














# How to set up Tracky?

	Set up Object tracking using Tracking Console	1
	Manual tracking setup	1
	Dashboards setup	1
	Open object SLA Tracker	1
	Specifying an object to track	2
	Setting up an object under tracking	3
	Setting up an appropriate tab	5
	Creating the record	6
	Creating the flow	6
	Setting up flow logic	7
	Everything is done!	10

## Set up Object tracking using Tracking Console

1. Assign '**Tracky Managed Objects Access**' and '**Tracky Admin**' permission sets
2. Setup Tracky Authentication app:  
Setup → App Manager → TrackyAuthentication → Manage → Edit Policies → Set Permitted Users  
'Admin approved users are pre-authorized' → Assign System Administrator or custom profile to the app
3. In **App Launcher**, type in **Tracky**
4. In the **Select Object** search window select the object to track. This action automatically creates: trigger on selected object, lookup field on Tracky\_\_SLA\_Tracker\_\_c object.  
Permission is also created for the new lookup field in 'Tracky Dynamic Objects Access' permission set.
5. In the **Select Field** search window select the field to track. This action automatically creates a Tracky\_\_Field\_List\_\_c record.
6. Tracking of fields is possible only for Users who have the Tracky license.

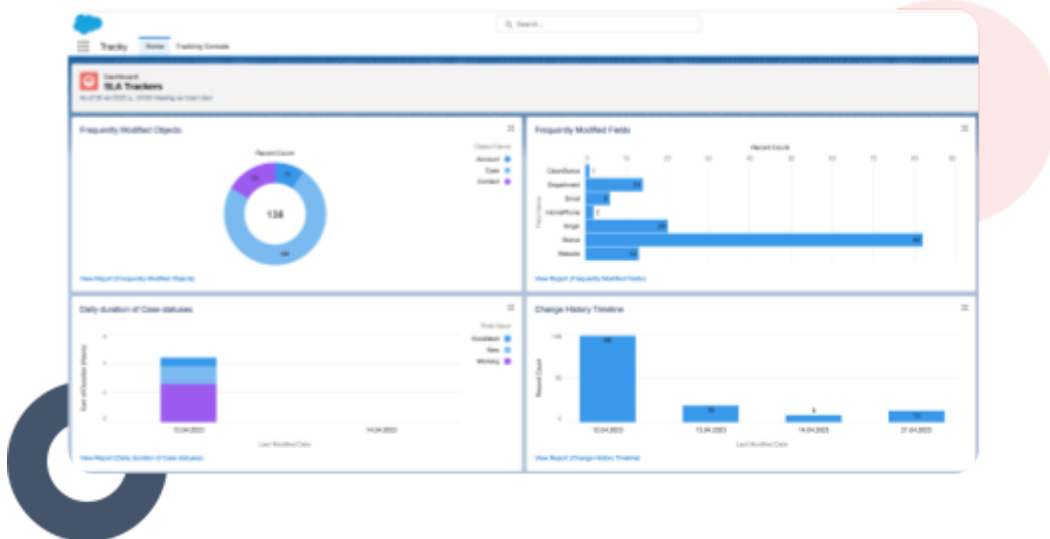
## Manual tracking setup

- In the current version field tracking is performed using automatically generated triggers. Since triggers must have at least 1% test coverage, tests are automatically generated for trigger deployment. When generating tests, the configuration of validation rules, triggers, and other rules present in the organization are not taken into account. If the automatically generated trigger cannot be deployed, use the autolaunched flow described below or deploy the trigger manually with a written test that takes into account the requirements of the rules. Example trigger:

```
trigger Tracky_AccountTrigger on Account (after insert, after update) {
    Boolean isPackageInstalled = Type.forName('Tracky__SLA_Tracker__c') != null;
    Boolean isTrackingAccessible = isPackageInstalled &&
    UserInfo.isCurrentUserLicensed('Tracky')
                                || !isPackageInstalled;
    if (isTrackingAccessible) {
        Tracky.TrackerHandler.triggerHandler();
    }
}
```

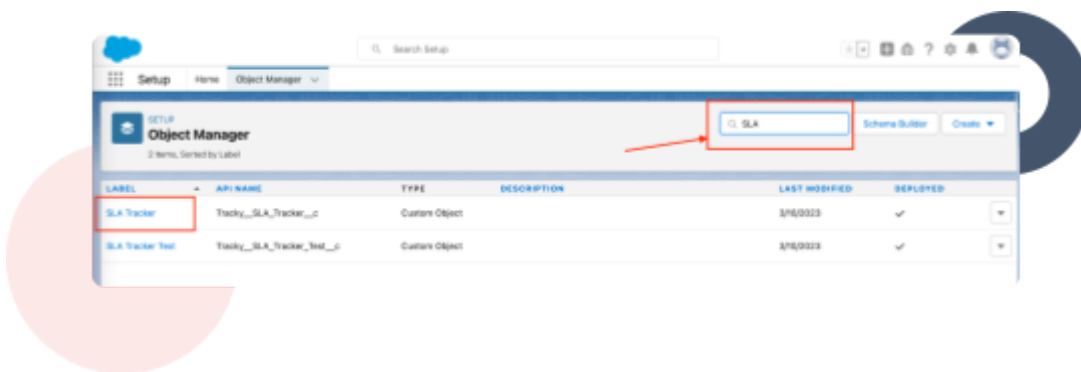
## 📊 Dashboards setup

Go to Setup > Lightning App Builder > TrackyHome > Set Tracky App as default.



## 🔗 Open object SLA Tracker

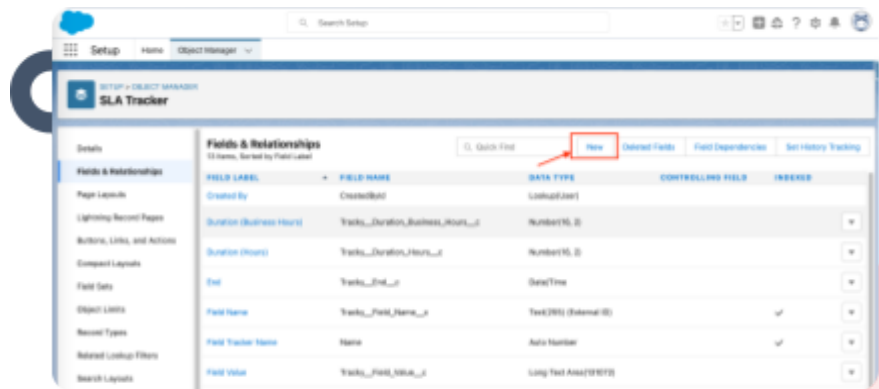
→ Open **Setup** by clicking the gear icon from the right upper corner. Click **Object Manager**, type “SLA” in the search bar type, and among the objects below select **SLA Tracker**



## Specifying an object to track

- Click **Fields & Relationships** and after that by clicking the “New” button create a lookup to needed object (limited to 39 lookup fields)

In our example, we will be tracking an Account object



## Setting up an object under tracking

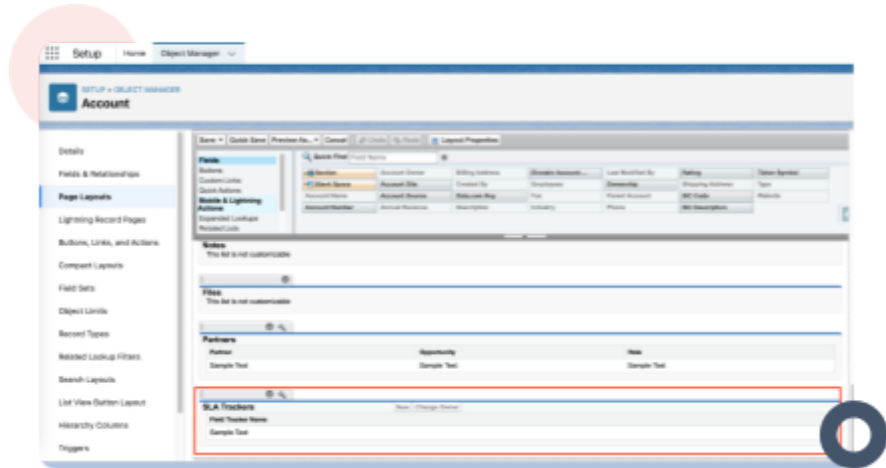
- After the lookup field is created, open the specified object by clicking **Object Manager** and type in **object name**.

- Click **Page Layouts** and open the only one page layout in the shown list

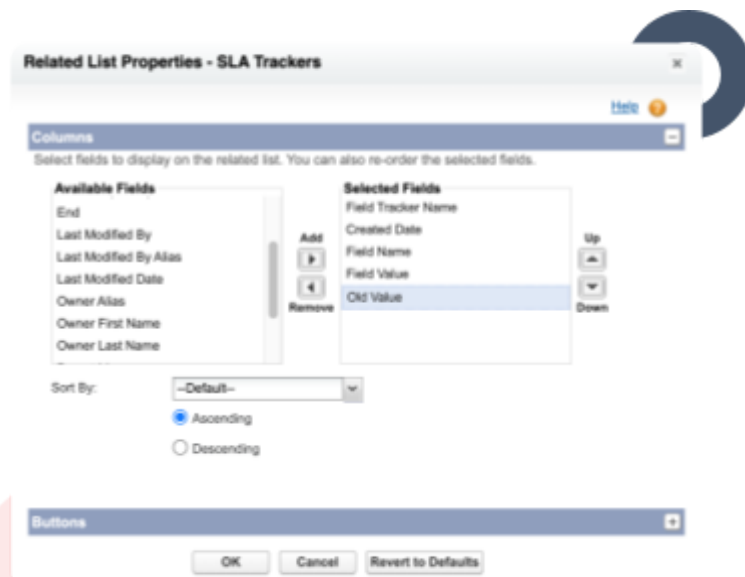


→ Scroll down to the very bottom to find a related list called **SLA Tracker**

If you cannot find the related list, find it in the list under the button “Save” in section Related Lists, and add to the page layout



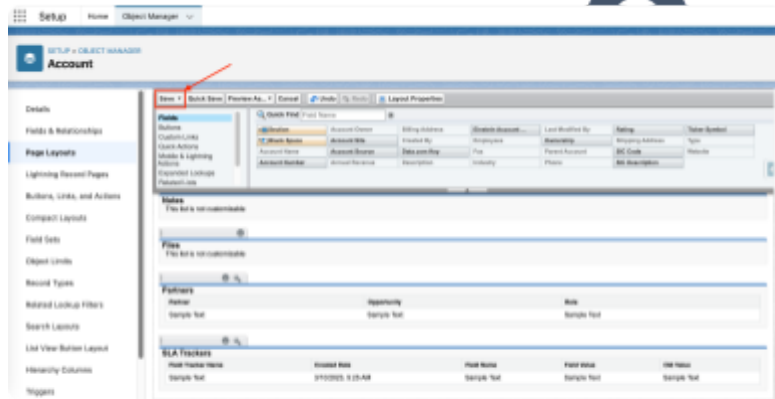
→ Add fields to display by clicking the **wrench** icon



→ Uncheck **New** and **Change Owner**



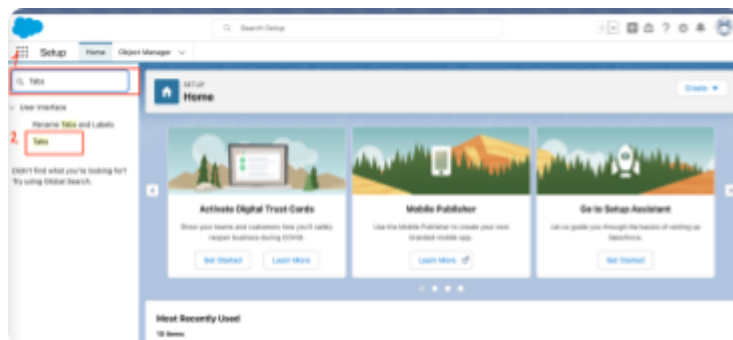
→ **Save** the page layout



## Setting up an appropriate tab

### 1. Using Salesforce standard components

→ From Setup, type in **“Tabs”**



→ Find **“Custom Object Tabs”**



→ After clicking **“New”** select the object **Field List** (Tab Style is optional)


→ Click **Next**, then **Save**



## 2. Using Tracky managed components

Go to Record Page > Setup > Edit page > Components > Custom-Managed > Add **“Tracking Related List”** to the Record Page > Save

Feed **Related**

 **Tracking Records (4)**

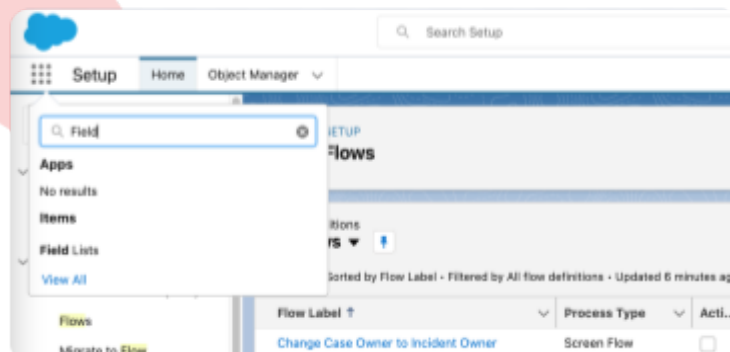
Page Size: 10

Field Tracker Name	Field Name	Old Value	Field Value	Duration (Business ...
SLA-000002	Status	Escalated	On Hold	0,01
SLA-000004	Status	Escalated	On Hold	
SLA-000005	Description2		<b>Sample Case</b>	
SLA-000003	Status	On Hold	Escalated	0,71

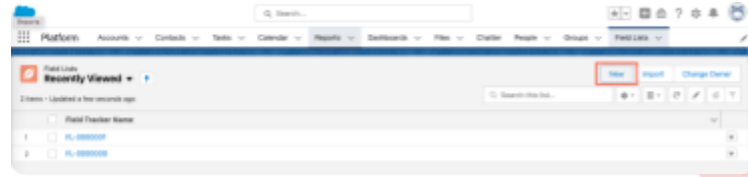
< Page: 1 of 1 >

## Creating the record

→ In **App Launcher**, type in **Field List**



→ Create a new record by clicking the **New** button

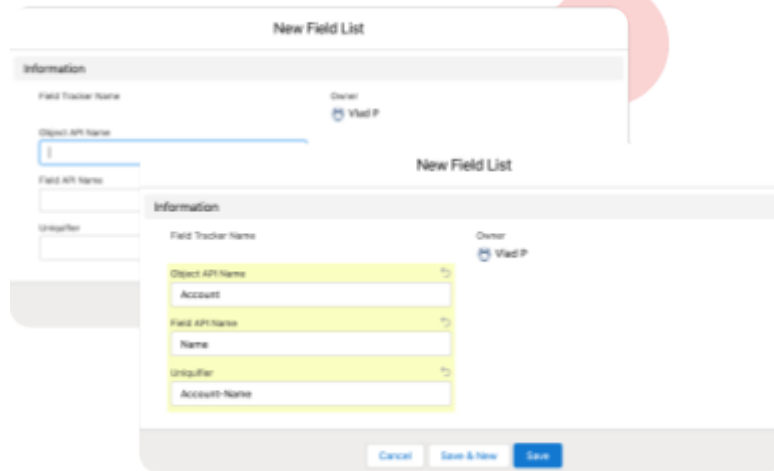


→ Fill in the following fields:

**Object API Name** - object to be tracked

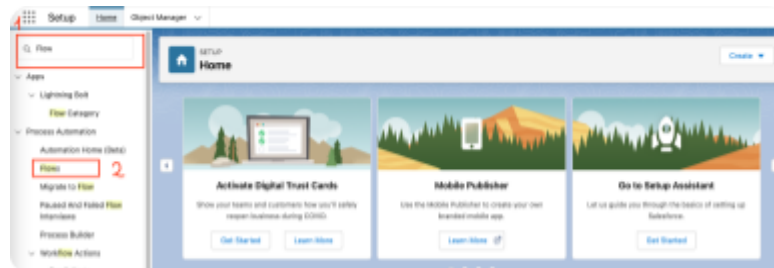
**Field API Name** - field in that object

Uniquifier should follow the template **“Object API Name - Field API Name”**

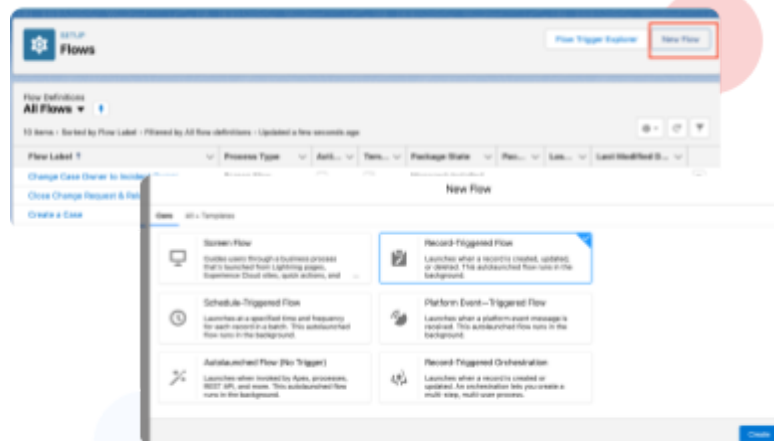


## » Creating the flow

→ In Setup, type in **Flows**



→ Create a new **Record-Triggered Flow**



- In **Object**, select the tracked object and in **“Trigger the Flow When”**, choose **A record is created or updated**



## → Setting up flow logic

1. **Create a formula** to distinguish create/update inside the flow

Formula function:  
ISNEW()



2. Add a decision element where your **IsCreated formula will be utilized**



3. Add **Apex Action** for “Created case” with following parameters for positive path.



- Start typing **Track...** and select the Action **Track Object Field Tracks**



- Enter **Label** and **API Name** (auto-populated depending on Label field).



- Under **Set Input Values**, objectId enter **!{\$Record.Id}** to pass the ID of the current record, for subj enter **!{\$Record}** to pass the current record.

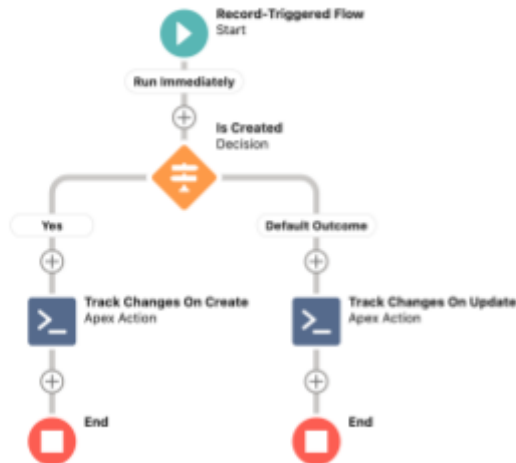
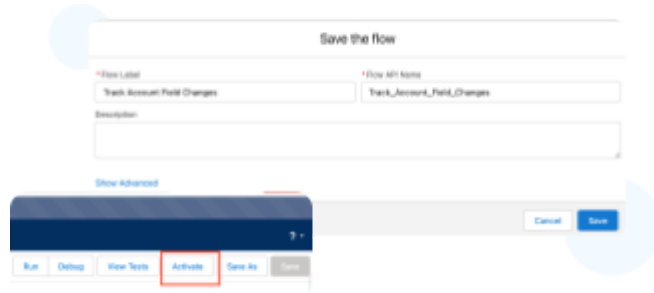
**4. Add Apex Action for “Updated case” with following parameters**

- Enter **Label** and **API Name** (auto-populated depending on Label field).



- Under **Set Input Values**, objectId enter **!{\$Record.Id}** to pass the ID of the current record, for subj enter **!{\$Record\_\_Prior}** to pass prior record values

→ **Save** and **Activate** the flow



## Everything is done!

→ Now Tracky is ready to be used. Every change in the specified objects and fields will be tracked.

