

# Integrate the HelpCenter.io widget with Gorgias Chat

## Integrating the widget

You can substantially reduce the number of live chat customer support requests if you display the [HelpCenter.io](#) widget first and show your Gorgias Chat only when people are unable to find what they need.

Here's how you can integrate the HelpCenter.io widget with Gorgias Chat to do exactly that.

## Install Gorgias Chat

If you haven't done so already, go ahead and install Gorgias chat on your website by [following the instructions in their own help center](#).

## Hide the Gorgias Chat launcher

You want users to see the Gorgias Chat only when they are unable to find the information, they need in your help center. That's why we have to hide that chat bubble and open it up by command from the HelpCenter.io widget.

So, let's hide the Gorgias default chat launcher. Simply, add this script right before the closing `</body>` tag on all your pages where you have Gorgias Chat installed:

```
<script>
  GorgiasChat.init().then(async function() {
    var chatElem = document.getElementById("gorgias-chat-container")
```

```
chatElem.style.display = "none";
GorgiasChat.on('widget:closed', function(data) {
    chatElem.style.display = "none";
});
});
</script>
```

This would make the Gorgias chat icon remain hidden and since we are going to show it manually at some point, it will hide every time the chat messenger gets closed, so we don't end up with 2 overlapping buttons - the [HelpCenter.io](#) widget and Gorgias Chat.

You may need the help of a developer, if you don't know what you are doing. You can also reach out to us. We might be able to help.

## Install the HelpCenter.io Smart Widget

Now that we have the Gorgias chat button hidden, we would like to get the help center widget in place.

Go to your HelpCenter.io dashboard, choose the appropriate help center and then navigate to **Settings -> Widget**.

If you still haven't enabled the widget for your help center, go ahead and click the button to enable it.

You will be presented with the widget options for customization, together with an installation script at the top of the page.

If the installation script is hidden, click on the "Show installation script" button on the top right corner of the page to show it.

Here's the interesting part. **You should modify a bit the JS that you see so that the HelpCenter.io widget would open up the Gorgias chat when somebody clicks on the "I can't find what I'm looking for" button.**

The default HelpCenter.io widget code looks like this:

```
<!-- HelpCenter.io smart widget code -->
<script type="text/javascript">(function(){
  var s = document.createElement('script');
  s.src = "//helpcenter.io/js/init.js";
  s.async = true;
  window.hcOptions = {
    app_id: 'ab12c3'
  };
  document.body.appendChild(s);
})();
</script>
<!-- End HelpCenter.io smart widget code -->
```

where 'ab12c3' will be specific to your help center.

What we need to do is add one more line after `app_id: 'ab12c3'`.

Start with a **comma after the closing single quote on the `app_id` line**, and **place the following piece of code on the next line**:

```
onContactsRequest: function(){GorgiasChat.open(); document.getElementById("go
```

This tells the HelpCenter.io widget to open up your Gorgias chat messenger when a user is unable to find enough information in your help center to solve a certain problem by themselves.

The final variant of your HelpCenter.io widget JS would look something like this:

```
<!-- HelpCenter.io smart widget code -->
<script type="text/javascript">(function(){
  var s = document.createElement('script');
  s.src = "//helpcenter.io/js/init.js";
  s.async = true;
  window.hcOptions = {
    app_id: 'ab12c3',
```

```
        onContactsRequest: function(){GorgiasChat.open(); document.getElement
    };
    document.body.appendChild(s);
  })();
</script>
<!-- End HelpCenter.io smart widget code -->
```

where the 'ab12c3' text will be specific to your help center.

Place the JS snippet that you have right before the closing `</body>` tag of your website or web app and that is all.

Now that you've done all of this, the HelpCenter.io widget will show first, providing your users with helpful information from your help center and a chance to solve their problems alone, unless there's something they couldn't find.

For those cases, they will have the option to activate your Gorgias live chat and easily reach out to your support team with their question, so you know where to improve next.

You can [learn more here about the JS API Gorgias provides for advanced integration of their chat widget.](#)

Listen to your customers when they reach out and extend your knowledge base every time, they ask something new. Doing this continuously should make up for less and less repetitive customer support requests.