



Embedded Chat

Salesforce, Summer '19



CONTENTS

Embedded Chat	1
Set Up Embedded Chat	4
Set Up Appointment Home	23
Embedded Flows	25
Test Your Chat Window	26
Limitations	26
Index	28

EMBEDDED CHAT

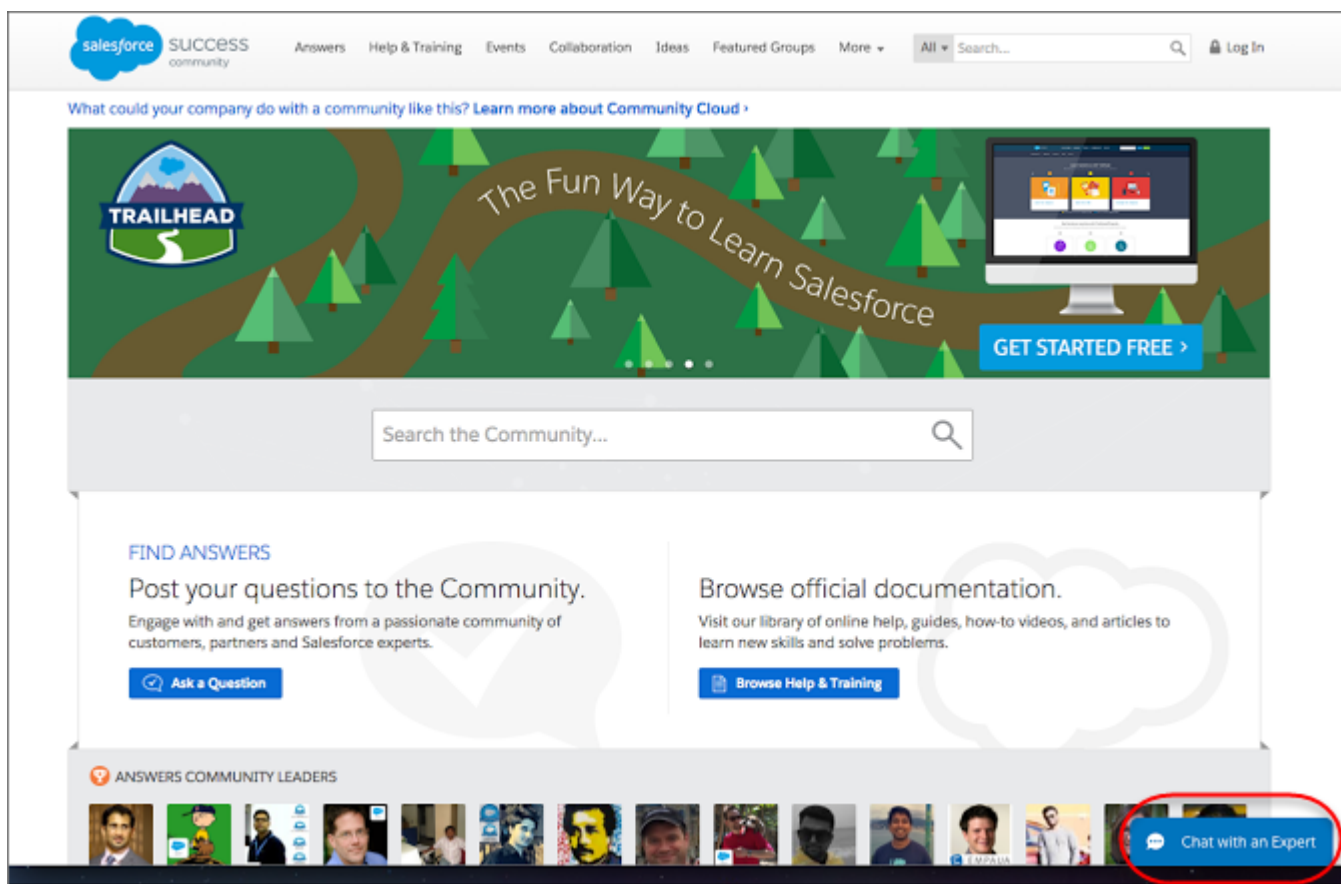
Add an embedded chat window to your website so customers can quickly get answers to their questions by chatting with an agent while browsing your site.

The chat window sits unobtrusively on the web page. When customers want to chat, they just click the button to launch the chat.

EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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



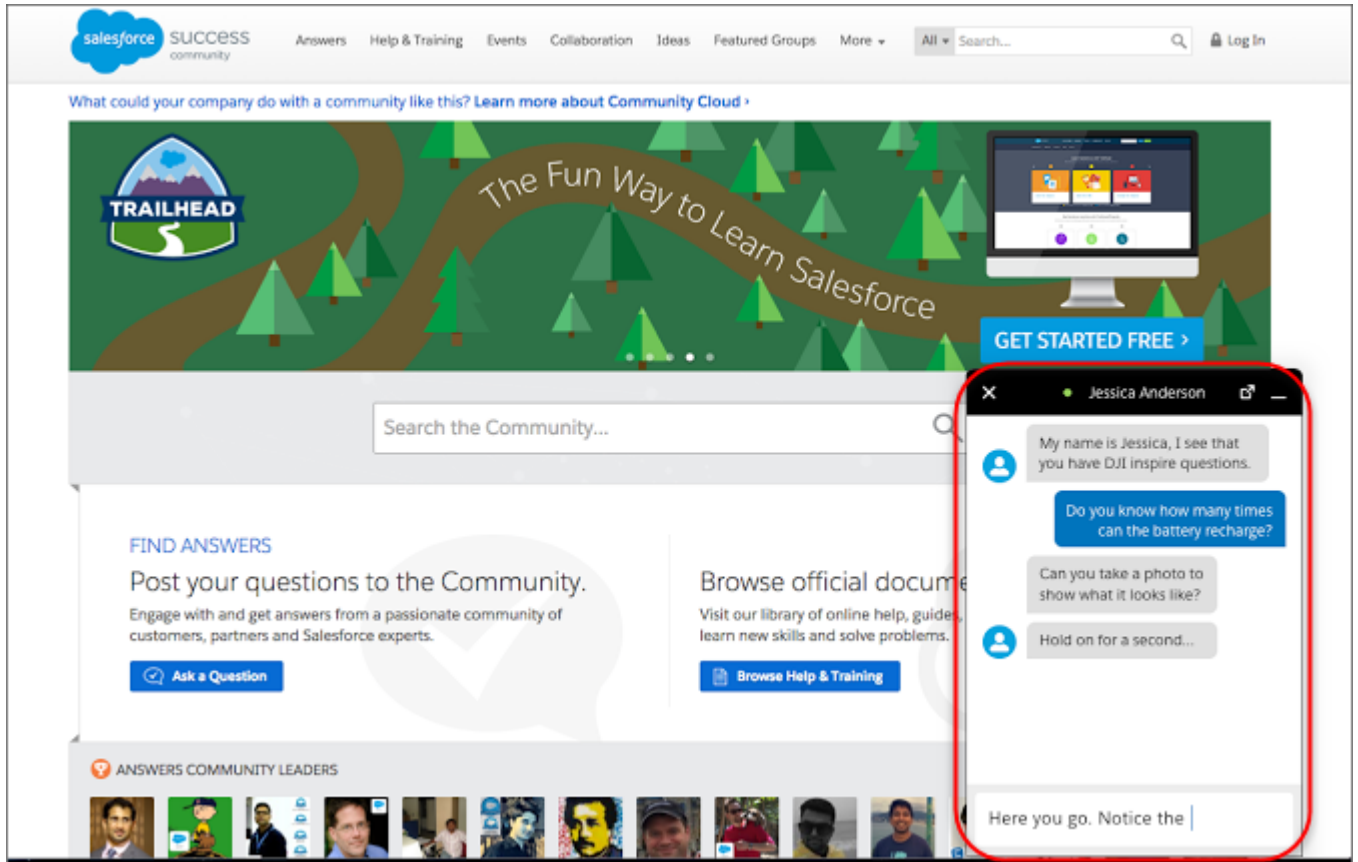
Customers fill out the brief pre-chat form, which helps agents gather basic information about the customer, like their contact information, and their needs.

Embedded Chat

The image shows a screenshot of the Salesforce Success Community website. The page features a navigation bar with links for Answers, Help & Training, Events, Collaboration, Ideas, Featured Groups, and More. A search bar and a Log In button are also present. The main content area includes a banner for Trailhead with the text "The Fun Way to Learn Salesforce" and a "GET STARTED FREE" button. Below the banner is a search bar for the community. The page is divided into sections for "FIND ANSWERS" (Post your questions to the Community) and "Browse official documents" (Visit our library of online help, guides, and learn new skills and solve problems). At the bottom, there is a section for "ANSWERS COMMUNITY LEADERS" with a row of profile pictures. An embedded live chat window is overlaid on the right side of the page, containing a form with the following fields: Full Name* (Esther Nelson), Your Email* (esthern@gm), and Description (What brings you here today?). A "Start Chatting" button is at the bottom of the chat window.

Customers can start chatting while viewing your web page and can minimize the chat window as they browse, so it's not in their way. The chat window persists across your web pages and their subdomains, so customers can continue browsing other pages on your site while chatting with an agent.

Embedded Chat



Embedded chat windows use a Chat deployment that you can quickly configure. Then, simply add the code snippet to the web pages where you want the chat window to be available. Setting up Embedded Chat doesn't change how your agents chat with customers in their console.

[Set Up Your Embedded Chat Window](#)

Ensure that your org meets the following prerequisites before starting to set up Embedded Chat. Follow these steps to configure the embedded chat window and add it to your web pages. The setup node for Embedded Chat is available only in Lightning Experience.

[Set Up Appointment Home](#)

Appointment Home gives your customers an easy way to schedule, modify, and cancel appointments. The experience is powered by Visual Workflow. This lets you craft the perfect interaction and decide when and how to create related records like work orders.

[Add a Flow to Your Website](#)

Use Embedded Service to add a flow to your website. Embedded flows help your users complete simple, automated tasks. They're even guest user-friendly—no login required. Simply create your flow using Flow Builder and add it to an Embedded Service deployment.

[Test Your Chat Window](#)

Now that you've added Embedded Chat to your web pages or Community, it's time to test it out.

[Embedded Chat Limitations](#)


Embedded Chat has the following limitations.

Set Up Your Embedded Chat Window

Ensure that your org meets the following prerequisites before starting to set up Embedded Chat. Follow these steps to configure the embedded chat window and add it to your web pages. The setup node for Embedded Chat is available only in Lightning Experience.

To set up Embedded Chat, your org must meet these prerequisites:

- [Lightning Experience must be enabled](#) to set up Embedded Service
- Service Cloud License
- Chat License
- [Chat must be enabled](#) in your org
- A Chat button and a Chat deployment must be set up and available in your org
- A [Salesforce Community](#) (preferable) or a Salesforce Site must be set up on your org and available for guest user access
- Ensure your browsers are supported for Embedded Chat. We support the same browsers as Lightning Communities. See [Browser Support for Communities](#) for more information.

 **Note:** Administrators must use Lightning Experience to access the Embedded Service setup pages. However, other users in the org aren't required to have access to or use Lightning Experience with Embedded Chat.

To set up Embedded Chat, follow these steps:

1. [Create an Embedded Service Deployment](#)
Create an Embedded Service deployment for each embedded chat window that you're using. The setup node for Embedded Chat is available only in Lightning Experience.
2. [Specify Your Chat Window's Settings](#)
Select your Chat deployment and chat button. Customize the pre-chat form and offline support form. Select custom Lightning components. Set chat images, dimensions, and labels. Learn about available Chat features in Embedded Chat.
3. [Customize the General Branding for Your Chat Window](#)
Select the colors and font used in your chat window to reflect your company's brand identity.
4. [Adding Your Chat Window to a Website](#)
Salesforce generates a unique code snippet based on the information you provided during the Embedded Service setup. Copy and paste the chat code snippet so you can add the chat window to your web pages. Copy and paste the optional meta tag code snippet to make your web pages responsive to different form factors such as mobile and desktop.
5. [Adding Your Chat Window to a Community](#)
Use the Embedded Chat component to add your chat window to a community. Create a snippet settings file to use snippet-only settings with your community's chat window.

SEE ALSO:

- [Turn on Lightning Experience for Your Org](#)
- [Supported Browsers for Lightning Experience](#)

EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- [Customize Application AND Modify Metadata](#)

Create an Embedded Service Deployment

Create an Embedded Service deployment for each embedded chat window that you're using. The setup node for Embedded Chat is available only in Lightning Experience.

The Embedded Chat setup uses a Salesforce Community or Salesforce Site, so that you can associate users with a guest user profile. Check if you already have any communities or sites that you can use. If a site is already available, you can use that as your endpoint. Otherwise, create a site for this purpose.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. In the Embedded Service configuration page, click **New Deployment**.
3. In the `Embedded Service Deployment Name` field, enter a name for your chat window.
The deployment is shown in the list of deployments on the Embedded Service page. If you have several Embedded Service deployments, use a descriptive name so it's easy to distinguish each of your deployments.
4. The `API Name` field is automatically populated.
5. In the `Site Endpoint` menu, select a Salesforce community or Salesforce Site from the dropdown list. If nothing appears in the menu, it's because there aren't any communities or sites configured in your org.
6. Click **Save**.

SEE ALSO:

[Specify the Chat Settings for Your Embedded Chat Deployment](#)

Specify Your Chat Window's Settings

Select your Chat deployment and chat button. Customize the pre-chat form and offline support form. Select custom Lightning components. Set chat images, dimensions, and labels. Learn about available Chat features in Embedded Chat.

[Specify the Chat Settings for Your Embedded Chat Deployment](#)

Embedded Chat uses a Chat deployment ID and button. The Chat configuration details that you provide are used to generate the chat code snippet that you add to your web pages.

[Customize the Pre-Chat Form](#)

Gather contact information from your customers and find out about their needs using a pre-chat form. You can create a pre-chat form that addresses different business needs and associates customer information with Salesforce records like leads, cases, and contacts. You can also customize the fields used on the form.

[Customize the Offline Support Form](#)

Let customers submit a case form when your support agents are offline.

[Customize Additional Chat Branding](#)

Set the base font size, width and height, and images for your Embedded Chat deployment.

[Use Custom Components With Your Chat Window](#)

Create custom components to control the user interface for your chat window.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- **Customize Application AND Modify Metadata**

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

[Available Chat Features in Embedded Chat](#)

Most Chat features are supported with Embedded Service for Web. Enable the following features in your Chat configuration, deployment, and chat button to provide chat features for your support agents and chat visitors. Chat features can be enabled only in Salesforce Classic.

Specify the Chat Settings for Your Embedded Chat Deployment

Embedded Chat uses a Chat deployment ID and button. The Chat configuration details that you provide are used to generate the chat code snippet that you add to your web pages.

Before you specify the Chat settings, ensure that you have a Chat deployment and a Chat button available to use with Embedded Chat.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. In the Embedded Service configuration page, go to the `Chat settings` section and click **Start**.
4. In the `Chat Deployment` menu, select the Chat configuration that you want to use with the chat window from the dropdown list.
5. In the `Chat Button` menu, select the Chat button or automated invitation that you want to use with the chat window from the dropdown list.
6. Select **Show queue position** if you want to display the customer's place in line while they wait to chat with a support agent or to be transferred. Make sure that the Chat button you selected has **Enable Queue** selected in your Chat button settings.
7. Click **Save**.

After you've provided the Chat settings, get the code to add the chat window to your web pages.

SEE ALSO:

[Available Chat Features in Embedded Chat](#)

[Adding Your Chat Window to a Website](#)

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata

Customize the Pre-Chat Form

Gather contact information from your customers and find out about their needs using a pre-chat form. You can create a pre-chat form that addresses different business needs and associates customer information with Salesforce records like leads, cases, and contacts. You can also customize the fields used on the form.

Important: Before you start, make sure that the permission “Enable Support QuickAction Rest endpoint for Guest Users” is turned on for your org. If the permission isn’t enabled, your customers can’t request a chat.

EDITIONS

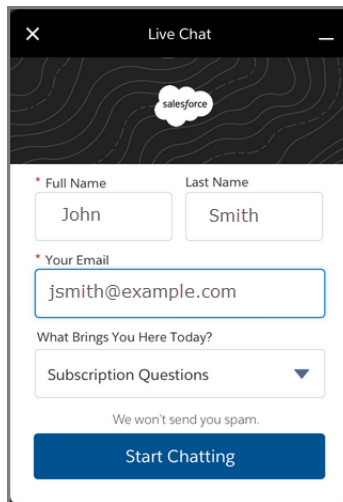
Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata




When you design your pre-chat form, you can select a use case scenario that automatically associates information from the form with Salesforce records. When a customer enters their name or email address into the form, Salesforce matches the information with a Salesforce contact record or lead record. If no match is found, Salesforce creates a record.

For example, if you select the Service scenario, Salesforce relates the customer’s information with Case and Contact records.

Table 1: Use Cases for Associating Pre-Chat Information with Salesforce Records

If you select:	Then Salesforce associates the pre-chat information with these record types:
Sales	Lead records
Service	Contact records Case records
Basic	Contact records


You can select the fields that are shown on the pre-chat form, change the order of the fields, and make fields required. To make the form more user-friendly to customers, add up to four fields.

 **Important:** Create and edit your pre-chat forms from Embedded Service setup, not at the quick action. Editing the quick action can lead to unexpected results.

Embedded Service setup creates quick actions specifically for pre-chat. Using these quick actions for other purposes causes pre-chat setup to fail.

To set up the pre-chat form:

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. Click **Edit** next to Chat Settings.
4. In the Pre-chat section, move the radio button to **Active**.
5. Click **Edit**.
6. Select the use case for the pre-chat form.
 - a. Select the main reason that customers use chat. The reason that you select affects the type of Salesforce record that's created from the information that customers enter into the pre-chat form.
 - b. Select the record type that's created from the pre-chat form. The drop-down menu shows record types that are available on the object.
 - c. Click **Next**.
7. Select the fields shown on the pre-chat form.
 - a. To change the order of the fields, use the up arrow and down arrow.


 **Note:** For the Service scenario, fields are grouped by object. Contact fields always appear above Case fields.
 - b. To add a field, click **+**. You can add only fields that are available on the object.
 - c. To remove a field, click **X**.
 - d. To require customers to fill out a field, select **Required** next to the field.
 - e. Click **Save**.
8. Optionally, select a custom Lightning Component to replace the standard component. To learn more about using custom components for pre-chat, see Custom Lightning Components in the *Embedded Service for Web Developer Guide*.

Turn off your pre-chat form at any time by viewing your deployment in Setup and moving the pre-chat page slider to **Inactive**.

 **Note:** You can use the pre-chat APIs to send nonstandard pre-chat details along with what's available in setup. For more information, see the Enhance the Pre-Chat Page in the *Embedded Service for Web Developer Guide*.


Customize the Offline Support Form

Let customers submit a case form when your support agents are offline.


 **Important:** Before you start, enable Web-to-Case and make sure that the permission “Enable Support QuickAction Rest endpoint for Guest Users” is turned on for your org. If the permission isn't enabled, your customers can't request a chat.

You can select the fields that are shown on the offline support case form, change the order of the fields, and make fields required. To make the form more user-friendly to customers, add up to 4 fields.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. Click **Edit** next to Chat Settings.
4. In the Offline support section, move the radio button to **Active**.
 - a. If you see a warning about enabling guest access for your associated Salesforce site, click the link to enable it.
5. Click **Edit**.
6. Select a Case record type.
7. Select the fields shown on the form.
 - a. To change the order of the fields, use the up arrow and down arrow.
 - b. To add a field, click **+**. You can add only fields that are available on the object.

 **Note:** If you add a field with the `unique` attribute and a visitor enters a duplicate field value, the case submission confirmation is displayed even though the case isn't created. Add unique fields to this form only if you're certain that visitors are unlikely to enter duplicate values.
 - c. To remove a field, click **X**.
 - d. To require customers to fill out a field, select **Required** next to the field.
8. Optionally, enter a URL for the offline support header image. The image is displayed in the header of the chat window when the customer opens the offline support form.
9. Click **Save**.
10. Refresh your code snippet and paste it into your website.

Turn off your offline support form at any time by viewing your deployment in Setup and moving the Offline support slider to **Inactive**.

 **Important:** Any time you turn offline support on or off, you must refresh the code snippet and repaste the code snippet into your website.

EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata

Customize Additional Chat Branding

Set the base font size, width and height, and images for your Embedded Chat deployment.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. Click **Edit** next to Chat Settings.
4. In the Additional branding section, click **Edit**.
5. In the Images & Dimensions tab, set your chat images.

EDITIONS

Embedded Service setup node is available in: Lightning Experience


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

USER PERMISSIONS

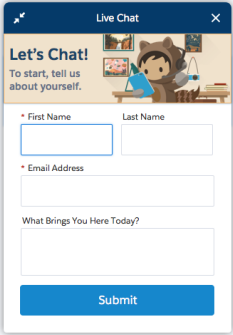
To set up Embedded Chat:

- Customize Application AND Modify Metadata

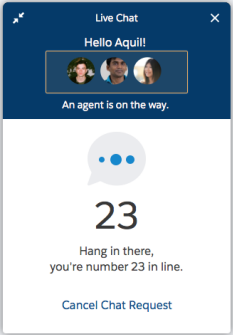
Minimized Logo
16x16



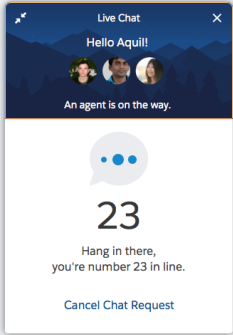
Pre-Chat Background
320x100



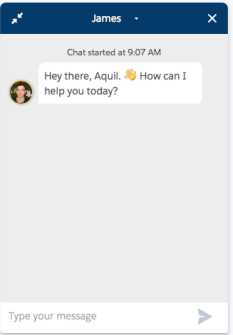
Waiting State
200x60



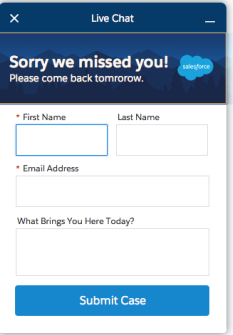
Header Background Image
320x164



Agent Avatar
40x40



Offline Case Image
320x100



6. Set the width and height of your chat window. If you don't make any changes, the default sizes of 320 px and 498 px are used.
7. Set the base font size for your chat window.
8. Click **Finish** to close the window.

[Customize Labels](#)

Create your own labels for your Embedded Chat deployment.

Customize Labels

Create your own labels for your Embedded Chat deployment.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. Click **Edit** next to Chat Settings.
4. In the Additional branding section, click **Edit**.
5. Click the **Labels** tab.

EDITIONS

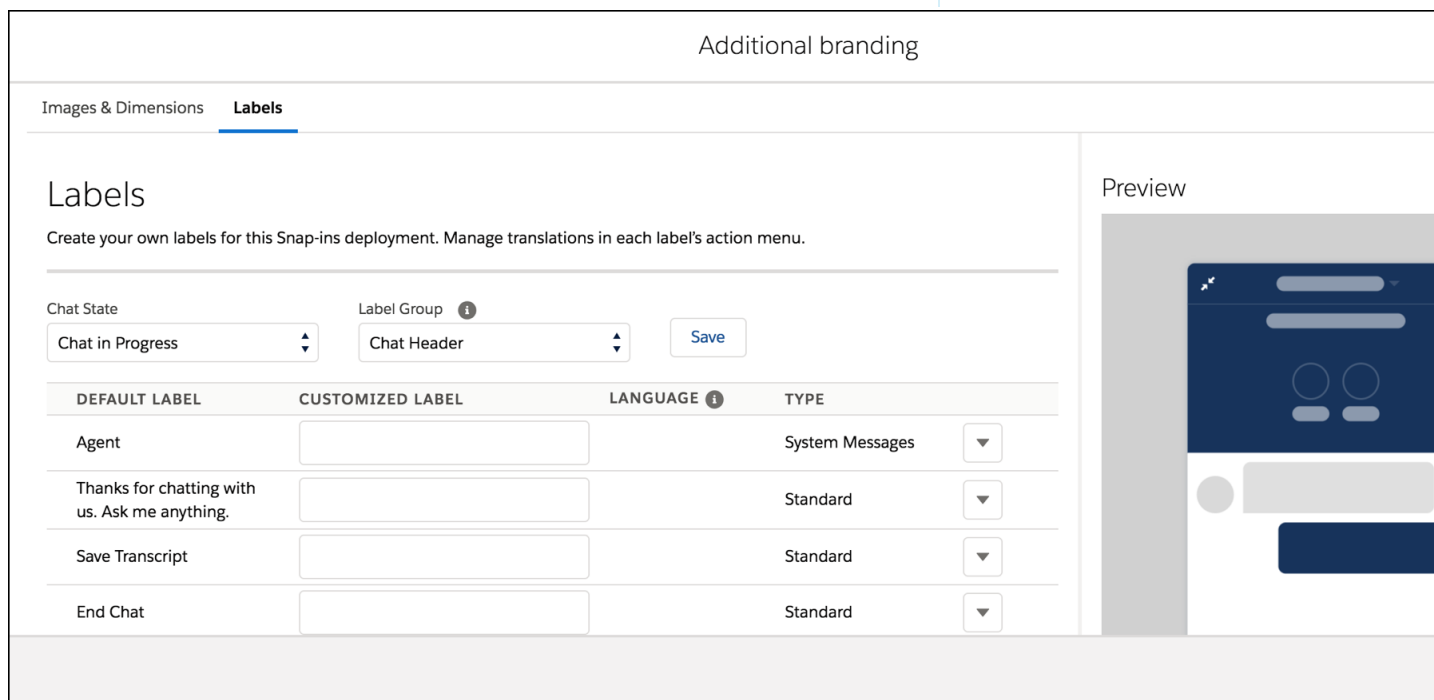
Embedded Service setup node is available in: Lightning Experience

Available in: **Enterprise, Performance, Unlimited,** and **Developer** editions.

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata

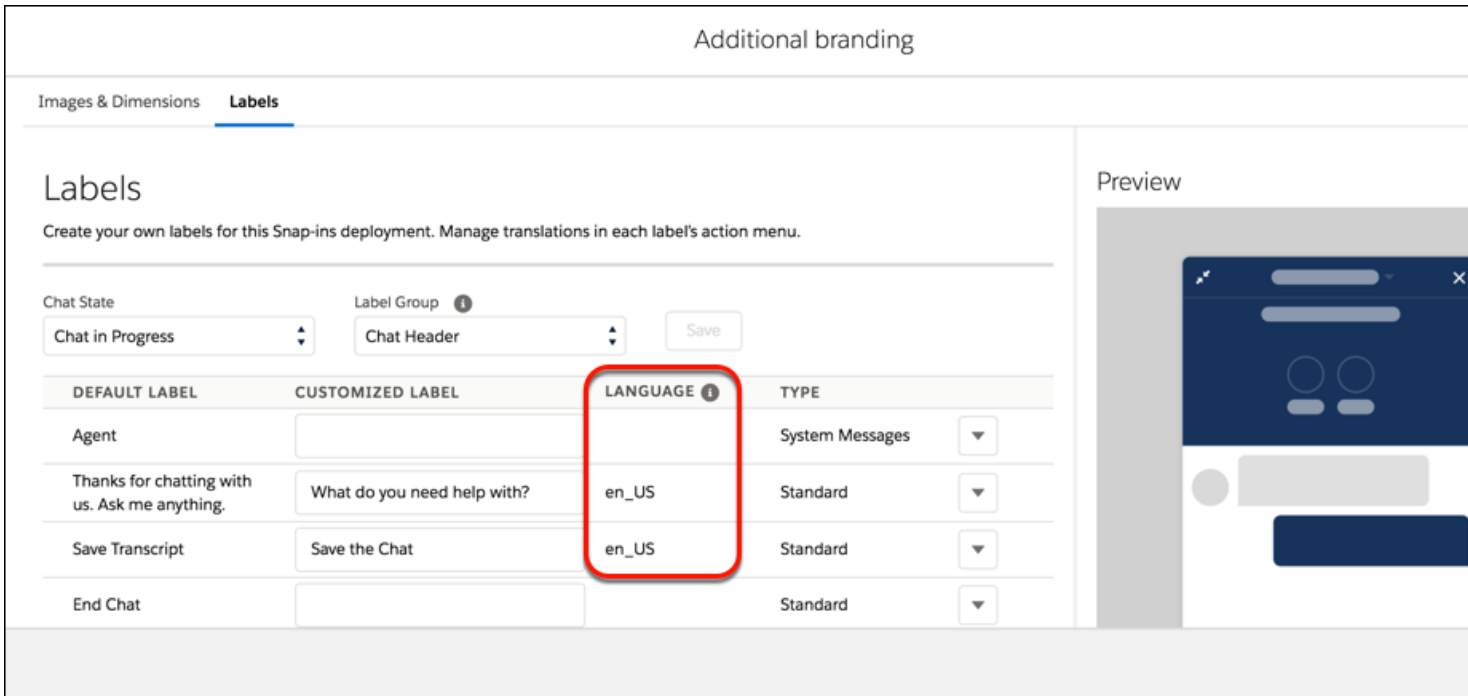


6. Select the Chat State and Label Group you want to create custom labels for.

7. Under Customized Label, enter new labels for your embedded component. The live preview highlights where the selected label appears on the screen.
8. Save your customized labels before moving on to another chat state or label group.
9. To close the window, click **Finish**.

When you or another administrator go to the Labels tab, the Language column indicates the language that each customized label was originally written in.

To reset your customized labels, click the arrow next to the label type. Then select **Reset to Default**.



[Translate Your Chat Window](#)

The primary language for a chat window is set differently when Translation Workbench is enabled or disabled.

SEE ALSO:

[Translate Your Chat Window](#)

Translate Your Chat Window

The primary language for a chat window is set differently when Translation Workbench is enabled or disabled.

If you're using Translation Workbench:

- The language set in the visitor's Accept-Language HTTP header for their browser is used when it matches a language in the list of active Translation Workbench languages. When the visitor has multiple languages set for their browser, the highest-ranked language that matches a language in the list of active Translation Workbench languages is used.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

- When there's no match, the language specified for the Site Guest User of the Site associated with your Embedded Service deployment is used.

If Translation Workbench is disabled, the language specified for the Site Guest User of the Site associated with your Embedded Service deployment is used.

! **Important:** To use translation, you must allow cookies for your browser. For example, enable "Allow third-party cookies" in Safari iOS.

To translate labels in your chat window, view your Embedded Service deployment in Setup, open your Chat or Field Service settings, and click **Edit** next to Additional branding. Click the **Labels** tab to view and customize the labels in your chat window. Click the arrow next to a label and click **View Label Record**, where you can select to translate the label in the Translation Workbench.

SEE ALSO:

[Support Users in Multiple Languages](#)

Use Custom Components With Your Chat Window

Create custom components to control the user interface for your chat window.

SEE ALSO:

[Embedded Service for Web Developer Guide: Custom Lightning Components for Embedded Service for Web](#)

Available Chat Features in Embedded Chat

Most Chat features are supported with Embedded Service for Web. Enable the following features in your Chat configuration, deployment, and chat button to provide chat features for your support agents and chat visitors. Chat features can be enabled only in Salesforce Classic.

Table 2: Supported Chat Features for Embedded Chat

Chat feature:	Enable it in:	Description
Save Transcript for Visitors	Always enabled	Lets the chat visitor save their chat transcript. They can save the transcript during or after the chat.
Sneak Peek	Chat Configuration	Lets agents see what the chat visitor is typing before sending a chat message.
Visitor Blocking	Chat Configuration	Lets agents block visitors from an active chat. When a visitor is blocked, any chat attempt from the IP address is denied.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

Chat feature:	Enable it in:	Description
Agent File Transfer	Chat Configuration	Lets agents request a file from the chat visitor.
Chat Transfer (to agent, skill, or chat button)	Chat Configuration	Lets agents transfer a chat directly to another agent, or to an agent assigned to a particular skill or chat button. When a chat is transferred to a button or skill, the transfer request is sent to all available agents assigned to that button or skill, and the chat is transferred to the first agent to accept the request.
Customer Time-Out	Chat Button	Lets you set a warning and timeout for when the chat visitor is idle during a chat.
Post-Chat Page URL	Chat Button	Lets you provide a link at the end of each chat to direct chat visitors to more resources or a survey. Adding a post-chat link creates a button. Add a code snippet setting to automatically open the post-chat URL when the chat ends.
Automated Invitations	Chat Button	Lets you use a proactive chat invitation instead of a static button. Requires code snippet version 4.0 or later.
Sensitive Data Rules	Chat Sensitive Data Rules	Lets you omit or replace specified patterns like credit card or social security numbers.
Enable Queue	Chat Button	Lets you display the chat visitor's place in line while they wait to chat with a support agent or to be transferred. Requires code snippet version 5.0 or later.
Chat Conferencing	Chat Configuration	Lets multiple agents join a chat. Chat conferencing is only available for Chat routing in Salesforce Classic. Requires code snippet version 5.0 and later.
Omni Routing	Chat Configuration	Lets you unify your channels and manage your agent workload. Set up rules and queues to automate your service process.

Customize the General Branding for Your Chat Window

Select the colors and font used in your chat window to reflect your company’s brand identity.

Before you start, talk with the team that maintains your website to get the color codes for your company’s brand. You can specify branding colors by entering their hex code or RGB code, or by selecting a color in the color palette.

Important: We don’t support adding custom CSS to your embedded component. We support using only the branding options in setup and in the customizable parameters in the code snippet. If you have custom CSS in your embedded component, it’s your responsibility to test your chat window each release to ensure that it functions properly.

The screenshots show how the branding elements appear in the chat window.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata

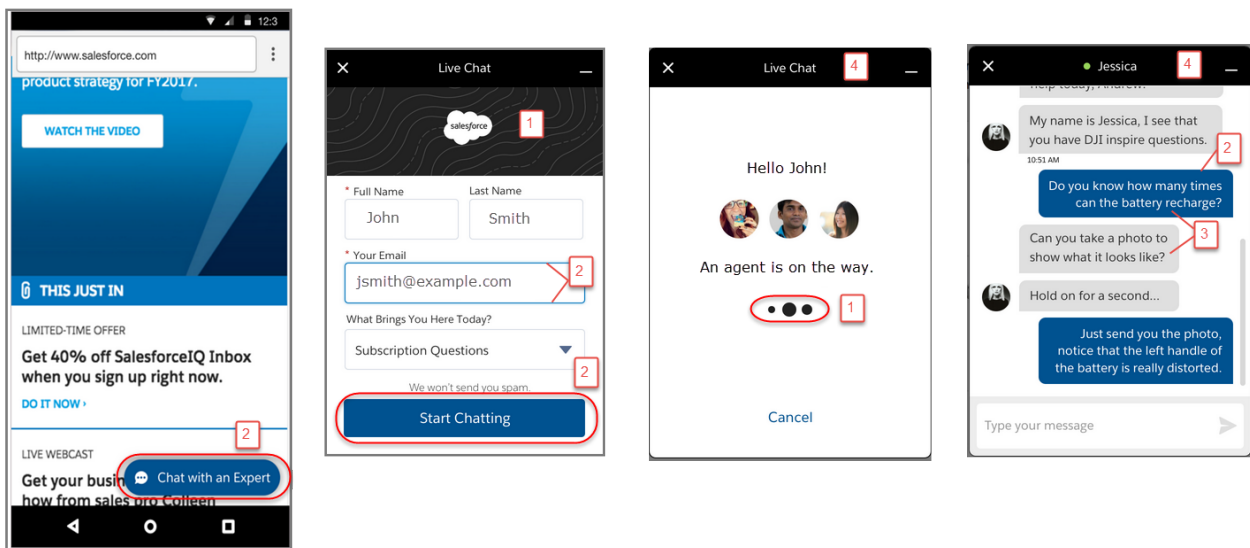



Table 3: Brand Elements That Can Be Customized in a Chat Window

Brand Element	Description
Brand Primary	<p>Sets the color of the:</p> <ul style="list-style-type: none"> • Pre-chat image backdrop • Pop message outlines • Loading balls • Loading balls when chat is minimized <p>By default, the color is set to hex #222222 (Night Black).</p> <p>In the screenshots, brand primary elements are indicated by [1].</p>

Brand Element	Description
Brand Secondary	<p>Sets the color of the:</p> <ul style="list-style-type: none"> • Call to Action buttons, such as “Start Chat” or “Chat with an Expert” • Input field focus • Send button • Guest message chat payload • Secondary buttons • Chat when it is minimized and there’s a new message <p>By default, the color is set to #005290 (Nimbostratus Blue).</p> <p>In the screenshots, brand secondary elements are indicated by [2].</p>
Contrast Primary	<p>Sets the color of the:</p> <ul style="list-style-type: none"> • Chat body text • Chat input text <p>By default, the color is set to #333333 (dark gray).</p> <p>The background color is white.</p> <p>In the screenshots, contrast primary elements are indicated by [3].</p>
Nav Bar	<p>Sets the color of the:</p> <ul style="list-style-type: none"> • Chat widget’s navigation bar • Chat button when the chat is minimized <p>By default, the color is set to hex #222222 (Night Black).</p> <p>In the screenshots, the nav bar element is indicated by [4].</p>
Font	<p>Sets the font used in the chat widget.</p> <p> Note: Some fonts aren’t supported on mobile browsers. If you select a font that isn’t available for a chat visitor’s mobile browser, the browser’s default font is displayed.</p> <p>The following fonts are supported in mobile browsers for both iOS and Android:</p> <ul style="list-style-type: none"> • Georgia • Times New Roman • Arial • Courier New <p>The following fonts are supported in mobile browsers for iOS only:</p> <ul style="list-style-type: none"> • Trebuchet MS • Verdana • Lucida Console

To customize the branding of the chat window:

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with.
3. In the Embedded Service configuration page, go to the Branding section and click **Edit**.
4. Select the colors that you want to show in the chat window.

Enter the color using the hex code. To enter an RGB code or to select a color from the palette, open the color palette by clicking the down arrow in the color field.

5. Select the font that you want to use in the chat window.
6. Click **Finish**.

Important: Changing the branding of an existing chat window changes the code snippet for the chat window. If you update the branding for an existing chat window, you must update the code snippet on your webpages. If you customized the code snippet, you must add those customizations to the new code snippet.

[Customize Chat Window Branding in a Community](#)

Select the colors and font used in your community chat window to reflect your company's brand identity.

SEE ALSO:

[Adding Your Chat Window to a Website](#)

[Customize the Embedded Service Code Snippet](#)

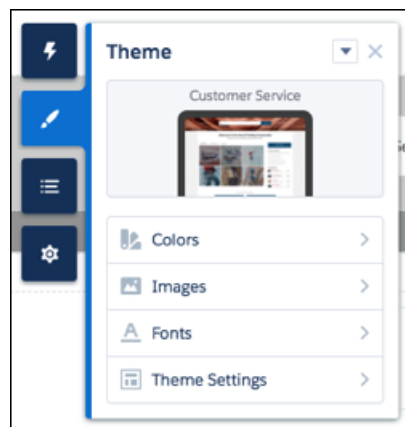
[Customize Additional Chat Branding](#)

Customize Chat Window Branding in a Community

Select the colors and font used in your community chat window to reflect your company's brand identity.

To customize chat window branding colors in a community, use the settings in the Embedded Chat community component. Using Embedded Service setup to specify branding colors in communities is not supported.


1. From the Communities Builder, make sure your chat window is selected. Navigate to the Theme tab.




EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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

2.  **Note:** In Communities, Action Color maps to the Brand Primary and Brand Secondary colors for web chat windows. Text Color maps to the Contrast Primary color for web chat window.

Select the colors for your chat window.

-  **Important:** Standard branding tokens don't work. Add the following CSS to your community:

```
.sidebarHeader {
background-color: #000000 !important;
}
```

Replace #000000 with the Hex Color Code you want.

3. Select your font and add images to your chat window.
4. Save your community.

SEE ALSO:

[General Embedded Service Branding](#)

Adding Your Chat Window to a Website

Salesforce generates a unique code snippet based on the information you provided during the Embedded Service setup. Copy and paste the chat code snippet so you can add the chat window to your web pages. Copy and paste the optional meta tag code snippet to make your web pages responsive to different form factors such as mobile and desktop.

Ensure that you have access to the web pages where you want to add the chat window. You might need to work with your company's webmaster.

When you add the chat code snippet to your web pages, customers can see and use the chat window. If you're not ready for customers to access the chat yet, add the code snippets to a private web page.

You can customize the code to change certain aspects of the chat window's appearance.

The `<meta>` tag code makes your web page responsive, so that the web page and the chat window look good and perform well on different devices. For example, if a customer is looking at your web page and chat window on a mobile phone, then the page and chat are resized to accommodate the smaller form factor.

This tag is the recommended `<meta>` tag for proper code snippet responsiveness. If your page is already responsive, then you might not need to add this snippet or change your `<meta>` tag. If you see issues with behavior on other devices, you can try updating your `<meta>` tag to code snippet that's provided. This tag is added to the head section of every page where the chat window is used. If you don't include an appropriate `<meta>` tag, then the chat window appears as it does on a desktop, regardless of the device used to access the page.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with by clicking the arrow beside the deployment name and selecting **View**.
3. In the Embedded Service configuration page, go to the `Embedded Service code snippets` section and click **Get Code**.

EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- **Customize Application AND Modify Metadata**

4. Copy the chat code snippet and paste it immediately above the closing `</body>` tag on your web page to add Embedded Chat to the page.
Paste the chat code snippet into every web page where you want the chat window to appear. Don't place your code snippet on the same page or community as a Chat button. Optionally, customize the chat code snippet.
5. Optionally, copy and paste the meta code snippet into the `<meta>` tag head section in every web page where you want the chat window to appear.
6. Click **Done**.
The code snippets page closes.

[Customize the Embedded Service Code Snippet](#)

Use a text or HTML editor to customize the optional parameters in the Embedded Service code snippet.

[Add Your Website to the CORS Whitelist](#)

Add the URLs of the web pages where you intend to add the chat window to the CORS whitelist in your org. The web page where you add the chat window is the page that customers use to access chat.

SEE ALSO:

[Customize the Embedded Service Code Snippet](#)

[Customizable Parameters in the Embedded Chat Code](#)

[Test Your Chat Window](#)

Customize the Embedded Service Code Snippet

Use a text or HTML editor to customize the optional parameters in the Embedded Service code snippet.

You can customize certain [parameters](#) that affect the appearance and behavior of the chat window. We respect the code snippet before your Embedded Chat settings when determining the chat window's appearance and behavior.

You can customize the code snippet once and paste it on all your web pages, or customize the code snippet for certain web pages.

1. From Setup, enter *Embedded Service* in the Quick Find box, then select **Embedded Service**.
2. Select the Embedded Service deployment that you want to work with.
3. In the Embedded Service configuration page, go to the `Embedded Service Code Snippets` section and click **Get Code**.
4. Copy the code snippet to a text editor or HTML editor.
5. In the text editor or HTML editor, specify the image URLs for the chat window parameters.
6. Save your changes.
7. Go to the community builder for your help center.
8. Below components, click **Settings**.
9. In the navigation tree, click **Security**.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- **Customize Application AND Modify Metadata**

10. Under **Content Security Policy**, select **Allow inline scripts and script access to whitelisted third-party hosts** in the **Script Security Level** dropdown.
11. In the code snippet, find the field `baseLiveAgentContentURL` and copy the URL.
12. Click **Add Trusted Site** and name your new site.
13. Click **Add Site** and paste the URL from `baseLiveAgentContentURL`.
14. Publish your community changes.

After modifying the code snippet, [add the code](#) to every web page where you want the chat window to appear.

[Customizable Parameters in the Embedded Chat Code](#)

You can customize certain parameters that affect the appearance and behavior of the chat window so that the chat experience reflects your company's branding. Use these parameters to customize the pre-chat banner image, logo, waiting state image, and your agent's avatar picture. You also can customize the wording that appears on the chat button and on the chat window when the chat is loading, when agents are online, when agents are offline.

SEE ALSO:

[Adding Your Chat Window to a Website](#)

[Customizable Parameters in the Embedded Chat Code](#)

Customizable Parameters in the Embedded Chat Code

You can customize certain parameters that affect the appearance and behavior of the chat window so that the chat experience reflects your company's branding. Use these parameters to customize the pre-chat banner image, logo, waiting state image, and your agent's avatar picture. You also can customize the wording that appears on the chat button and on the chat window when the chat is loading, when agents are online, when agents are offline.

The following parameters are customizable in the code snippet only.

- Set the domain for your chat window to persist across subdomains: `embedded_svc.settings.storageDomain = "..."`
- Set the language: `embedded_svc.settings.language = "..."`
- Display the default chat button: `embedded_svc.settings.displayHelpButton = "..."`
- Set a routing order: `embedded_svc.settings.fallbackRouting = ["...", "..."]`
- Use direct-to-button routing: `embedded_svc.settings.directToButtonRouting = ["...", "..."]`
- Load files for custom chat events:
 - `embedded_svc.settings.externalScripts = ["...", "..."]`
 - `embedded_svc.settings.externalStyles = ["...", "..."]`
- Pass nonstandard pre-chat details: `embedded_svc.settings.extraPrechatFormDetails = ["...", "..."]`

The following parameters are customizable in both the code snippet and Embedded Service setup. Settings in your code snippet override what you enter in Embedded Service setup.

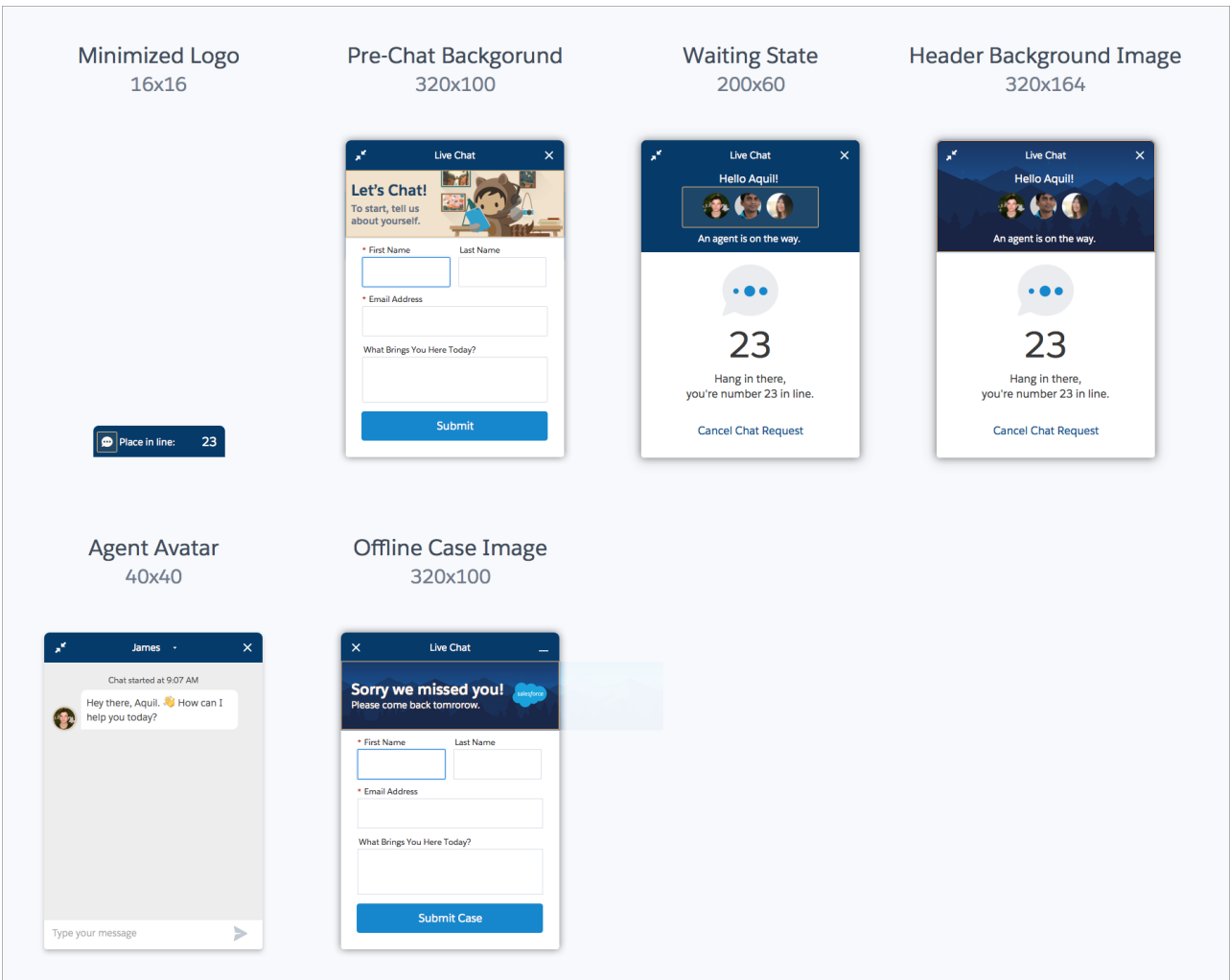
- Set the chat window size and base font size:
 - `embedded_svc.settings.widgetWidth = "..."`
 - `embedded_svc.settings.widgetHeight = "..."`
 - `embedded_svc.settings.widgetFontSize = "..."`

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

- Set the chat images:



- `embedded_svc.settings.prechatBackgroundImgURL = "..."`
- `embedded_svc.settings.smallCompanyLogoImgURL = "..."`
- `embedded_svc.settings.waitingStateBackgroundImgURL = "..."`
- `embedded_svc.settings.avatarImgURL = "..."`
- `embedded_svc.settings.chatbotAvatarImgURL = "..."`
- Customize the text on the chat window:
 - `embedded_svc.settings.defaultMinimizedText = "..."`
 - `embedded_svc.settings.disabledMinimizedText = "..."`
 - `embedded_svc.settings.offlineSupportMinimizedText = "..."`

```
- embedded_svc.settings.loadingText = "..."
```

SEE ALSO:

[Adding Your Chat Window to a Website](#)


[Customize the Embedded Service Code Snippet](#)

[Embedded Service for Web Developer Guide](#)

Add Your Website to the CORS Whitelist

Add the URLs of the web pages where you intend to add the chat window to the CORS whitelist in your org. The web page where you add the chat window is the page that customers use to access chat.

You can use HTTP and HTTPS domains with Embedded Chat. The protocol for the URL that you whitelist must match the URL in the site or community endpoint in the code snippet that's generated at the end of Embedded Service setup.

 **Important:** This information applies to Embedded Service only. If you want to use the CORS whitelist for other Salesforce products and features, see [Use CORS to Access Salesforce Resources from Web Browsers](#).

1. From Setup, enter `CORS` in the Quick Find box, then select **CORS**.
2. Select **New**.
3. Enter an origin URL pattern.

The origin URL pattern must include the HTTP or HTTPS protocol and a domain name. The origin URL pattern can include a port. The wildcard character (*) is supported and must be in front of a second-level domain name. For example, `https://*.example.com` adds all subdomains of `example.com` to the whitelist.

The origin URL pattern can be an IP address. However, an IP address and a domain that resolve to the same address are not the same origin, and you must add them to the CORS whitelist as separate entries.

Adding Your Chat Window to a Community

Use the Embedded Chat component to add your chat window to a community. Create a snippet settings file to use snippet-only settings with your community's chat window.

There are some differences to keep in mind when adding your chat window to a community instead of a website.

- Do not use your Embedded Service code snippet. Use the Embedded Chat community component.
- To ensure that chat sessions are continued across pages, place the Embedded Chat community component in your community's template footer.
- Use the settings in the component. What you set in the component overrides what you select in Embedded Service setup.
- You can't use automated invitations.
- You can't fill in pre-chat fields for logged-in users via the code snippet. Use the component option, which fills in contact fields (name and email).

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

USER PERMISSIONS

To set up Embedded Chat:

- Customize Application AND Modify Metadata

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

- To customize Embedded Chat branding colors in a community, use the settings in the component. Using Embedded Service setup to specify branding colors in communities is not supported.

To use Embedded Service code snippet settings in your community, use the **Snippet Settings File** field in the community component.

1. Create a JavaScript file with your Embedded Chat settings. Refer to the [Embedded Service for Web Developer Guide](#) for more details.
2. Upload the file to your Static Resources. Give the file a name that's easy to remember and doesn't include spaces.
3. In your Embedded Chat community component settings, enter the static resource name (not the file name) in the **Snippet Settings File** field.
4. Save and publish your changes.

SEE ALSO:

[Embedded Chat Community Component](#)

[Customize Chat Window Branding in a Community](#)

Set Up Appointment Home


Appointment Home gives your customers an easy way to schedule, modify, and cancel appointments. The experience is powered by Visual Workflow. This lets you craft the perfect interaction and decide when and how to create related records like work orders.

 **Important:** You must have Field Service Lightning enabled and the latest version of the Field Service Lightning Managed Package installed and working in your org.

1. Update your Field Service Lightning permission sets.
 - a. In the Field Service Admin app, on the Field Service Settings tab, Click **Permission Sets**.
 - b. Ensure the Self Service permission set is up to date.

If you don't have Self Service permission set on in the Field Service Lightning managed package, you don't have the latest version of the package.

2. Install the FSL Appointment Management package.

 **Note:** The templates in this package only give starting point. You can extend flows and apex code to meet your needs.

- a. In the same browser session, go to the URL:

`https://login.salesforce.com/packaging/installPackage.apexp?p0=04t1U0000021A21.`

- b. Click **FSL Appointment Management**.
- c. For What if existing component names conflict with ones in this package? select **Do not install**.
- d. Select **Install for all users**.
- e. Click **Install**.
- f. In the Approve Third-Party Access dialog box, select **Yes, grant access to these third-party web sites**, then click **Continue**.

EDITIONS

Available in: Salesforce Classic ([not available in all orgs](#)) and Lightning Experience

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

USER PERMISSIONS


To configure the Field Service Lightning managed package:


- Customize Application

To create a permission set:

- Manage Profiles and Permission Sets

- g. Wait for an email from support@salesforce.com confirming that Package FSL Snap-In Flows Install was successful before proceeding.
3. Create a connected app with Scheduling and Scheduling_CallBack Visualforce pages.
 - a. From Setup, enter *Visualforce* in the Quick Find box, then select **Visualforce Pages** and click **New**.
The Visualforce Pages page can take up to a minute to load.
 - b. In the `Label` field, enter *Scheduling* and click **Save**.
 - c. Click **Clone** on the Visualforce Pages, Scheduling page.
 - d. On the Visualforce Page page, in the Label and Name fields, enter *Scheduling_Callback* and click **Save**.
 4. Create and configure a connected app.
 - a. From Setup, enter *App Manager* in the Quick Find box, then select **App Manager** and click **New Connected App**.
 - b. On the New Connected App page, in Connected App Name, enter *Snap-ins Appointment Booking*.
 - c. Adjust the API Name to *Snapins_Appointment_Booking*.
Remove the dash (-) in Snap-ins.
 - d. In `Contact Email`, enter your email address.
 - e. Select **Enable OAuth Settings**.
The `Callback URL` field appears.
 - f. In the `Callback URL` field, enter:
*https://*COMMUNITY_DOMAIN_NAME_HERE*/consumer/Scheduling_Callback*.
Occasionally, copying and pasting the string creates erroneous characters. Ensure that the value in the `Callback URL` field matches the string exactly before proceeding to the next step.

 **Important:** Do **NOT** click save yet!
 - g. In `Callback URL` replace **COMMUNITY_DOMAIN_NAME_HERE** with the domain name field value you saved earlier.
 - h. Select **Allow access to your unique identifier (openid)** from Available OAuth Scopes and move it to Selected OAuth Scopes.
 - i. Now, click **Save**, and then **Continue**.
 - j. On the Manage Connected Apps page, copy and save the Consumer Key and the Callback URL.

 **Tip:** To return to the Managed Connected Apps detail page, if you need to get your Consumer Key and/or Callback URL values:

 - a. From Setup, go to App Manager (not Manage Connected Apps).
 - b. Find your scheduling app. For example, Snap-ins Appointment Booking.
 - c. From the row-menu, click the drop down and select **View**.
 - d. You can see the Consumer Key and Callback URL fields there.
 5. Configure the Embedded Service.
 - a. From Setup, enter *Embedded Services* in the Quick Find box, then select **Snap-ins** and click **New Deployment**.
 - b. On the New Snap-in Deployment page, in `Snap-in Deployment Name`, enter *Appointment Booking*.
 - c. Under `Site Endpoint`, select whichever community you want to use and click **Create**.

d. On Snap-in Deployment Settings, click **Start** next to `Field Service settings`.

 **Note:** If you don't have a line for Field Service settings, Snap-ins Appointment Booking isn't enabled.

e. Optionally, select **Let Customers Schedule New Appointments** and **Let Customers View Their Appointments** on the Field Service Settings page, and click **Save**.

f. In the Scheduling Flows section, click **Edit**.

g. Select the following flows.

- New Appointment Flow: **ESW_FS_Main_Flow_multi_data** or **FSL_Snap_ins_New_Appointment_Flow** depending upon your package version
- Modify Appointment Flow: **ESW_FS_Modify_Flow**
- Cancel Appointment Flow: **FSL_Snap_ins_Cancel_Flow**

 **Important:** If there are no values in Appointment Booking Flow, the FSL Scheduling Flows package failed to install.

h. To customize the images used in the flow, click **Edit** under Field Service custom branding.
You can enter your logo in Logo Image URL.

i. To go back to the deployment home, click **APPOINTMENT_BOOKING** at the top of the page.

j. To activate the snap-in, click the slider on Field Service settings.

k. Click **Get Code** next to Snap-in code snippets.

l. On the Snap-in Code Snippets page, click **Copy to Clipboard**, save the code snippets for later, and click **Done**.

6. Create a Cross-Origin Resource Sharing (CORS) record.


a. From Setup, enter `CORS` in the Quick Find box, then select **CORS** and click **New**.

b. Enter `http://sdodemo-main*.force.com` in `Origin URL Pattern`, and click **Save**.

Add a Flow to Your Website

Use Embedded Service to add a flow to your website. Embedded flows help your users complete simple, automated tasks. They're even guest user-friendly—no login required. Simply create your flow using Flow Builder and add it to an Embedded Service deployment.

Before you begin, create a flow using Flow Builder. Then add it to an Embedded Service deployment.

 **Important:** Embedded Service only supports flows of the Screen Flow type. Users do not need to log in to access an embedded flow.

1. [Create an Embedded Service deployment](#).
2. In the Embedded Service Deployment Settings page, click **Start** next to Flow Settings.
3. Under Flow Name, select the flow you want to use in your Embedded Service deployment.
4. Click **Save**.

EDITIONS

Embedded Service setup node is available in:
Lightning Experience

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

USER PERMISSIONS

To set up embedded flows:

- Customize Application AND Modify Metadata

To use a different flow in this Embedded Service deployment or to add custom labels, click **Edit** next to Flow Settings.

SEE ALSO:

[Create an Embedded Service Deployment](#)

[Flow Builder](#)

[Learn more about flow types](#)

Test Your Chat Window

Now that you've added Embedded Chat to your web pages or Community, it's time to test it out.

To test the chat window from the agent's perspective, log in to Salesforce as a Chat user with all the [Chat support agent permissions](#).

1. Open the console and go online as the agent assigned to the appropriate button.
2. Test the chat from the customer's perspective. In another window in the same browser, initiate a chat by navigating to a web page or Community where you've placed the code snippet and starting the chat.

The chat should be available.

3. On the customer side (that is, on your web page), verify that:

- Pre-chat form looks how you expected
- Chat window looks how you expected

4. In the console, verify that:

- You receive a chat request when the chat is initiated

If the chat window and experience appears as you expected, your customers can start using embedded chat to communicate with your agents.

SEE ALSO:

[Permissions for Chat Support Agents](#)

Embedded Chat Limitations

Embedded Chat has the following limitations.

You can't use Embedded Chat on the same page or community as a Chat button. You can't use Embedded Chat and embedded Appointment Management on the same page or community.

The embedded chat window title shows the agent's name instead of the `Chat Window Title` setting. When you enable this setting in the Chat Configuration, it doesn't work in the chat window.

If you want to use an existing Chat button for your embedded chat window, make sure the `Pre-Chat URL` field is blank. When this field is populated, Embedded chat doesn't work. Use the pre-chat fields in Embedded Service setup instead.

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

EDITIONS

Embedded Service setup node is available in: Lightning Experience

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

When you use Embedded Chat in a community, you can't use automated invitations. You also can't set pre-chat fields to fill in for logged-in users in your snippet settings file; use the option to fill in contact fields in your component settings.

Other usage limitations for Embedded Chat:

- Visitors can't chat with an agent if they're using private browsing mode on iOS or Mozilla Firefox with Tracking Protection enabled. They can't chat across tabs if they're using Internet Explorer 11, Microsoft Edge, or iOS devices.
- You can't embed Embedded Chat into Lightning components. Use a Visualforce page, Community, or web property only.
- Field-level validation rules aren't supported with Embedded Chat.
- Embedded Service setup creates quick actions specifically for pre-chat. Using these quick actions for other purposes causes pre-chat setup to fail.
- When you host your Embedded Chat deployment on a Visualforce page, you must access the page using HTTPS. If you use HTTP, the chat window doesn't load.
- To use translation, you must allow cookies for your browser. For example, enable "Allow third-party cookies" in Safari iOS.
- The visitor's recently viewed pages aren't updated during a chat, so the agent can only see which pages the visitor views before the chat starts.
- In offline support case forms, the Case Reason picklist field shows all picklist options regardless of what's specified in the record type.

SEE ALSO:

[Embedded Chat](#)

INDEX

C

Chat limitations [12, 26](#)
Cloud Flow Designer [25](#)
code snippet [19](#)
CORS whitelist [22](#)

E

embedded chat [1, 5–7, 9–10, 13, 15, 18–20, 26](#)
Embedded chat
 custom labels [11](#)
Embedded flow [25](#)
embedded service [1, 18](#)

embedded service for web [19](#)
Embedded Service for Web [13](#)

F

Flows [25](#)

L

Lightning Components [13](#)
limitations [12, 26](#)

T

test the chat window [26](#)