

External Widget Deployment Guide

Step-by-step instructions for customizing the widget appearance, generating secure deployment tokens, and embedding the chatbot using all four deployment methods.

- ⌚ ~10 min setup
- 👤 Admin access required
- 🔗 4 Deployment Methods

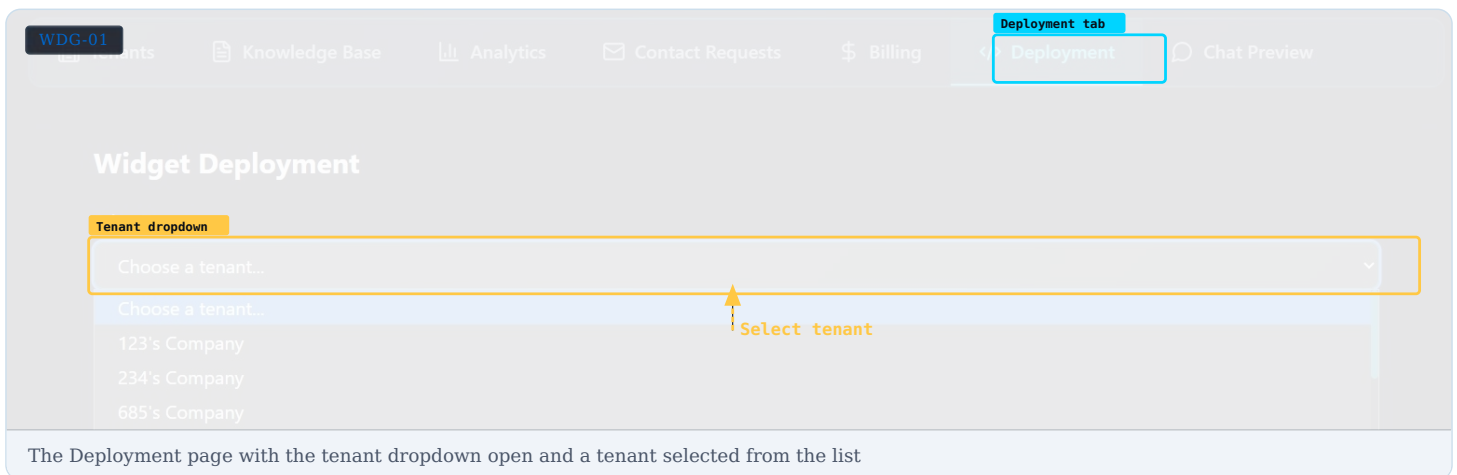


Prerequisite A tenant must already be created and configured before deploying a widget. If you haven't done that yet, complete the *Tenant Setup & Configuration Guide* first.

STEP 01

Open Widget Deployment Settings

1. Log in as an **Admin**.
2. Navigate to **Deployment** from the main dashboard menu.
3. Select the tenant you want to deploy from the dropdown list.



STEP 02

Customize the Widget Appearance

Once a tenant is selected, the **Widget Settings** panel appears. Configure the visual appearance of the chat widget here.

Setting	Description	Default
Widget Enabled	Turn the widget on or off globally for this tenant without deleting any tokens	Enabled
Company Logo	Upload a PNG, JPG, GIF, or SVG up to 2 MB — displayed in the widget header	None
Widget Position	Where the floating button appears: Bottom Right, Bottom Left, Top Right, or Top Left	Bottom Right
Button Color	Background color of the floating chat button — accepts hex codes	#3B82F6
Header Color	Background color of the chat window header bar	#3B82F6
Widget Title	Heading text shown in the chat window header (e.g., "Acme Support")	Chat with us
Widget Subtitle	Short line of text below the title (e.g., "We typically reply in minutes")	None
Welcome Message	First message shown when a visitor opens the chat window, before typing anything	None
Input Placeholder	Hint text shown inside the message input field before the user types	Type your message...
Show Branding	When enabled, a "Powered by" attribution line appears at the widget bottom	Enabled

Click **"Save Settings"** when done. A confirmation notification will appear.

The screenshot shows the 'Widget Settings' panel for a widget identified as 'WDG-02'. The panel includes a 'Save Settings' button in the top right corner. The settings are as follows:

- Enable Widget:** Checked.
- Company Logo:** An 'Upload Logo' button is visible. A note below it says 'Recommended: Square image, max 2MB'.
- Position:** A dropdown menu is set to 'Bottom Right'.
- Button Color:** A color picker is set to a light blue color.
- Header Color:** A color picker is set to a light blue color.
- Widget Title:** A text input field contains 'Chat with us'.
- Widget Subtitle:** A text input field contains 'Optional'.
- Welcome Message:** A text input field contains 'Hi! How can I help you today?'.
- Input Placeholder:** A text input field contains 'Type your message...'.
- Show "Powered by" branding:** Checked.

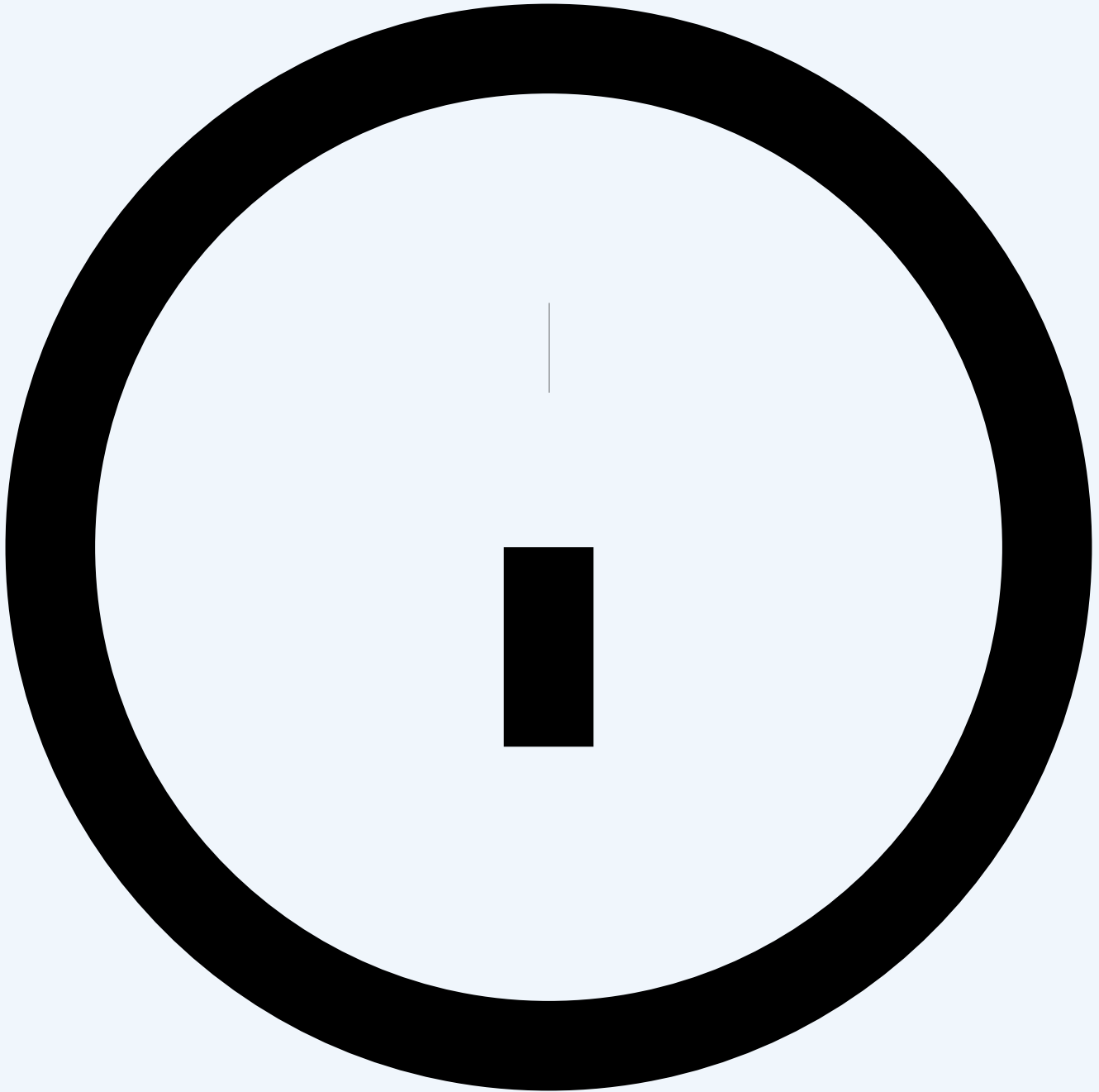
At the bottom of the panel, a summary line reads: 'The Widget Settings panel with all fields filled in, color pickers visible, and a logo preview displayed'.

STEP 03

Create a Deployment Token

Tokens are the security keys that authorize your widget to connect to the chatbot service. You need at least one token before deploying.

1. Click **"Add New Token"** in the tokens section.
2. Enter a **Token Name** — a human-readable label identifying where this token is used (e.g., *"Production Website"*, *"Staging Site"*).
3. Optionally enter **Allowed Domains** to restrict the token to specific websites. Leave empty for testing environments.
4. Click **"Create"** to generate the token.



Token Format The system generates a unique secure token in the format `wgt_XXXXXXXXXXXXXXXXXXXXXXXXXXXX`. Treat this like a password — do not share it publicly.

wgt_a1b2c3d4e5f6a1b2c3d4e5f6a1b2c3d4e5f6a1b2c3d4e5f6a1b2c3d4

Active

Token Name

Use descriptive labels per environment: [Production Website](#), [Staging Site](#), [Landing Page – Campaign Q4](#)

Allowed Domains

Example: [acmecorp.com](#), [www.acmecorp.com](#) — prevents unauthorized sites from using your token.

WDG-03

Access Tokens + Create Token

Create New Token

Token Name

Token Name

Production Website

Allowed Domains (optional)

example.com, subdomain.example.com (comma-separated)

Create

Leave empty to allow all domains

Create Cancel

The "Create Token" form with Token Name filled in as "Production Website" and Allowed Domains field showing domain restrictions

WDG-04

Access Tokens + Create Token

Token details

production site Active Disable

wgt_6eceb0b5db4c1647e4d0659e30a6d3848ab059d1ca16a8132ba45707d85b5fda

Allowed: acme.com

Used 0 times

Important: For the embed code to work, make sure to replace the widget.js URL with your actual deployed app URL (e.g., <https://yourdomain.com/widget.js>). The iframe embed will work immediately.

Embeddable Widget (Floating Button)

```
<!-- Add this script to your website -->
<script>
(function() {
  var script = document.createElement('script');
  script.src = 'https://zp1v56uxy8rdx5ypatb0ockcb9tr6a-oci3--5173--61636aac.local-credentialless.webcontainer-api.io/widget.js';
  script.setAttribute('data-token', 'wgt_6eceb0b5db4c1647e4d0659e30a6d3848ab059d1ca16a8132ba45707d85b5fda');
  script.async = true;
  document.head.appendChild(script);
})();
</script>
```

Copy Copy embed code

Standalone Chat Page

https://zp1v56uxy8rdx5ypatb0ockcb9tr6a-oci3--5173--61636aac.local-credentialless.webcontainer-api.io/widget?token=wgt_6eceb0b5db4c1647e4d0659e30a6d3848ab

Copy Copy URL

The token list after creation, showing the new token with its name, masked token value, Active badge, and action buttons

STEP 04

Choose Your Deployment Method

Once a token is created, four deployment methods are available. Choose the one that fits your use case.

METHOD A Floating Button
METHOD B Standalone Page
METHOD C iFrame Embed
METHOD D API Integration



Method A — Embeddable Widget (Floating Button)

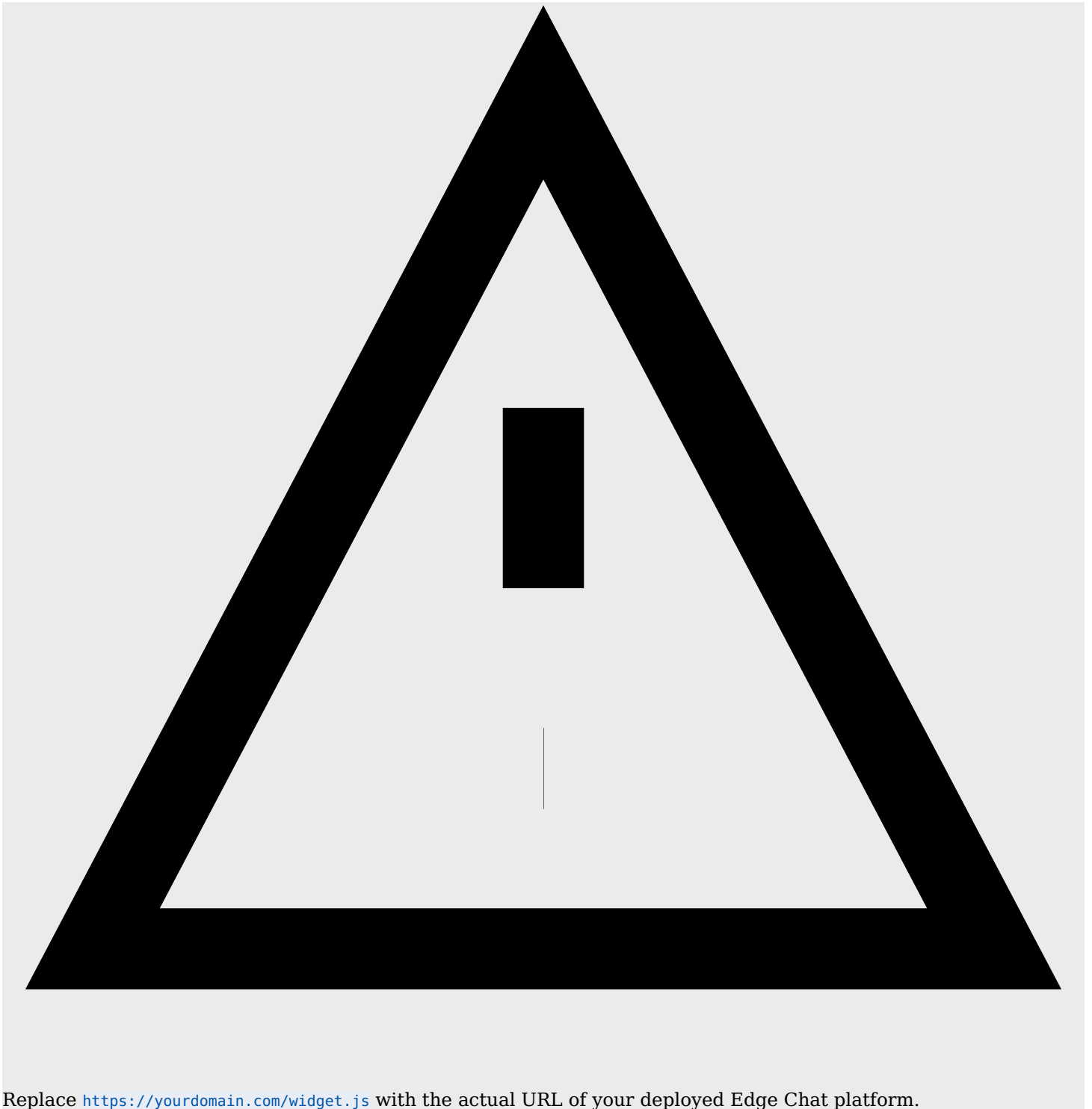
Best for: Adding a floating chat button to any existing website. A small JavaScript snippet loads the widget automatically.

Installation

1. Copy the code snippet shown for your token from the Deployment page.
2. Paste it into your website's HTML — either in the `<head>` section or just before the closing `</body>` tag.
3. The widget will appear automatically on page load.

HTML — Add to your website

```
<!-- Add this script to your website -->
<script>
  (function() {
    var script = document.createElement('script');
    script.src = 'https://yourdomain.com/widget.js';
    script.setAttribute('data-token', 'wgt_your_token_here');
    script.async = true;
    document.head.appendChild(script);
  })();
</script>
```



Replace <https://yourdomain.com/widget.js> with the actual URL of your deployed Edge Chat platform.



Method B — Standalone Chat Page

Best for: Sharing a direct link to a full-page chat experience via email, QR code, or a "Chat with us" button that opens a new tab.

 https://yourdomain.com/widget?token=wgt_your_token_here

How to Use It

- Share the URL directly with customers via email or messaging
- Link to it from a button on your website (opens in a new tab)
- Encode it in a QR code for physical marketing materials

<>

Method C — iFrame Embed

Best for: Embedding the chat interface directly inside a page layout, such as a dedicated "Contact Us" or "Support" page.

HTML — iFrame

```
<iframe
  src="https://yourdomain.com/widget?token=wgt_your_token_here"
  width="100%"
  height="600"
  frameborder="0"
  style="border: 1px solid #e5e7eb; border-radius: 8px;"
></iframe>
```

Customization Tips

- Adjust `height` to fit your layout: `500`, `700`, or `100vh`
- Set `width` to a fixed pixel value or `100%` for responsive layouts
- Remove the `border` and `border-radius` style attributes to match your own site's design



Method D — API Integration

Best for: Developers building a fully custom chat interface using their own frontend code.

JavaScript — API Request

```
const response = await fetch(
  'https://your-project.supabase.co/functions/v1/intelligent-chat',
  {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json',
      'x-widget-token': 'wgt_your_token_here'
    },
    body: JSON.stringify({
      message: 'Hello!',
      sessionId: null // pass a previous sessionId to continue a conversation
    })
  }
);

const data = await response.json();
console.log(data.message); // the AI's response text
console.log(data.sessionId); // save this to maintain conversation continuity
```

Key Details

- The endpoint is your Supabase project's Edge Function URL
- Always pass `x-widget-token` in the request header
- Store the `sessionId` from each response and pass it back to maintain conversation history
- A `sessionId` of `null` starts a new conversation

Managing Tokens

Each token appears in a list with its status, usage statistics, and management actions.

Field	Description
Name	The label you gave the token (e.g., "Production Website")
Token Value	The full <code>wgt_...</code> token string in monospace font
Status	<code>Active</code> or <code>Disabled</code>
Allowed Domains	Displayed if domain restrictions are set on the token
Usage Count	How many times this token has been used to initiate a chat
Last Used	The date the token was last used, or "Never"

Disabling a Token

Click **"Disable"** to immediately revoke access. Any widget using this token stops working. The token record is preserved — re-enable at any time.

Re-enabling a Token

Click **"Enable"** next to a disabled token to restore access immediately.

Best Practices

- Create **one token per deployment environment** (separate tokens for production and staging)
- Set **domain restrictions** on production tokens to prevent unauthorized use
- **Disable rather than delete** tokens when taking a deployment offline — preserves usage history
- For multiple websites on the same client, create a **separate token per site** for independent tracking

Troubleshooting

?

The widget is not appearing on my website.

- Confirm the script tag is correctly pasted in your site's HTML
- Check that the [widget.js](#) URL points to your actual deployed platform domain
- Verify the token is **Active** in the token list
- If domain restrictions are set, confirm the website's domain matches exactly

?

The widget loads but shows an error message.

- The token may be disabled — check the token status in Deployment
- The tenant may be set to Inactive — check the tenant settings
- If domain restrictions are set, confirm the requesting domain is in the allowed list

?

The chatbot is not answering questions correctly.

- Review the tenant's System Context and Response Guidelines in Tenant Management
- Ensure the knowledge base has relevant documents uploaded
- Try adjusting the AI model or response temperature in the tenant settings

?

I need to deploy on multiple websites for the same client.

- Create a separate token for each website — this lets you track usage per site and disable individual deployments independently without affecting the others