

---

# Workforce Engagement Developer Guide

Version 64.0, Summer '25

## Summer '25





# CONTENTS

- Chapter 1: Workforce Engagement Developer Guide . . . . . 1**
- Workforce Engagement Object Reference . . . . . 2
- Workforce Engagement Metadata and Tooling API Reference . . . . . 5
- Automate How Agents Accept and Decline Shift Assignments . . . . . 6
- Manage Training with REST and Apex APIs . . . . . 21
- Use Case Examples . . . . . 23
  - Automate Time Sheet Creation, Capture Omni Status Change, and Calculate Shrinkage . . . . . 23
  - Shifty the Bot - Manage Your Workforce Through SMS Bot Interactions . . . . . 27



# CHAPTER 1 Workforce Engagement Developer Guide

## In this chapter ...

- [Workforce Engagement Object Reference](#)
- [Workforce Engagement Metadata and Tooling API Reference](#)
- [Automate How Agents Accept and Decline Shift Assignments](#)
- [Manage Training with REST and Apex APIs](#)
- [Use Case Examples](#)

Get ready to fine-tune or extend your Workforce Engagement implementation. This guide introduces developers and technical admins to concepts and data in Workforce Engagement features: workload histories, intelligent forecasts, capacity plans, and shift creation and scheduling.

This guide includes:

- Reference information about Workforce Engagement objects
- REST and Apex code examples that show how to manage learning modules in Agent Engagement
- Sample use cases to automate time sheet creation or for using a bot to communicate agent schedules

If you're new to Workforce Engagement, we recommend checking out [Get Started with Workforce Engagement](#) in Trailhead.

Before using this guide, be sure to visit the [Workforce Engagement help topics](#).

## Workforce Engagement Object Reference

When you enable Workforce Engagement, you gain access to a suite of standard Salesforce objects.



**Note:** Workforce Engagement is scheduled for retirement. See [Workforce Engagement Retirement](#).

The following tables link to reference information for standard objects in Workforce Engagement. Some objects are specific to Workforce Engagement, while others are used in a variety of Salesforce features, such as Field Service and Omni-Channel.



**Note:** These tables don't include sharing, feed, or history objects.

## Workload Histories and Intelligent Forecasts

If you're using an Omni-Channel queue-based workflow, here's the objects that Workforce Engagement uses in workload histories and forecasts.

| Object Name                     | Definition   | Tab in Salesforce? |
|---------------------------------|--|--------------------|
| <a href="#">AgentWork</a>       | A work assignment that's been routed to an agent who's a queue member.   |                    |
| <a href="#">Group</a>           | A set of user records.   | ✓                  |
| <a href="#">QueueSubject</a>    | A mapping between a queue Group and the sObject types associated with the queue, including custom objects.                                     |                    |
| <a href="#">ServiceChannel</a>  | A channel used to route work items to agents associated with the queue.  |                    |
| <a href="#">ServiceResource</a> | An agent user who's added to the queue as a queue member.  | ✓                  |
| <a href="#">Workload</a>        | A time series of work volumes and average handle times. A Workload record contains the results of workload history aggregation or forecasting. |                    |
| <a href="#">WorkloadUnit</a>    | A number of work items and average handle times within a time interval.  |                    |

If you're not using an Omni-Channel queue-based workflow, you specify one or more of the standard channel objects or a custom object that has your channel data.

| Object Name                        | Definition                               | Tab in Salesforce?   |
|------------------------------------|--|----------------------|
| <a href="#">Case</a>               | A customer issue or problem.             | ✓                    |
| <a href="#">ConversationEntry</a>  | A message or an event in a chat history. |                      |
| <a href="#">LiveChatTranscript</a> | An interaction in a chat channel.        | ✓ (Chat Transcripts) |
| <a href="#">MessagingSession</a>   | A session in a messaging channel.        |                      |

| Object Name                     | Definition   | Tab in Salesforce? |
|---------------------------------|--|--------------------|
| <a href="#">ServiceResource</a> | An agent user who can be assigned to shifts.   | ✓                  |
| <a href="#">VoiceCall</a>       | A call in the voice channel, either for Service Cloud Voice or Sales Dialer.   |                    |
| <a href="#">Workload</a>        | A time series of work volumes and average handle times. A Workload record contains the results of workload history aggregation or forecasting. |                    |
| <a href="#">WorkloadUnit</a>    | A number of work items and average handle times within a time interval.  |                    |

## Capacity Plans

| Object Name                          | Definition   | Tab in Salesforce? |
|--------------------------------------|--|--------------------|
| <a href="#">JobProfile</a>           | A category that describes the expertise that's needed for the work.  | ✓                  |
| <a href="#">JobProfileQueueGroup</a> | A queue and job profile mapping that includes characteristics about the work to be done. This object is used only in a queue-based workflow. |                    |
| <a href="#">OperatingHours</a>       | Hours that you can define for your business and your workers.  | ✓                  |
| <a href="#">ServiceChannel</a>       | The channel used to route work items to agents.  |                    |
| <a href="#">ServiceResource</a>      | An agent user and queue member who can receive work assignments.   | ✓                  |
| <a href="#">ServiceTerritory</a>     | A location in which work is performed.   | ✓                  |
| <a href="#">Shift</a>                | A record used to schedule service resources.   | ✓                  |
| <a href="#">ShiftSegment</a>         | A scheduled activity within a shift.   |                    |
| <a href="#">ShiftSegmentType</a>     | A type of scheduled activity within a shift.   |                    |
| <a href="#">ShiftTemplate</a>        | A template used to define commonly used shifts.  | ✓                  |
| <a href="#">TimeSlot</a>             | A period of time on a specified day of the week when work can be performed.  |                    |
| <a href="#">WorkDemographic</a>      | Descriptions of channel-region-skill-custom slices in a forecast or capacity plan.   |                    |

| Object Name                           | Definition  | Tab in Salesforce? |
|---------------------------------------|---|--------------------|
| <a href="#">WorkforceCapacity</a>     | A time series for actual or forecasted workforce capacity.                            |                    |
| <a href="#">WorkforceCapacityUnit</a> | The number of resources allocated or predicted for work items within a time interval. |                    |

## Shift Creation and Scheduling

| Object Name                                | Definition   | Tab in Salesforce?       |
|--|--|--------------------------|
| <a href="#">JobProfile</a>                 | A category that describes the expertise that's needed for the work.                                    | ✓                        |
| <a href="#">OperatingHours</a>             | Hours that you can define for your business and your workers.  | ✓                        |
| <a href="#">ServiceResource</a>            | An agent user who can be assigned to shifts.   | ✓                        |
| <a href="#">ServiceResourcePreference</a>  | An agent's scheduling preference that can be considered by the scheduling logic.                       | ✓                        |
| <a href="#">ServiceTerritory</a>           | A location in which work is performed.   | ✓                        |
| <a href="#">ServiceTerritoryMember</a>     | An agent who works in the associated territory.  |                          |
| <a href="#">SchedulingAdherenceDetail</a>  | A breakdown of shift adherence data by agent status.   |                          |
| <a href="#">SchedulingAdherenceSummary</a> | Shift adherence data for a service resource in a service territory and job profile on a specific date. | ✓ (Historical Adherence) |
| <a href="#">SchedulingConstraint</a>       | A limit on when or how work is performed.  | ✓                        |
| <a href="#">SchedulingObjective</a>        | A business goal that acts as a guideline for scheduling.   |                          |
| <a href="#">SchedulingRule</a>             | A hard limit that restricts which agents are candidates for shifts.                                    |                          |
| <a href="#">ServicePresenceStatus</a>      | A presence status that can be assigned to a service channel in Omni-Channel.                           |                          |
| <a href="#">Shift</a>                      | A record used to schedule service resources.   | ✓                        |
| <a href="#">ShiftSegment</a>               | An activity that's scheduled during a shift.   |                          |
| <a href="#">ShiftSegment Type</a>          | A type of activity that's scheduled during a shift.  |                          |



| Object Name                         | Definition  | Tab in Salesforce? |
|-------------------------------------|---|--------------------|
| <a href="#">ShiftTemplate</a>       | A template used to define commonly used shifts.                             | ✓                  |
| <a href="#">Skill</a>               | A certification or area of expertise.                                       |                    |
| <a href="#">SkillRequirement</a>    | A skill that is required to complete a particular task.                     |                    |
| <a href="#">TimeSlot</a>            | A period of time on a specified day of the week when work can be performed. |                    |
| <a href="#">UserServicePresence</a> | A user's real-time presence status in Omni-Channel.                         |                    |

## Agent Empowerment and Engagement

| Object Name                          | Definition  | Tab in Salesforce? |
|--------------------------------------|---|--------------------|
| <a href="#">SkillLevelDefinition</a> | A skill that can be acquired by taking a learning module.         |                    |
| <a href="#">SkillLevelProgress</a>   | Training progress for a given user.                               |                    |
| <a href="#">PersonTraining</a>       | A learning module assignment.                                     |                    |
| <a href="#">ResourceAbsence</a>      | A time period in which a service resource is unavailable to work. | ✓                  |
| <a href="#">TimeSheet</a>            | A schedule of a service resource's time.                          | ✓                  |

### SEE ALSO:

[Developer Guide: Introducing SOAP API](#)

[Salesforce Help: Plan Data in an Omni-Channel Queue-Based Workflow](#)

[Salesforce Help: Plan Data in a Non-Omni Workflow](#)

## Workforce Engagement Metadata and Tooling API Reference

When you enable Workforce Engagement, you gain access to metadata components and tooling objects.



**Note:** Workforce Engagement is scheduled for retirement. See [Workforce Engagement Retirement](#).

| Metadata Component                          | Description  |
|---|--|
| <a href="#">WorkforceEngagementSettings</a> | Settings for Workforce Engagement. For example, enable Workforce Engagement and the Workforce Engagement Configuration app, opt to use Machine Learning-based forecasting, and turn on |

| Metadata Component                  | Description  |
|-------------------------------------|--|
|                                     | features such as historical adherence, intraday management, and real-time adherence.   |
| <a href="#">SchedulingObjective</a> | Represents a scheduling objective in Workforce Engagement. Scheduling objectives are business goals that are considered when finding agents for shift assignments. |
| <a href="#">SchedulingRule</a>      | Represents a scheduling rule in Workforce Engagement. Scheduling rules determine which agents are assigned to shifts.  |

| Tooling Objects                     | Description  |
|-------------------------------------|--|
| <a href="#">SchedulingObjective</a> | Represents scheduling objective settings for Workforce Engagement. |
| <a href="#">SchedulingRule</a>      | Represents scheduling rule settings for Workforce Engagement.      |

## Automate How Agents Accept and Decline Shift Assignments

Create an approval process and link it to a flow so that agents can acknowledge shift assignments. When a planner tentatively assigns a shift, the agent can accept or reject the shift assignment in Agent Home.

Set up an approval process for Shift records that's triggered by a flow. When a planner assigns or creates a shift, the flow starts an approval process that lets the agent approve or decline the shift assignment.

After you create and activate them, the flow triggers the approval process when these conditions are true.

- The shift status is Tentative.
- The shift is assigned to an agent's service resource.
- The owner of the shift record is the assigned service resource.

The flow also triggers the approval process when planners create and assign themselves to a shift.

You can extend the approval process and flow examples provided here, or use them as is. For example, if you customize shift status values, you can use a custom value instead of Tentative. If you want to turn off the ability for agents to approve shifts, be sure to deactivate both the approval process and flow.

Because there are many steps, we break them into sections.

1. [Create an Approval Process for Shift Records](#) on page 7
2. [Create a Flow to Trigger the Approval Process](#) on page 12
3. [Test Your Approval Process and Flow](#) on page 20

### EDITIONS

Workforce Engagement is available in Lightning Experience

Available in: **Enterprise**, **Performance**, and **Unlimited** Editions

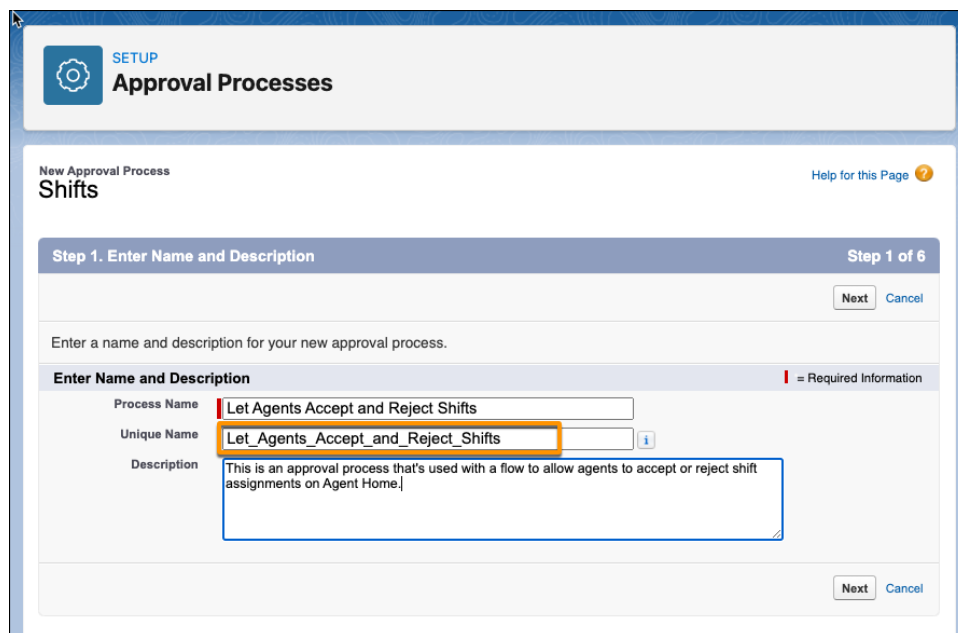
## Create an Approval Process for Shift Records

An approval process automates how records are approved in Salesforce. When you create the flow, you specify the approval process as the action that the flow takes.

From Setup, in the Quick Find box, enter *Approval Processes*, and select **Approval Processes**. Under Manage Approval Process For, select **Shift** as the record type. Under Create New Approval Process, select **Use Standard Setup Wizard**. The wizard walks you through steps to create the approval process.

## Create an Approval Process with the Standard Setup Wizard

1. For Step 1: Enter Name and Description, enter a name, a unique name, and an optional description for the approval process. You specify the unique name, such as *Let\_Agents\_Accept\_and\_Reject\_Shifts*, when you create your flow, so remember it. Click **Next**.



The screenshot shows the 'New Approval Process' wizard in Salesforce. The page title is 'Approval Processes' under the 'SETUP' tab. The sub-header is 'New Approval Process Shifts'. The wizard is at 'Step 1 of 6: Enter Name and Description'. The instructions say 'Enter a name and description for your new approval process.' The form has three fields: 'Process Name' with the value 'Let Agents Accept and Reject Shifts', 'Unique Name' with the value 'Let\_Agents\_Accept\_and\_Reject\_Shifts' (highlighted with an orange box), and 'Description' with the text 'This is an approval process that's used with a flow to allow agents to accept or reject shift assignments on Agent Home.' There are 'Next' and 'Cancel' buttons at the bottom right.

2. Click **Next** to skip Step 2: Specify Entry Criteria.
3. For Step 3: Specify Approver Field and Record Editability Properties, select **Administrators OR the currently assigned approver can edit records during the approval process**. Click **Next**.

**Step 3. Specify Approver Field and Record Editability Properties** Step 3 of 6

Previous Next Cancel

When you define approval steps, you can assign approval requests to different users. One of your options is to use a user field to automatically route these requests. If you want to use this option for any of your approval steps, select a field from the picklist below. Also, when a record is in the approval process, it will always be locked-- only an administrator will be able to edit it. However, you may choose to also allow the currently assigned approver to edit the record.

**Select Field Used for Automated Approval Routing**

Next Automated Approver Determined By: --None-- ⓘ

Use Approver Field of Shift Owner: ☐

**Record Editability Properties**

☐ Administrators **ONLY** can edit records during the approval process.

☒ Administrators **OR** the currently assigned approver can edit records during the approval process.

Previous Next Cancel

4. Click **Next** to skip Step 4: Select Notification Templates.
5. For Step 5: Select Fields to Display on Approval Page Layout, make these changes.
  - a. Add Shift Number, Owner Name, and Status to the selected fields.
  - b. Under Approval Page Fields, select **Display approval history information in addition to the fields selected above**.
  - c. Under Security Settings, select **Allow approvers to access the approval page from within the Salesforce application, or externally from a wireless-enabled mobile device**.
  - d. Click **Next**.

**Step 5. Select Fields to Display on Approval Page Layout** Step 5 of 6

Previous Next Cancel

The approval page is where an approver will actually approve or reject a request. Using the options below, choose the fields to display on this page.

**Available Fields**

- Background Color
- Created By
- Created Date
- End Time
- Job Profile
- Label
- Last Modified By
- Last Modified Date
- Non-Standard
- Service Resource
- Service Territory
- Shift Template
- Start Time
- Status Category

Add

Remove

**Selected Fields**

- Shift Number
- Owner Name
- Status

Up

Down

[Click here to view an example](#)

**Approval Page Fields**

☒ Display approval history information in addition to the fields selected above.

**Security Settings**

☐ Allow approvers to access the approval page only from within the Salesforce application. (Recommended)

☒ Allow approvers to access the approval page from within the Salesforce application, or externally from a wireless-enabled mobile device.

Previous Next Cancel

6. For Step 6: Specify Initial Submitters, make these changes.
  - a. Under Submitter Type, in Search, select Creator.
  - b. Add Record Creator and Shift Owner as allowed submitters. Record Creator allows planners who create shifts to approve or reject shifts that are assigned to them.
  - c. Select **Add the Submit for Approval button and Approval History related list to all Shift page layouts**.

- d. Save your approval process.

**Step 6. Specify Initial Submitters** Step 6 of 6

Using the options below, specify which users are allowed to submit the initial request for approval. For example, expense reports should normally be submitted for approval only by their owners.

**Initial Submitters**

Submitter Type: Search: **Creator** for: Find

**Available Submitters**

- None--

**Allowed Submitters**

- Shift Owner
- Record Creator

Add Remove

**Page Layout Settings**

☒ Add the Submit for Approval button and Approval History related list to all Shift page layouts

**Submission Settings**

☐ Allow submitters to recall approval requests

## Create an Approval Step

Create an approval step in your process. Select **Yes, I'd like to create an approval step now**, and click **Go**.

- For Step 1: Enter Name and Description, enter a name, a unique name, a step number, and an optional description for the approval step. Click **Next**.

**New Approval Step** Help for this Page

**Step 1. Enter Name and Description** Step 1 of 3

Enter a name, description, and step number for your new approval step.

**Enter Name and Description** Required Information

Approval Process Name: Let Agents Accept and Reject Shifts

Name: **Assign Shift Step**

Unique Name: **Assign\_Shift\_Step**

Description: This step allows an agent to approve or reject the assigned shift.

Step Number: **1**

- For Step 2: Specify Step Criteria, select **All records should enter this step**, and click **Next**.

**Step 2. Specify Step Criteria** Step 2 of 3

Specify whether a record must meet certain criteria before entering this approval step. If these criteria are not met, the approval process can skip to the next step, if one exists. [Learn more](#)

**Specify Step Criteria**

☒ All records should enter this step.

☐ Enter this step if the following criteria are met, else approve record

- For Step 3: Select Assigned Approver, under Select Approver, select **Automatically assign to approver(s)**. For the approver, select **Related user** and **Owner Name**. Save your work.

- You created an approval step, but you can skip specifying workflow actions right now. Select **No, I will do this later. Take me to the approval process detail page to review what I've just created** and click **Go**.

## Create an Approval Action

Create an approval action that updates the shift status to Published when the agent approves their shift assignment in Agent Home.

- Click the name of your approval process to open it.
- In the Approval Steps section, click **Show Actions**.
- Under Approval Actions, click **Add New** and select **Field Update**.

- Under Identification, enter a name, unique name, and optional description. For example, name the action *Update Status to Published*. For the Shift object, select **Status** as the Field to Update.
- Under Specify New Field Value and Picklist Options, select **A specific value**. Select **Published** as that value.
- Save your approval action. You can skip defining a rejection action. If the agent rejects the shift assignment, the shift status remains set to Tentative.

7. In the **Final Approval Actions** section of your approval process, click **Edit** for the Record Lock type. Select **Unlock the record for editing**.

Unlocking the Shift record is necessary so that the approval process can update shift status. Otherwise an error occurs when the agent tries to approve the shift assignment.



The screenshot shows a web interface titled "Let Agents Accept and Reject Shifts" with a sub-header "Final Approval Record Lock". In the top right corner, there is a link "Help for this Page" with a question mark icon. The main content area is divided into two sections. The top section contains "Save" and "Cancel" buttons. The bottom section, titled "Choose Final Approval Record Locking Behavior", contains two radio button options: "Lock the record from being edited" and "Unlock the record for editing". The "Unlock the record for editing" option is selected, indicated by a blue dot. At the bottom of this section, there are also "Save" and "Cancel" buttons.

8. Save your settings.
9. Review your approval process. If it looks OK, activate it.

Approval Processes  
**Shift: Let Agents Accept and Reject Shifts** [Help for this Page](#)

[« Back to Approval Process List](#)

**Process Definition Detail** [Edit](#) [Clone](#) [Delete](#) [Activate](#)

|                                    |  |  |                               |
|------------------------------------|--|--|-------------------------------|
| Process Name                       | Let Agents Accept and Reject Shifts  | Active                                       | <input type="checkbox"/>      |
| Unique Name                        | Let_Agents_Accept_and_Reject_Shifts  | Next Automated Approver Determined By        |                               |
| Description                        | This is an approval process that's used with a flow to allow agents to accept or reject shift assignments on Agent Home. |  |                               |
| Entry Criteria                     |  |  |                               |
| Record Editability                 | Administrator OR Current Approver  | Allow Submitters to Recall Approval Requests | <input type="checkbox"/>      |
| Approval Assignment Email Template |  |  |                               |
| Initial Submitters                 | Shift Owner, Record Creator  |  |                               |
| Created By                         | Admin User, 4/9/2022, 3:11 PM  | Modified By                                  | Admin User, 4/9/2022, 4:22 PM |

**Initial Submission Actions** [Add Existing](#) [Add New](#)

| Action      | Type | Description                       |
|-------------|------|-----------------------------------|
| Record Lock |      | Lock the record from being edited |

**Approval Steps** [New Approval Step](#)

| Action  | Step Number | Name              | Description  | Criteria | Assigned Approver                        | Reject Behavior |
|---|-------------|-------------------|--|----------|--|-----------------|
| <a href="#">Show Actions</a>   <a href="#">Edit</a>   <a href="#">Del</a> | 1           | Assign Shift Step | This step allows an agent to approve or reject the assigned shift. |          | <a href="#">Related User: Owner Name</a> | Final Rejection |

**Final Approval Actions** [Add Existing](#) [Add New](#)

| Action               | Type        | Description                   |
|----------------------|-------------|-------------------------------|
| <a href="#">Edit</a> | Record Lock | Unlock the record for editing |

**Final Rejection Actions** [Add Existing](#) [Add New](#)

| Action               | Type        | Description                   |
|----------------------|-------------|-------------------------------|
| <a href="#">Edit</a> | Record Lock | Unlock the record for editing |

**Recall Actions** [Add Existing](#) [Add New](#)

| Action | Type        | Description                   |
|--------|-------------|-------------------------------|
|        | Record Lock | Unlock the record for editing |

[^ Back To Top](#) Always show me [more](#) records per related list

## Create a Flow to Trigger the Approval Process

To trigger the approval process, you define outcomes in your flow that check for these conditions when a Shift record is updated.

- The shift status is changed to Tentative.
- A service resource is assigned the shift.
- The owner of the shift record is changed to the assigned service resource.

If any two conditions are true, the flow starts when the third condition becomes true. A fourth outcome covers the scenario when a planner creates a shift and self-assigns it.

1. From Setup, in the Quick Find box, enter *Flows*, and select **Flows**.
2. Click **New Flow** to get started creating your flow. Select Record-Triggered Flow as the flow type and click **Create**.
3. Configure when the flow starts.
  - a. Select the Shift object as the object whose records trigger the flow.



- b. Select **A record is created or updated** as the trigger.
- c. Select **Action and Related Records**, and click **Done**.

**Configure Start**

---

**Select Object**  
Select the object whose records trigger the flow when they're created, updated, or deleted.

• Object  
Shift

---

**Configure Trigger**  
• Trigger the Flow When:

☐ A record is created  
☐ A record is updated  
☒ A record is created or updated  
☐ A record is deleted

---

**Set Entry Conditions**  
Specify entry conditions to reduce the number of records that trigger the flow and the number of times the flow is executed. Minimizing unnecessary flow executions helps to conserve your org's resources.

If you create a flow that's triggered when a record is updated, we recommend first defining entry conditions. Then select the **Only when a record is updated to meet the condition requirements** option for When to Run the Flow for Updated Records.

Condition Requirements  
None New Formulas for Conditions ⓘ

---

• Optimize the Flow for:

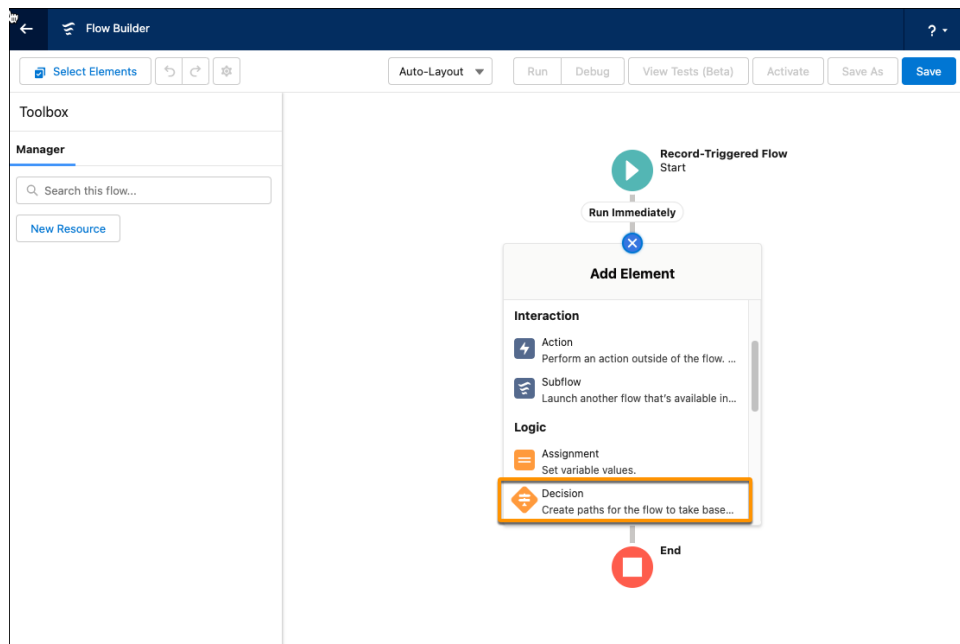
**Fast Field Updates**  
Update fields on the record that triggers the flow to run. This high-performance flow runs *before* the record is saved to the database.

**Actions and Related Records**  
Update any record and perform actions, like send an email. This more flexible flow runs *after* the record is saved to the database.

☐ Include a Run Asynchronously path to access an external system after the original transaction for the triggering record is successfully committed

Cancel Done

4. Click + on the Flow Builder and add a decision element to the flow.



5. Enter a label, API name, and optional description for your flow. For example, enter *Trigger Accept or Reject Shift Assignment* as the name.
6. Add the first outcome and the conditions that must be met. This outcome starts the flow when the shift status is changed to Tentative and the other conditions are met.
  - a. Enter a label and an outcome API name. For example, *Status is Changed to Tentative*.
  - b. Select **All Conditions are Met(AND)**.
  - c. Specify resource, operator, and value fields to define each condition. For example, set `{!$Record.Status}` as the resource, *is Changed* as the operator, and `{!$GlobalConstant.True}` as the value. After it's defined, the condition appears as `$Record > Status Is Changed True`. Set these conditions for the first outcome.
    - `{!$Record.Status} is Changed {!$GlobalConstant.True}`
    - `{!$Record.Status} Equals Tentative`
    - `{!$Record.OwnerId} Equals {!$Record.ServiceResource.RelatedRecordId}`

7. Next to Outcome Order, click **+** to add the second outcome and the conditions that must be met. This outcome starts the flow when the shift owner is changed to the agent and the other conditions are met.
  - a. Enter a label and an outcome API name. For example, *Owner is Changed to Agent*.
  - b. Select **All Conditions are Met(AND)**.
  - c. Set these conditions.
    - `{!$Record.OwnerId} is Changed {!$GlobalConstant.True}`
    - `{!$Record.OwnerId} Equals {!$Record.ServiceResource.RelatedRecordId}`
    - `{!$Record.Status} Equals Tentative`

**New Decision**

\* Label: Trigger Accept or Reject Shift Assignment  
 \* API Name: Trigger\_Accept\_or\_Reject\_Shift\_Assignment

Description: This flow launches an approval process when a shift record is created or updated. The flow checks that the shift status is Tentative, that it was assigned to a service resource, and that the service resource is the owner of the Shift record.

**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

**OUTCOME ORDER** +

- Status is Changed to Tentative
- Owner is Changed to Agent

**OUTCOME DETAILS** Delete Outcome

\* Label: Owner is Changed to Agent  
 \* Outcome API Name: Owner\_is\_Changed\_to\_Agent

Condition Requirements to Execute Outcome  
 All Conditions Are Met (AND)

| Resource                    | Operator   | Value   |
|-----------------------------|------------|---|
| A_3 \$Record > Owner ID     | Is Changed | True  |
| AND A_3 \$Record > Owner ID | Equals     | A_3 \$Record > Service Resource ID > User ID... |
| AND \$Record > Status       | Equals     | Tentative                                       |

+ Add Condition

**When to Execute Outcome**

☒ If the condition requirements are met  
☐ Only if the record that triggered the flow to run is updated to meet the condition requirements

Because you selected the Is Changed operator in a condition, you can't change when to execute the outcome.

Cancel Done

8. Next to Outcome Order, click + to add the third outcome. This outcome starts the flow when the shift is assigned to a service resource and the other conditions are met.
  - a. Enter a label and an outcome API name. For example, *Service Resource is Changed to Agent*.
  - b. Select **All Conditions are Met(AND)**.
  - c. Set these conditions.
    - `{!$Record.ServiceResourceId} is Changed { !$GlobalConstant.True }`
    - `{!$Record.OwnerId} Equals { !$Record.ServiceResource.RelatedRecordId }`
    - `{!$Record.Status} Equals Tentative`

**New Decision**

\* Label: Trigger Accept or Reject Shift Assignment

\* API Name: Trigger\_Accept\_or\_Reject\_Shift\_Assignment

Description: This flow launches an approval process when a shift record is created or updated. The flow checks that the shift status is Tentative, that it was assigned to a service resource, and that the service resource is the owner of the Shift record.

**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

**OUTCOME ORDER**

- Status is Changed to Tentative
- Owner is Changed to Agent
- Service Resource is Changed to Agent

**OUTCOME DETAILS**

\* Label: Service Resource is Changed to Agent

\* Outcome API Name: Service\_Resource\_Is\_Changed\_to\_Agent

Condition Requirements to Execute Outcome: All Conditions Are Met (AND)

| Resource                                   | Operator   | Value                                       |
|--|------------|---|
| \$Record > Service Resource ID > Resour... | Is Changed | True  |
| <b>AND</b>                                 |            |   |
| \$Record > Owner ID                        | Equals     | \$Record > Service Resource ID > User ID... |
| <b>AND</b>                                 |            |   |
| \$Record > Status                          | Equals     | Tentative                                   |

**When to Execute Outcome**

☒ If the condition requirements are met

☐ Only if the record that triggered the flow to run is updated to meet the condition requirements

**Warning:** Because you selected the Is Changed operator in a condition, you can't change when to execute the outcome.

Buttons: Cancel, Done

9. Next to Outcome Order, click **+** to add the fourth outcome. This outcome starts the flow when planners create and assign themselves a shift and the other conditions are met.
  - a. Enter a label and an outcome API name. For example, *Shift is Created and Assigned to Creator*.
  - b. Select **All Conditions are Met(AND)**.
  - c. Set these conditions.
    - `{!$Record.OwnerId} Equals {!$Record.CreatedBy.Id}`
    - `{!$Record.OwnerId} Equals {!$Record.ServiceResource.RelatedRecordId}`
    - `{!$Record.Status} Equals Tentative`
    - `{!$Record__Prior} is Null {!$GlobalConstant.True}`

**New Decision**

\* Label: Trigger Accept or Reject Shift Assignment      \* API Name: Trigger\_Accept\_or\_Reject\_Shift\_Assignment

Description: This flow launches an approval process when a shift record is created or updated. The flow checks that the shift status is Tentative, that it was assigned to a service resource, and that the service resource is the owner of the Shift record.

**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

**OUTCOME ORDER** +

- Status is Changed to Tentative
- Owner is Changed to Agent
- Service Resource is Changed to Agent
- Shift is Created and Assigned to Creator
- Default Outcome

**OUTCOME DETAILS** Delete Outcome

\* Label: Shift is Created and Assigned to Creator      \* Outcome API Name: Shift\_is\_Created\_and\_Assigned\_to\_Creator

Condition Requirements to Execute Outcome: All Conditions Are Met (AND)

| Resource                | Operator | Value                                       |
|-------------------------|----------|---|
| \$Record > Owner ID     | Equals   | \$Record > Created By ID                    |
| AND \$Record > Owner ID | Equals   | \$Record > Service Resource ID > User ID... |
| AND \$Record > Status   | Equals   | Tentative                                   |
| AND \$Record__Prior     | Is Null  | True  |

+ Add Condition

**When to Execute Outcome**

☒ If the condition requirements are met

☐ Only if the record that triggered the flow to run is updated to meet the condition requirements

Cancel Done

10. Under Outcome Order, click **Default Outcome**. Enter *No Action* as the label of the default outcome.

**New Decision**

\* Label: Trigger Accept or Reject Shift Assignment      \* API Name: Trigger\_Accept\_or\_Reject\_Shift\_Assignment

Description: This flow launches an approval process when a shift record is created or updated. The flow checks that the shift status is Tentative, that it was assigned to a service resource, and that the service resource is the owner of the Shift record.


**Outcomes** For each path the flow can take, create an outcome. For each outcome, specify the conditions that must be met for the flow to take that path.

**OUTCOME ORDER** +

- Status is Changed to Tentative
- Owner is Changed to Agent
- Service Resource is Changed to Agent
- Shift is Created and Assigned to Creator
- No Action

**OUTCOME DETAILS**

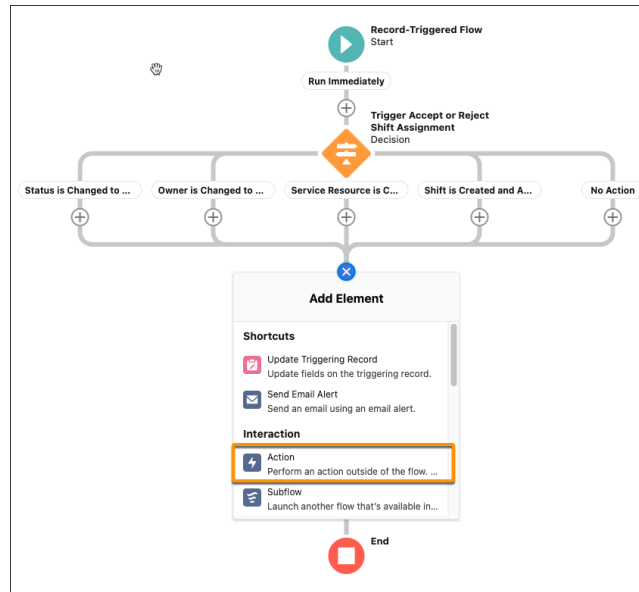
\* Label: No Action



**No conditions needed**  
If no outcome's conditions are met, the flow takes this path.

Cancel Done

11. In the Flow Builder, click + above the End element and add an action element.



- a. Select the **Submit for Approval** action. To make it easy to find this action, select **Approvals** as the filter category.
- b. Enter a label, API name, and optional description for the action. For example, enter *Accept or Reject Shift* as the name.
- c. Set the input value for the record ID to the shift record ID `{!$Record.Id}`.
- d. Turn on the toggle to Include the approval process name.
- e. Specify the exact API name of the approval process that you created earlier. For example, *Let\_Agents\_Accept\_and\_Reject\_Shifts*.
- f. Click **Done**.

**New Action**

Filter By  
Category ▼

Account  
Contact  
Event  
Lead  
Note  
Opportunity  
Messaging  
Notifications  
Email  
**Approvals**  
Uncategorized

Action  
Submit for Approval

Use values from earlier in the flow to set the inputs for the "Submit for Approval" core action. To use its outputs later in the flow, store them in variables.

\* Label  
Accept or Reject Shift

\* API Name  
Accept\_or\_Reject\_Shift

Description  
Submit the shift record for approval.

**Set Input Values**

A<sub>a</sub> \* Record ID  
{\$Record.Id}

A<sub>a</sub> Approval Process Name Or ID  
Let\_Agents\_Accept\_and\_Reject\_Shifts ☒ Include

A<sub>a</sub> Next Approver IDs ☐ Don't Include

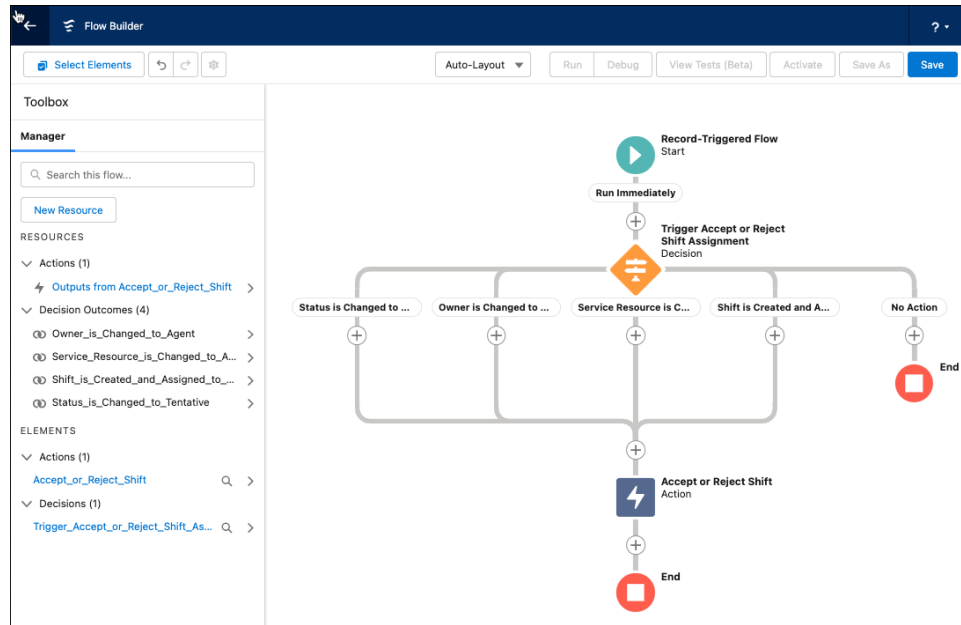
☐ Skip Entry Criteria ☐ Don't Include

Cancel Done

**12.** In the Flow Builder, link these outcomes to the Accept or Reject Shift action.

- Status is Changed to Tentative
- Owner is Changed to Agent
- Service Resource is Changed to Agent
- Shift is Created and Assigned to Creator

**13.** In the Flow Builder, link the No Action outcome to the End element.



14. Save and name your flow, for example, *Trigger Shift Approval*.

15. Activate your flow.

## Test Your Approval Process and Flow

Now you're ready to see the results of your work.

1. As a planner, create or select a shift. Set the shift status to Tentative.
2. Assign it to a service resource.
3. Change the shift owner to the service resource.
4. In Agent Home for that service resource, the agent can then accept or reject the shift.

If the agent approves the shift, its status changes to Published. If the agent rejects the shift, the shift's status remains as Tentative. When the agent refreshes Agent Home, the shift no longer appears on the agent's schedule.

If a planner assigns a shift to the wrong agent, the planner must terminate the approval process in the shift's Approval History related list. Otherwise, the agent can reject the erroneously assigned shift. The planner can then reassign the shift to the right agent.

SEE ALSO:

[Salesforce Help: Approvals](#)

[Salesforce Help: Flows](#)

[Salesforce Help: Email Notifications for Intraday Management](#)



## Manage Training with REST and Apex APIs

To manage learning module assignments programmatically, you can use REST or Apex APIs with record objects.

[Person Training](#) records represent learning assignments in Workforce Engagement. Assignments are Trailhead learning modules, which are [Learning Content](#) records.

### EDITIONS

Workforce Engagement is available in Lightning Experience

Available in: **Enterprise**, **Performance**, and **Unlimited** Editions

## REST API Examples

These REST API examples list, create, update, and delete learning assignments in PersonTraining records.

### List all learning assignments

```
/services/data/v54.0/query?q=SELECT+Id+from+PersonTraining
```

### Create a learning assignment

```
/services/data/v54.0/objects/PersonTraining
{
  "Name": "Person Training 001",
  "TrainingId": "028af5d6-7e23-3cea-66e5-fd8e3bfe7e9c",
  "TrainingType": "T"
}
```

### Update a learning assignment

In this example, 0hRSG00000000zJ2AQ is the ID of the Person Training record to update.

```
/services/data/v54.0/objects/PersonTraining/0hRSG00000000zJ2AQ
{
  "Name": "Person Training Module 003"
}
```

### Delete a learning assignment

In this example, 0hRSG00000000zJ2AQ is the ID of the Person Training record to delete.

```
/services/data/v54.0/objects/PersonTraining/0hRSG00000000zJ2AQ
```

## Apex API Examples

If you use Apex classes to look up learning modules programmatically, place a limit clause on the SOQL query, for example:

```
List<LearningContent> aa = [SELECT ExternalId FROM LearningContent
WHERE Title LIKE '%Accessibility%' LIMIT 5];
```

These example Apex classes create, search, update, delete, and route learning assignments.

### Create a learning assignment

```
/* InsertPersonTraining.apex */
PersonTraining Test001 = new PersonTraining();
Test001.Name = 'Test001';
Test001.TrainingId = '028af5d6-7e23-3cea-66e5-fd8e3bfe7e9c';
Test001.TrainingType = 'T';
```

```
insert Test001;
```

### Search for a learning assignment

```
/* SearchPersonTraining.apex */
PersonTraining training =
    [SELECT Name, ID FROM PersonTraining
     WHERE Name='Test001'
     LIMIT 1];
```

### Update a learning assignment

```
/* UpdatePersonTraining.apex */
PersonTraining training =
    [SELECT Name, Id FROM PersonTraining
     WHERE Name='newTest001'
     LIMIT 1];

// Update the training Name
training.Name = 'newTest001';

update training;

// Verify that the name was updated
PersonTraining training02 =
    [SELECT Name, Id FROM PersonTraining WHERE Id=:training.Id];
```

### Delete a learning assignment

```
PersonTraining training =
    [SELECT Name, Id FROM PersonTraining
     WHERE Name='newTest001'
     LIMIT 1];

delete training;
```

### Route a learning assignment

```
/* RoutePersonTraining.apex */
PendingServiceRouting routing = new PendingServiceRouting();
routing.CapacityWeight = 1;

// To route correctly, PreferredUserId must be the same as AssigneeId.
routing.preferredUserId = UserInfo.getUserId(); // Assign to current user for testing.
routing.isPreferredUserRequired = true;
routing.routingPriority = 1;
routing.routingType = 'SkillsBased';

// set service channel ID
routing.serviceChannelId = '0N9SG0000000CHJ'; // Hard-coded the service channel id
routing.isReadyForRouting = true;

// set workItemId to be person training ID
routing.workItemId = '0hRSG000000009h2AA';
```

```
insert routing;
```

## Use Case Examples

Extend your Workforce Engagement implementation. Learn how to automate time sheet creation or use a bot to communicate agent schedules.

You can build upon the included examples to support your business processes. If your processes differ, modify your setup to meet your needs.

The example use cases indicate the ease of implementation and an estimated time to implement. They assume familiarity with basic Salesforce admin skills.

Before you get started, set up [Workforce Engagement](#).

### [Automate Time Sheet Creation, Capture Omni Status Change, and Calculate Shrinkage](#)

Set up basic time sheet automation. Log agents' Omni-Channel status changes throughout the day and use that information to calculate the shrinkage percentage within a shift.

### [Shifty the Bot - Manage Your Workforce Through SMS Bot Interactions](#)

When your agents aren't at work, they don't always have access to their schedule in Salesforce. With SMS bots you can update a schedule without needing to access Salesforce.

SEE ALSO:

[Trailhead: Get Started with Workforce Engagement](#)

## Automate Time Sheet Creation, Capture Omni Status Change, and Calculate Shrinkage

Set up basic time sheet automation. Log agents' Omni-Channel status changes throughout the day and use that information to calculate the shrinkage percentage within a shift.

|                                    |            |
|------------------------------------|------------|
| <b>Ease of Implementation</b>      | Easy       |
| <b>Estimated Time to Implement</b> | 60 minutes |

## Prerequisites

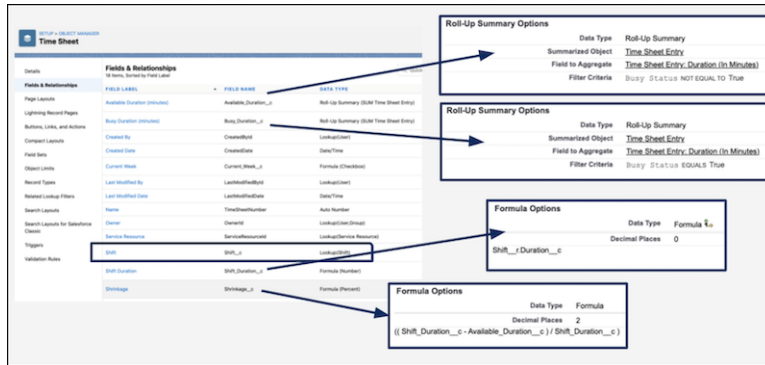
This example requires Omni Channel and Workforce Engagement to be turned on.

## Step 1: Create Custom Fields on the Time Sheet Object

Create custom fields on the Time Sheet object in Salesforce to correspond to the data that you use to calculate shrinkage.

1. From Setup, select **Object Manager**.
2. In the Quick Find box, enter *Time Sheet*, and then select the **Time Sheet** object.
3. Select **Fields & Relationships** from the left pane.

4. To create a field, select **New**.
5. To create a custom field to hold the data, follow the steps. To learn more about creating custom fields, see [Create Custom Fields](#).
6. Create these fields.



7. Make sure to add these fields to the layout for testing and deployment.

## Step 2: Create Custom Fields on the Time Sheet Entry Object

Create custom fields on the Time Sheet object in Salesforce to correspond to the data that you use to calculate shrinkage.

1. From Setup, at the top of the page, select **Object Manager**.
2. In the Quick Find box, enter *Time Sheet Entry*, and then select the **Time Sheet Entry** object.
3. Select **Fields & Relationships** from the left pane.
4. To create a field, select **New**.
5. To create a custom field to hold the data, follow the steps. To learn more about creating custom fields, see [Create Custom Fields](#).
6. Create these fields.

The screenshot shows the 'Overview' page for a bot configuration. It includes sections for Bot Information, Conversation Languages, and Connections.

**Bot Information**

| NAME    | API NAME | DESCRIPTION | LANGUAGE | LAST MODIFIED | LAST MODIFIED BY |
|---------|----------|-------------|----------|---------------|------------------|
| Cat bot | Cat_bot  |             | English  | Jul 27, 2021  |                  |

**Conversation Languages**

| Language          |
|-------------------|
| English (Primary) |

**Connections**

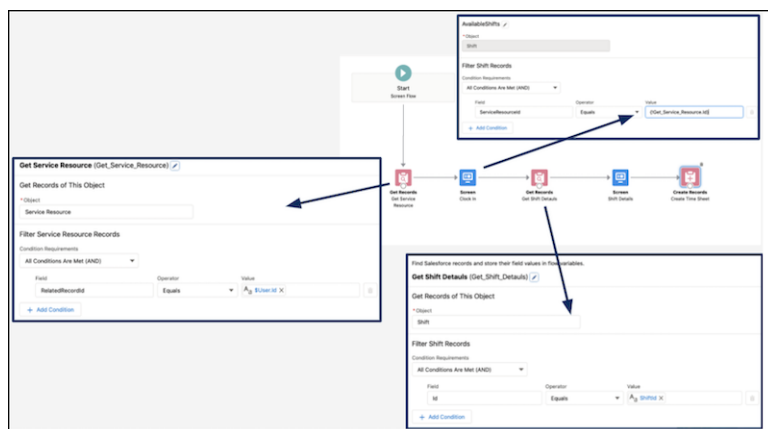
| Connection | Deployment   | Require Agent Online |
|------------|--------------|----------------------|
| Chat       | The Cat team | Off                  |

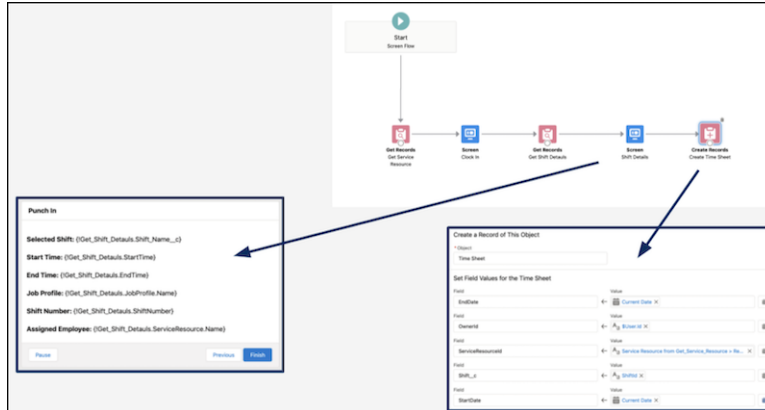
7. Make sure to add these fields to the layout for testing and deployment.

### Step 3: Introduce Automation Using Salesforce Flow

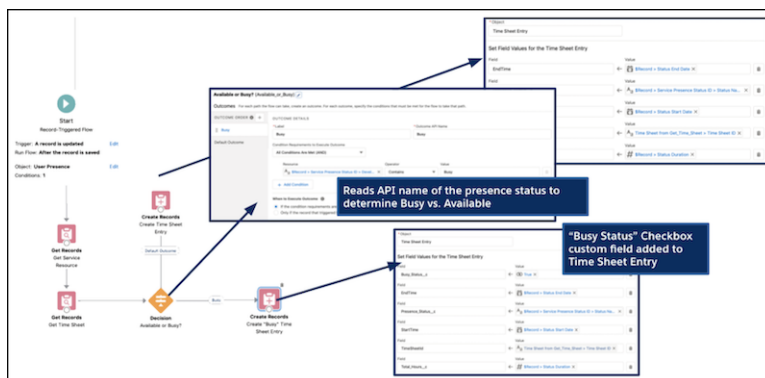
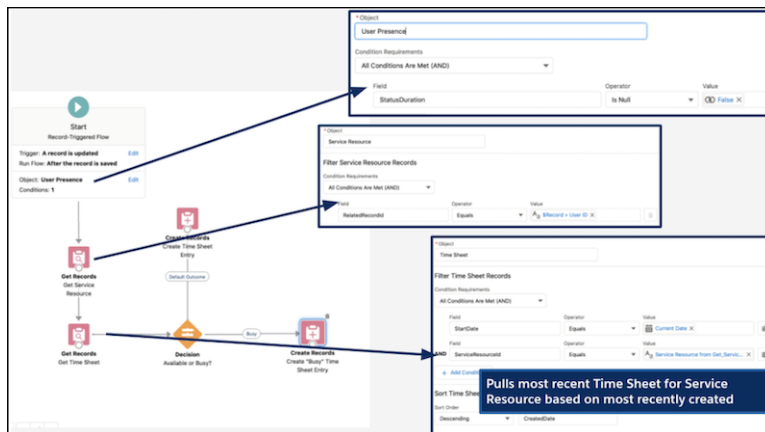
Two flows are used to allow agents to clock in, generate a Time Sheet, and capture Omni Channel status changes throughout the day.

1. From Setup, in the Quick Find box, enter *Flows*, and then select **Flows**.
2. To create your first flow, select **New**.
3. For more information, see: [Salesforce Flows](#)
4. Create a screen flow to place on the home page.





## 5. Create a record-triggered flow to capture an Omni status change.



## Test This Example

1. Clock in using the screen flow.

2. To simulate the changes in agent availability, change Omni statuses between different presences.
3. Verify that time sheet entries are created and shrinkage is calculated.

## Shifty the Bot - Manage Your Workforce Through SMS Bot Interactions

When your agents aren't at work, they don't always have access to their schedule in Salesforce. With SMS bots you can update a schedule without needing to access Salesforce.

|                                    |            |
|------------------------------------|------------|
| <b>Ease of Implementation</b>      | Easy       |
| <b>Estimated Time to Implement</b> | 15 minutes |

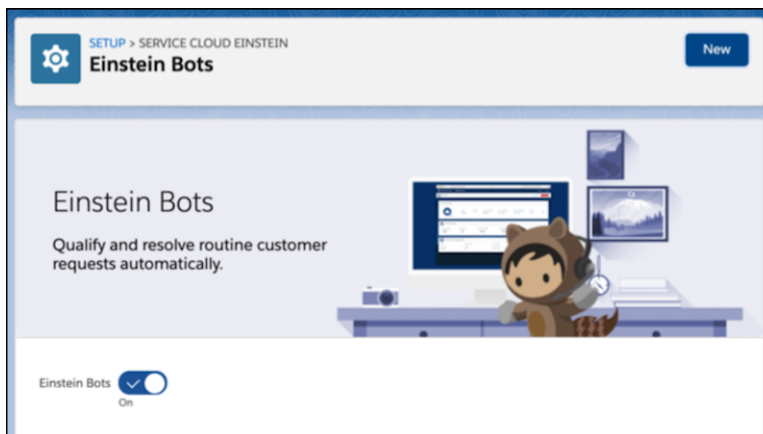
### Prerequisites

- Digital Engagement Licenses
- SMS Messaging Setup ([Get Started](#))
- User phone numbers must be in a User Record to be recognized by Shifty

### Step 1: Turn on Einstein Bots

To install the package, turn on bots.

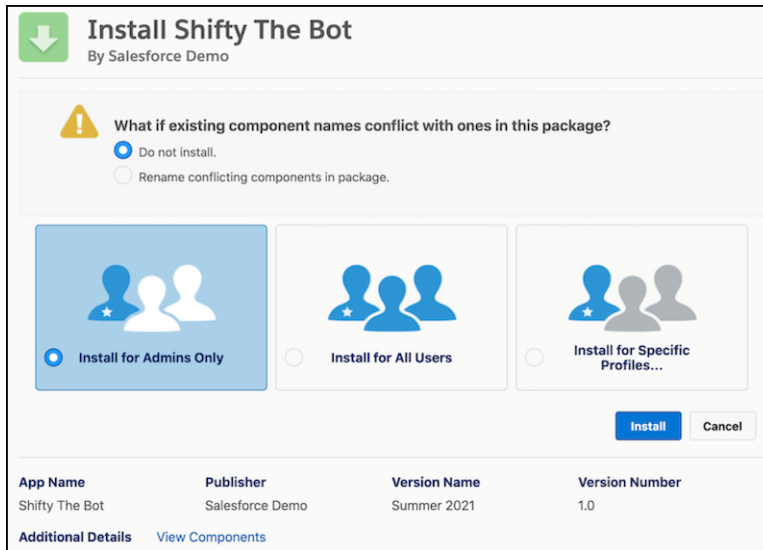
1. From Setup, in the Quick Find box, enter *Einstein Bots*, and then select **Einstein Bots**.



### Step 2: Install Package

Install Shifty the bot in Salesforce.

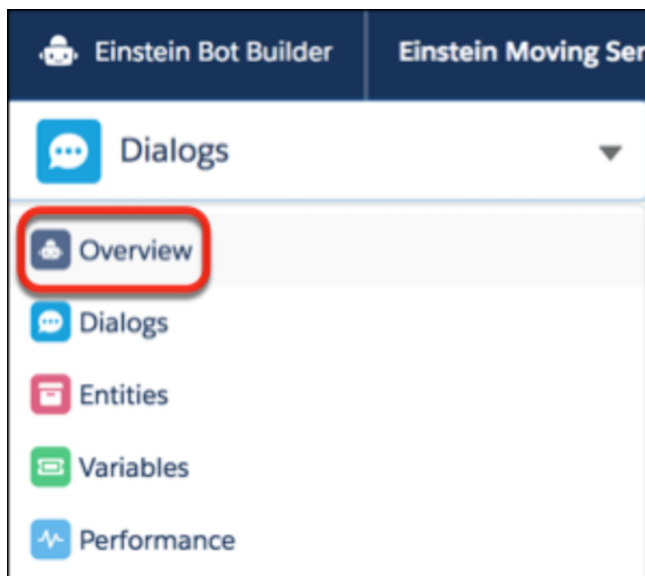
1. Go to the following link: <https://login.salesforce.com/package/installPackage.apexp?p0=04t5e000000W0ao>
2. Log in as a Salesforce Admin.
3. Select **Install for Admins only**.



### Step 3: Deploy Shifty the Bot to Your SMS Channel.

To allow users to start interacting with Shifty, deploy your bot to SMS.

1. From the Bot Builder menu, click *Overview*.



2. In the Connections section, click **Add**.




Overview

Activate

Bot Information

Edit



| NAME    | API NAME | DESCRIPTION | LANGUAGE | LAST MODIFIED | LAST MODIFIED BY |
|---------|----------|-------------|----------|---------------|------------------|
| Cat bot | Cat_bot  |             | English  | Jul 27, 2021  |                  |

Conversation Languages

Add

Language

English (Primary)

Connections

Add

| Connection | Deployment   | Require Agent Online |
|------------|--------------|----------------------|
| Chat       | The Cat team | Off                  |

3. Select the channel and search for the deployment for your bot.
4. To find your SMS Channel name, from Setup, in the Quick Find box, enter *Messaging Settings*, and then select **Message Settings**.

## Test This Example

To test this example: Text from a phone number that has an associated user record. Shifty answers and offers options.

