

# Sticky Elements Extension

Last Modified on 06/10/2026 11:03 pm EDT

## What you will learn

- How to install the Sticky Buttons Extension into your Digioh widget
- Which widget metadata keys you need to set to turn on sticky behavior

## About This Extension

This extension generates sticky navigation elements such as "Next", "Back", and "Continue" buttons. Its primary function is to render these elements outside the iframe, ensuring proper sticky behavior, which is not reliably supported within iframes.

## Installation

To take advantage of this feature, you can install the "Sticky Elements" extension within Digioh. Here's how to do it:

1. Access your profile menu in the top-right corner
2. Select "Extensions"
3. Find and install the "Sticky Elements" app from the Available section

## Configure Your Metadata

### Required Campaign-Level Metadata

To enable sticky buttons, add the following metadata key at the campaign level:

| Metadata Key          | Level    | Value        | Description  |
|-----------------------|----------|--------------|--|
| enable_sticky_feature | Campaign | true / false | Set to <code>true</code> to turn on the extension. <code>false</code> to turn off the extension. |

### Button-Level Metadata

Add the following to any button or element you want to make sticky:

| Metadata Key         | Level  | Value        | Description  |
|----------------------|--------|--------------|--|
| sticky_element       | Button | true / false | Set to <code>true</code> on any button or element you want to stick.   |
| sticky_element_order | Button | Number       | Defines the left-to-right display order of sticky elements. Lower values appear further left (1 = leftmost). |

Once installed and configured, your "Next", "Back", or "Continue" buttons will clone out of the iframe and stick to the bottom of the viewport, styled exactly as you like.

## Optional Styling Metadata

The sticky bar's visual appearance can be fully customized using the following optional widget-level metadata keys. If a key is not set, the default value is used automatically.

To remove a default style and let the parent page CSS take over instead, set the value to `{!{code class="bg-text-200/5 border border-0.5 border-border-300 text-danger-000 whitespace-pre-wrap rounded-[0.4rem] px-1 py-py-text-[0.9rem]"!};czo1OlwidW5zZXRcljt7WYyqJl19{!{/code}!};`.

| Metadata Key    | Level  | Value          | Default   |
|-----------------|--------|----------------|---|
| position        | Widget | CSS value      | fixed   |
| bottom          | Widget | CSS value      | 0   |
| left            | Widget | CSS value      | 0   |
| width           | Widget | CSS value      | 100%  |
| backgroundColor | Widget | CSS value      | <code>{!{code class="bg-text-200/5 border border-0.5 border-border-300 text-danger-000 whitespace-pre-wrap rounded-[0.4rem] px-1 py-py-text-[0.9rem] inline-flex items-center h-5"!};czo3Olwil2Y5ZjlmOVwiO3tbJiomXX0= {!{/code}!};</code> |
| padding         | Widget | CSS value      | 16px  |
| boxShadow       | Widget | CSS value      | 0 -2px 8px rgba(0,0,0,0.1)  |
| zIndex          | Widget | Integer string | 999999  |
| display         | Widget | CSS value      | display:flex  |
| justifyContent  | Widget | CSS value      | center  |
| alignItems      | Widget | CSS value      | center  |
| gap             | Widget | CSS value      | 12px  |

## Single Page Application (SPA) Support: URL Change Monitoring

On single-page application sites (React, Vue, Angular), the page URL can change without a full page reload. In these cases, the sticky bar may persist after the user navigates away from the campaign page.

The extension includes an optional account-level metadata key to handle this:

| Metadata Key           | Level   | Value        | Description   |
|------------------------|---------|--------------|---|
| enable_sticky_spa_func | Account | true / false | Enables URL change monitoring. When active, the sticky bar is automatically hidden if the page URL changes away from the campaign URL |

**How it works:**

- On page load, the current campaign URL is stored in `localStorage` under the key `campaignUrl`
- The URL is then checked every 500ms
- If the URL changes to one that differs from the stored campaign URL, the sticky bar is automatically hidden
- This prevents the sticky bar from persisting on non-campaign pages after SPA navigation

## Preview Mode Limitation

The Sticky Elements extension does not render inside Digioh Preview mode. Sticky elements will appear with `data-sticky="true"` in preview, which signals that the element is configured as sticky, but the actual sticky bar UI will not be rendered.

**To properly test sticky elements**, use your live or staging site with `data-sticky="true"` appended to the URL.

Additional behavior to be aware of when testing:

- Sticky elements are re-cloned on each page change within a multi-page campaign
- The `data-sticky="true"` value controls left-to-right display order, with lower values appearing furthest left