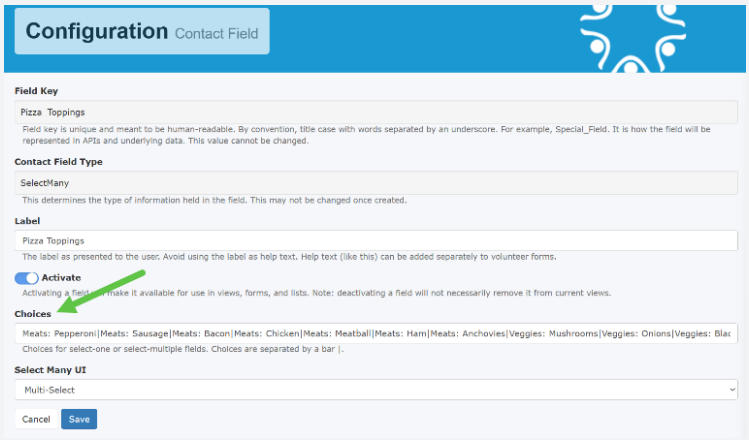


- You can choose any standard or custom field.
- To add additional field rows, click the “Add” button. You can only add one row at a time.
- For Required fields in Salesforce, toggle the “Required” button. This would mean that whenever a record is updated on VolunteerMatters side and sent to Salesforce it should include a value in this field. If there is no value and there

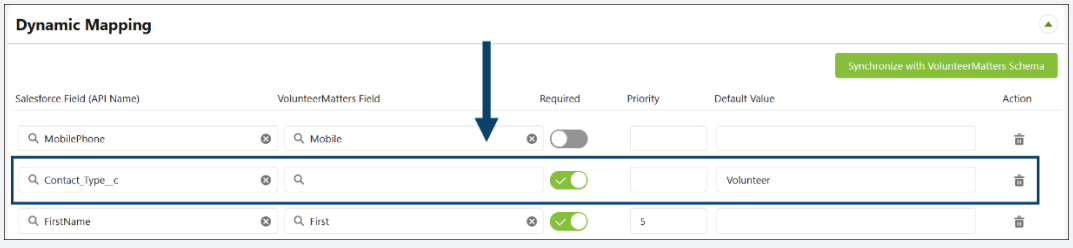
is no default value, the Salesforce record will not be updated.

If a required field is not compatible with the matched Salesforce field, no update is performed. For example, if the Salesforce email field is mapped to a non - email field in VolunteerMatters the update will be blocked.

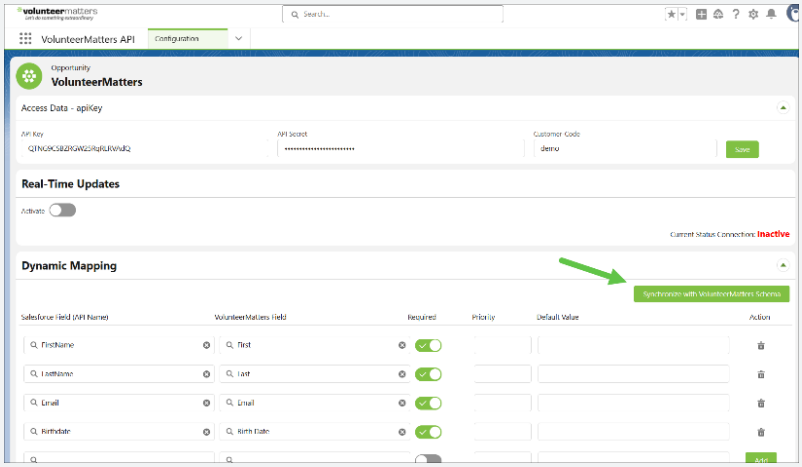
- For Select from a List/Picklist fields, the values in both systems must match exactly. For those values in VolunteerMatters that use Groups (ie: Meats: Pepperoni|Veggies: Mushrooms), the values in Salesforce must match those exactly with the group and colon. Basically, match the choices as designated in the VolunteerMatters field configuration.



- For those fields required in Salesforce but with no corresponding field in VolunteerMatters, in Dynamic Mapping, you will select the Salesforce Field, leave the VolunteerMatters Field empty, and add the Default Value for all contacts. This field must be toggled as “Required” for the field to be populated in the Saleforce Contact. For example, if all contacts would have “Volunteer” with the Salesforce Contact Type field.



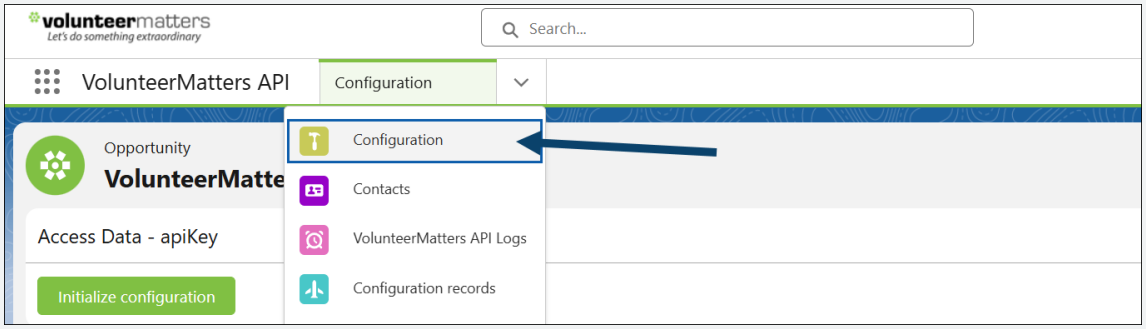
- If you add new fields to VolunteerMatters, you will need to utilize the “Synchronize with VolunteerMatters Schema” button in **Dynamic Mapping** to pull those fields into the drop-down.



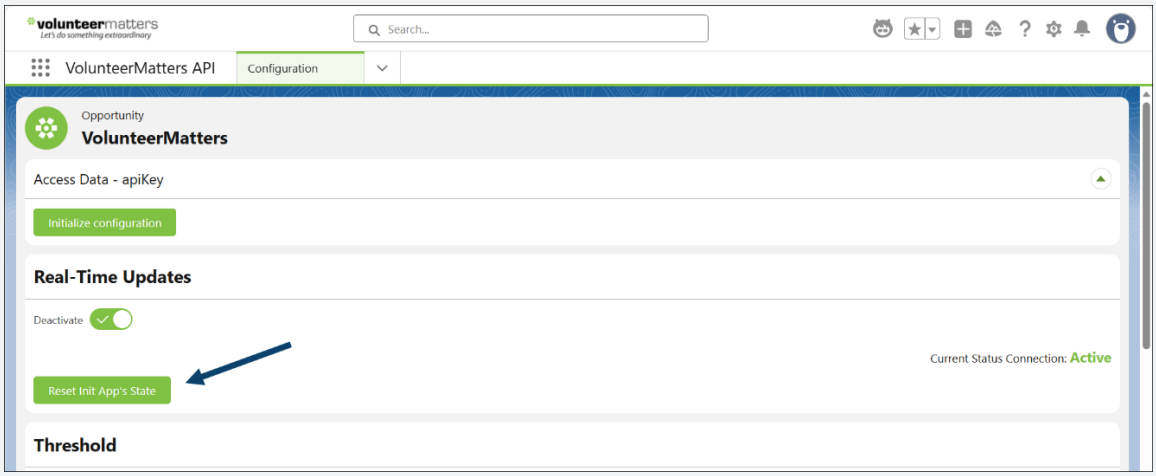
Once you are satisfied with your mapping, Under **Real-Time Updates**, toggle the “Activate” button to turn on. Please note that this can be done either before or after the mapping is completed. If the **Dynamic Mapping** changes are “Saved” they will be included.

Resetting Connection

In rare instances, you may need to reset the webhook connection with VolunteerMatters. To do so, open the Configuration section of the VolunteerMatters App.



Once open, click the “Reset Init App's State” button.



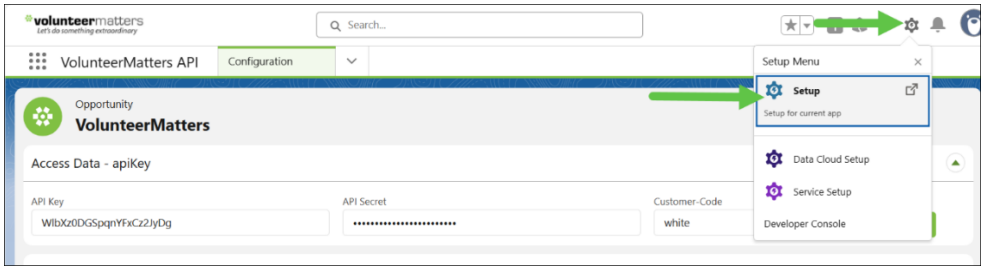
This mechanism allows the administrative user to be able to reset communication with the webhooks. Once clicked, click “OK” button to proceed. This will reinitialize the configuration. Then reenter the Consumer Key and Consumer secret of the External App and click “Save”. The Dynamic Mapping settings will be maintained, but you will need to reenter the Threshold number desired. Once that is set, you will toggle the Real-Time Updates to activate the synch.

If you no longer possess that information, you can access as before by going to Setup, entering “External Client App Manager” in the search box and click on it. Click on “Edit Settings”, open the OAuth settings section and click on ‘Consumer Key and Secret’. A verification email will be sent to the account used. Once you enter the code, you will see the Consumer Key and Secret.

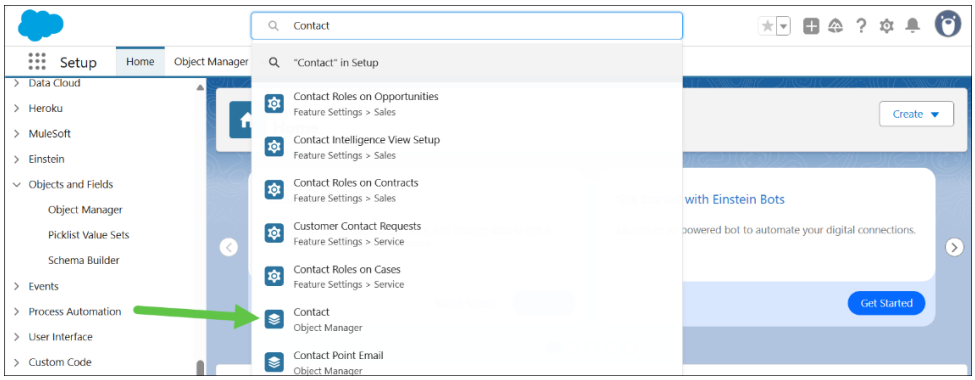
Optional Salesforce Help

ADDING SALESFORCE FIELDS

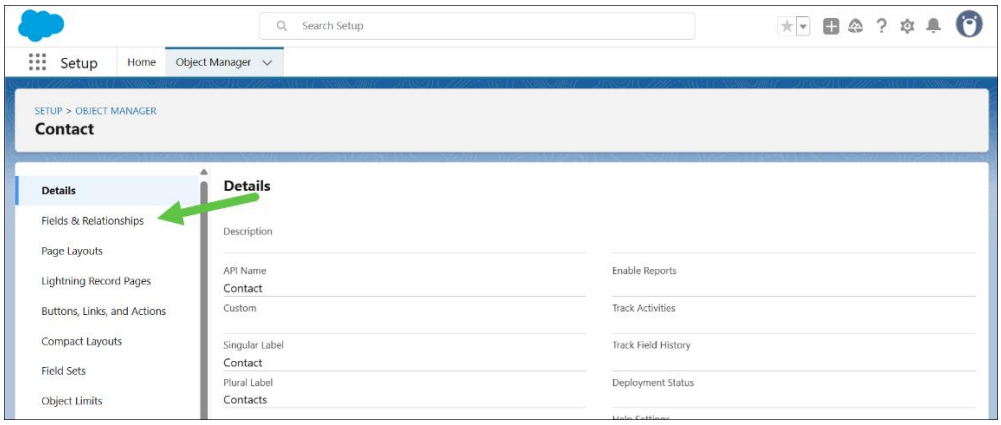
Go to Setup page by clicking on the icon in the header and clicking on Setup.



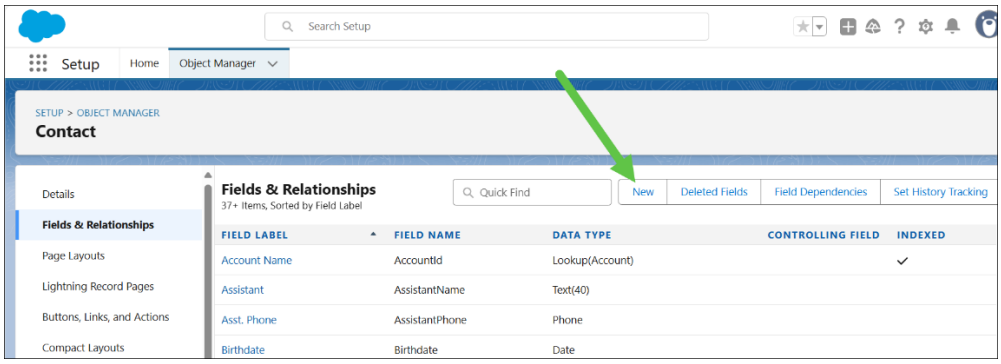
This will open the Salesforce Setup Page in a new browser tab.
In the quick find box, type “Contact” and click on the "Contact" (Object Manager) link.



Within the Contact object settings, click on the "Fields & Relationships" link.



Click the "New" button to start the process of creating a new custom field.



The rest of the steps are just:

Select Data Type: Salesforce will present various data types to choose from. Select the type that matches the data you plan to collect with the new field. For example, if it's a select one type of field, you select Picklist.
Then configure the field details:

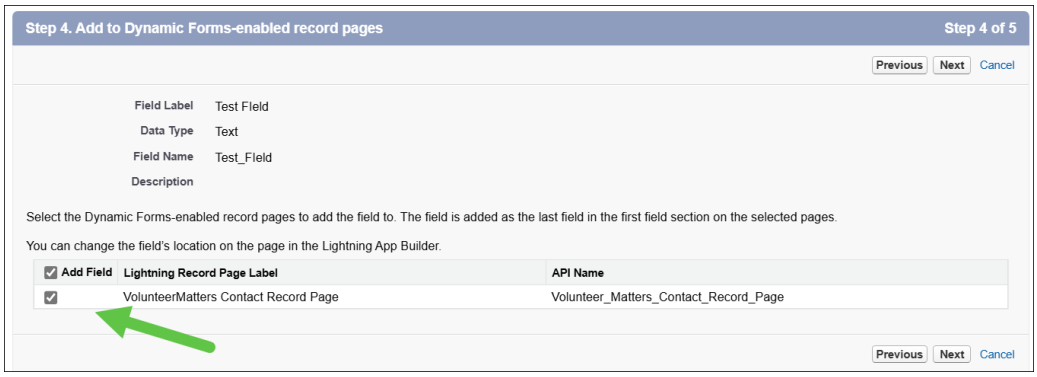
- **Field Label and Name**: Enter a Field Label (how the field will appear to users) and Field Name (API name used for backend operations).
- **Additional Properties**: Set other properties such as default values, help text, and field-level security. This ensures only the right profiles can view or edit the field.

Then configure the Field-Level Security

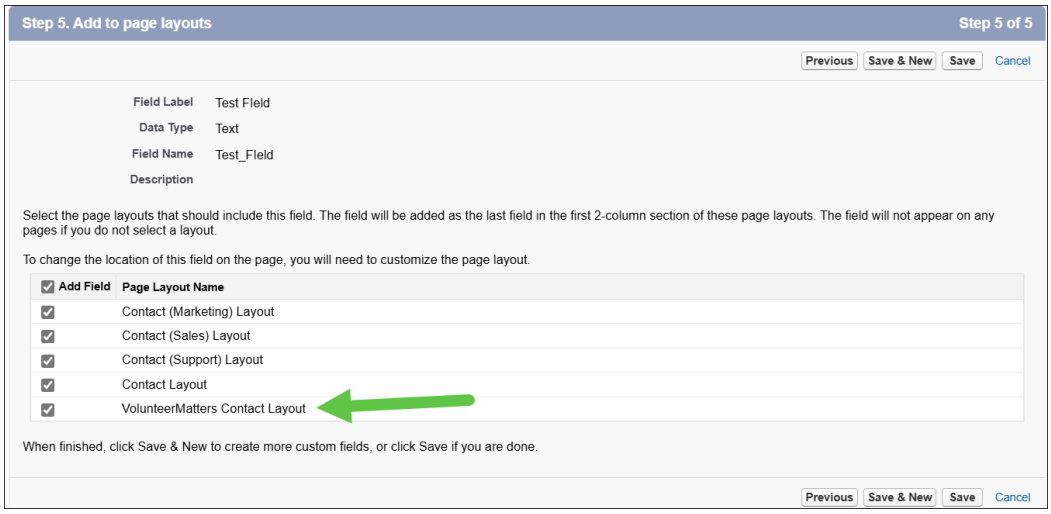
- **Profile Permissions**: Assign the visibility and edit permissions for different profiles. This step ensures that only those with appropriate access can view or modify this field.
- **Adjustment for Different Profiles**: You may choose to make the field visible to some profiles while hidden from others or editable by certain profiles.

I find it easiest to leave default settings. Just make sure “System Administrator” is checked under the visible column.

Then add to Dynamic Forms-enabled record pages – Leave “VolunteerMatters Contact Record Page” checked and click "Save."



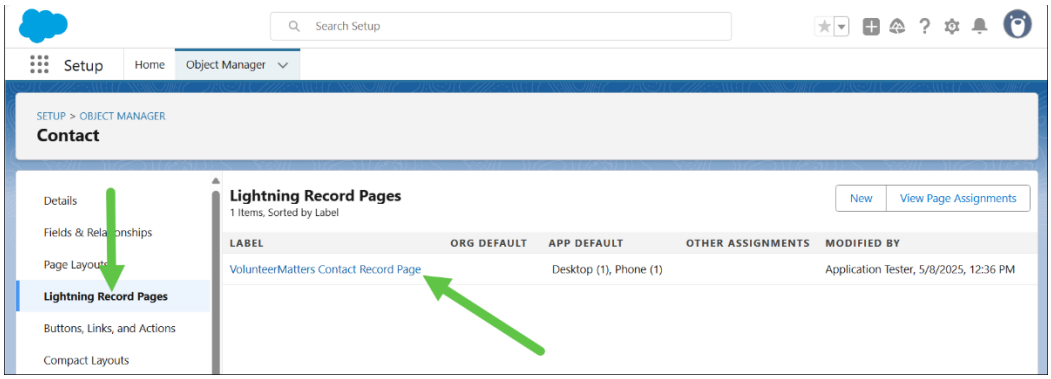
Then Select Page Layouts. Salesforce allows you to place the new field on the relevant Contact page layouts. Select the layouts where the field should be visible (making sure “VolunteerMatters Contact Layout” along with defaults) and click "Save."



This will return you to the main “Fields & Relationships” page.

Then you can verify placement by navigating to a Contact record to inspect the field’s placement and availability, ensuring it's displaying and functioning as intended.

If a field is not added correctly or to the right location on the contact record, you can edit the “VolunteerMatters Contact Record Page” under the Lightning Record Pages section of the Contact > Object Manager area. ***THIS SHOULD BE THE CLONED PAGE YOU CREATED ABOVE.***

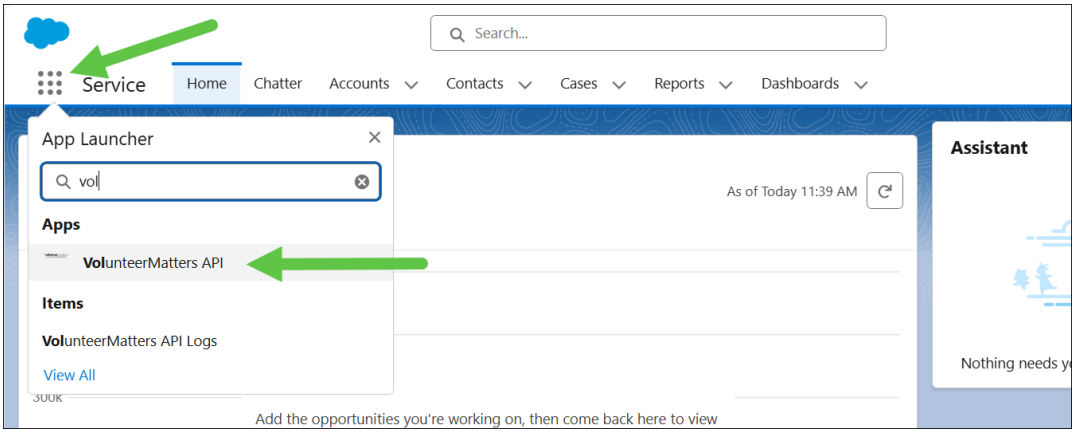


This will open the “Volunteer_Matters_Contact_Record_Page (Unlocked)” item, and you can click “Edit” and then you will be presented with Lightning App Builder where you can drag and drop fields and customize the contact page as desired.

Once the fields are added in Salesforce, mapped under the configuration area of the VolunteerMatters Salesforce API APP, and it is activated, the synch will begin. The

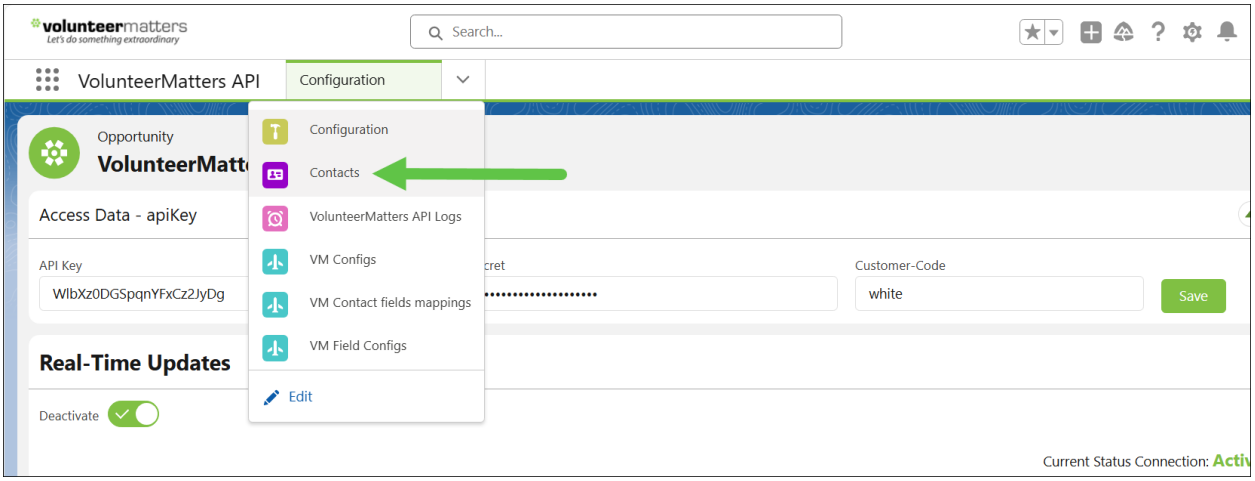
synch works both ways. Fields updated in VolunteerMatters will be updated in Salesforce and vice versa. Any new VolunteerMatters contacts or VolunteerMatters contacts that are updated where there is not a corresponding contact in salesforce, will have one created in Salesforce.

You can access the VolunteerMatters Contact list in salesforce after logging into Salesforce, from the dots in upper left corner type VolunteerMatters.



Then click on the VolunteerMatters API App.

The app will default to the last page you were on, so to access contacts, just click the down arrow and select contacts.



From here you can click into different contacts and verify information.