

FashionGo Widget Integration Guide for Magento

(Last Updated: 2024/10/07)

FASHIONGO

Table of Contents

FashionGo Widget Integration Guide for Magento.....	3
Method 1: Embedding Widgets Directly into Page/Block Content.....	3
Method 2: Adding Widgets to Predefined Locations (Sidebars/Footers) Using CMS Static Blocks.....	6
Method 3: Integrating the FashionGo Widget as a Pop-up	11

FashionGo Widget Integration Guide for Magento

This guide will walk you through integrating FashionGo widgets into your Magento store. The guide provides detailed steps for embedding the widget on specific pages, in predefined locations (like the sidebar or footer), and by using CMS blocks.

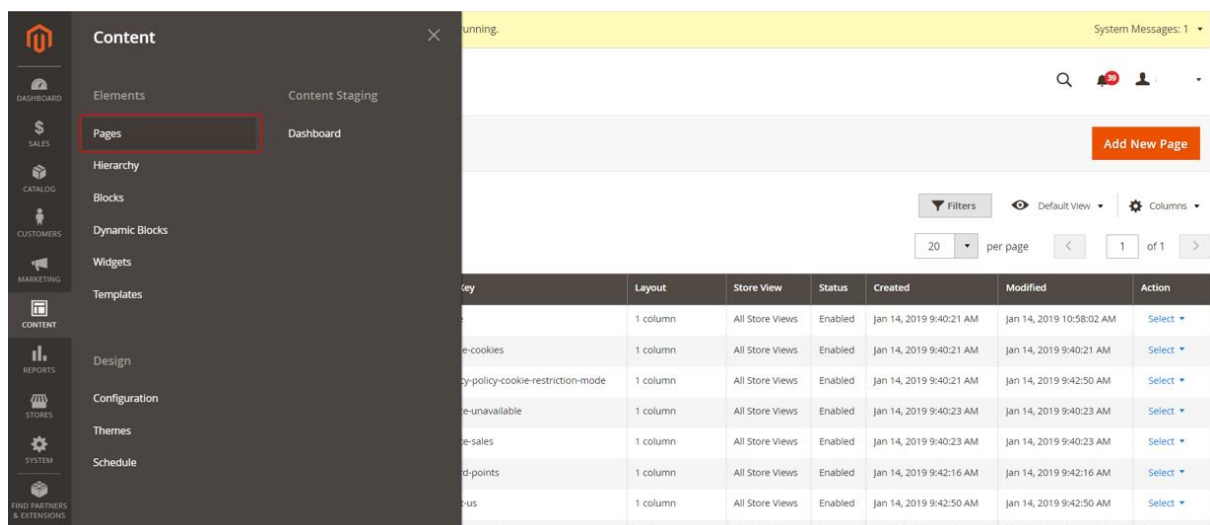
Method 1: Embedding Widgets Directly into Page/Block Content

This method is ideal if you want to add the FashionGo widget directly to specific pages, such as your homepage or any other custom content block.

Step-by-Step Guide:

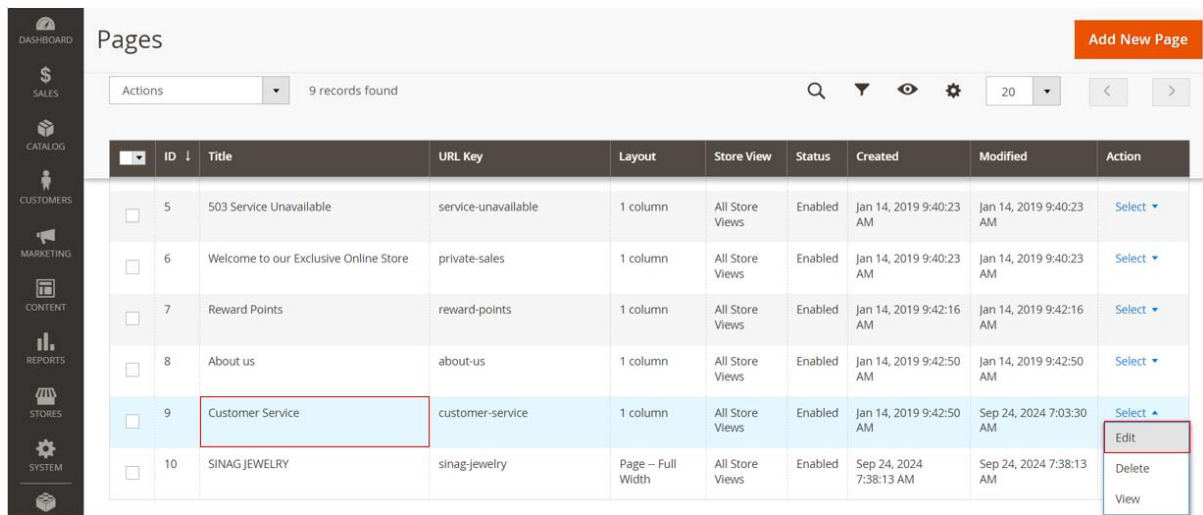
Step 1: Log in to your Magento admin panel.

Step 2: Navigate to Content > Pages.



Step 3: Edit a Page:

- Find the page where you want to add the widget (e.g., the homepage) and click Edit.
- You'll be taken to the page editor.



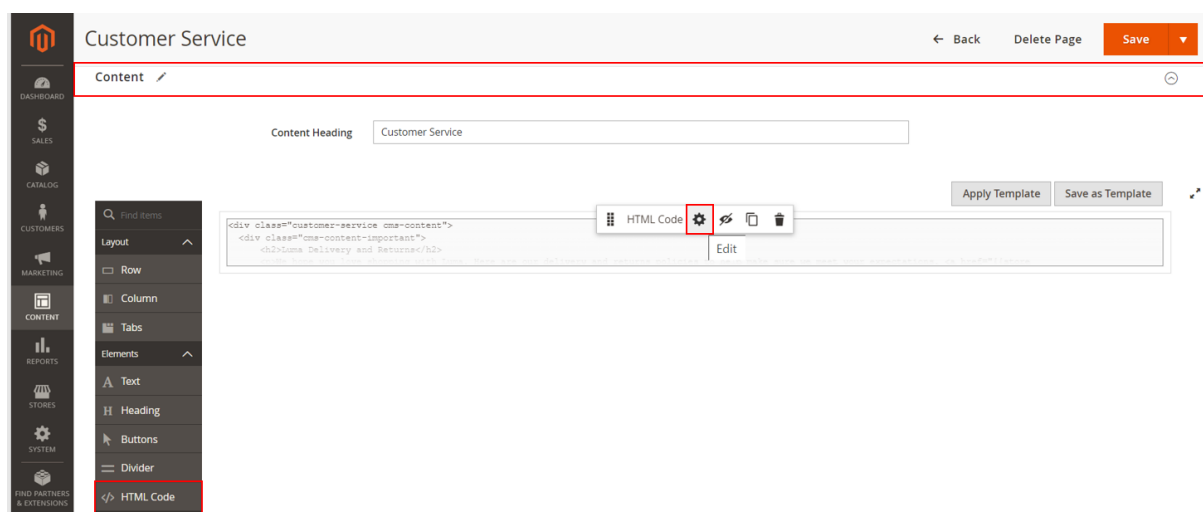
Pages Add New Page

Actions 9 records found

	ID	Title	URL Key	Layout	Store View	Status	Created	Modified	Action
☐	5	503 Service Unavailable	service-unavailable	1 column	All Store Views	Enabled	Jan 14, 2019 9:40:23 AM	Jan 14, 2019 9:40:23 AM	Select
☐	6	Welcome to our Exclusive Online Store	private-sales	1 column	All Store Views	Enabled	Jan 14, 2019 9:40:23 AM	Jan 14, 2019 9:40:23 AM	Select
☐	7	Reward Points	reward-points	1 column	All Store Views	Enabled	Jan 14, 2019 9:42:16 AM	Jan 14, 2019 9:42:16 AM	Select
☐	8	About us	about-us	1 column	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
☐	9	Customer Service	customer-service	1 column	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Sep 24, 2024 7:03:30 AM	Select Edit Delete View
☐	10	SINAG JEWELRY	sinag-jewelry	Page -- Full Width	All Store Views	Enabled	Sep 24, 2024 7:38:13 AM	Sep 24, 2024 7:38:13 AM	

Step 4: Access the HTML View:

- In the Page > Content section, find the left sidebar, locate Element, and click on </> HTML Code.
- Press Edit to switch to HTML mode and insert the widget code directly.



Customer Service Back Delete Page Save

Content

Content Heading: Customer Service

Apply Template Save as Template

HTML Code

```
<div class="customer-service cms-content">
<div class="cms-content-important">
<h2>Same Delivery and Returns</h2>
</div>
```

Step 5: Place the FashionGo Widget Code provided by Fashiongo portal Vendoradmin (standard, or button widget):

- Locate the area in the HTML where you want the widget to appear.
- Make sure the widget code is placed between the main content div or section tags that contain the content for that page.
- For example, if you have a section or div for the main content, place the widget code inside it:

```
<div class="main-content">
  <!-- Your other content here -->

  <!-- Paste your FashionGo Widget code here -->
  <widget-embed id="widget_id" ...></widget-embed>

  <!-- Additional content if necessary -->
</div>
```

- Ensure that the widget code is placed in a logical position within the HTML, so it appears where you want it on the page.

Step 6: Save the Page:

- Once you've placed the code, click Save in the top-right corner.

Step 7: Flush the Magento Cache:

- Go to System > Cache Management.

- Select all caches and click Flush Magento Cache to ensure the changes are applied to your store.

The screenshot shows the Magento Admin interface. On the left, the 'System' menu is expanded, and 'Cache Management' is highlighted. In the main content area, the 'Flush Cache Storage' button is visible, and the 'Flush Magento Cache' button is highlighted with a red box. Below this, a table lists various cache tags and their status.

	Tags	Status
	CONFIG	INVALIDATED
	LAYOUT_GENERAL_CACHE_TAG	ENABLED
	BLOCK_HTML	ENABLED
	COLLECTION_DATA	ENABLED
	REFLECTION	ENABLED
	DB_DDL	ENABLED
	COMPILED_CONFIG	ENABLED
	EAV	ENABLED

Step 8: Test:

Go to the page you just edited and verify that the widget is correctly displayed on the frontend.

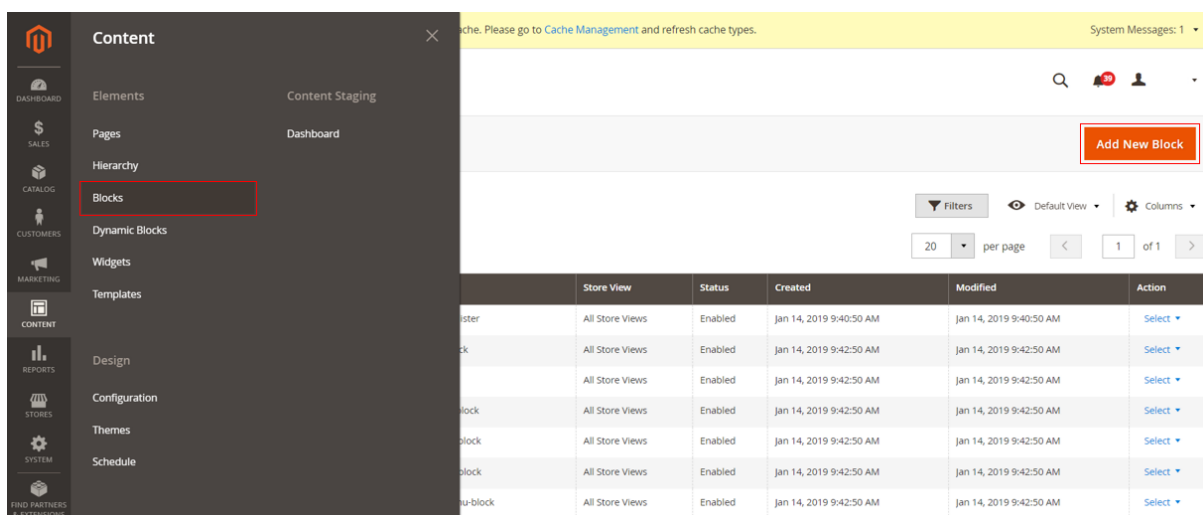
Method 2: Adding Widgets to Predefined Locations (Sidebars/Footers) Using CMS Static Blocks

This method allows you to place the FashionGo widget in predefined areas such as sidebars, footers, or headers using CMS Static Blocks. If the initial attempt didn't work, let's enhance the approach to ensure better functionality.

Step-by-Step Guide:

Step 1: Create a New CMS Static Block:

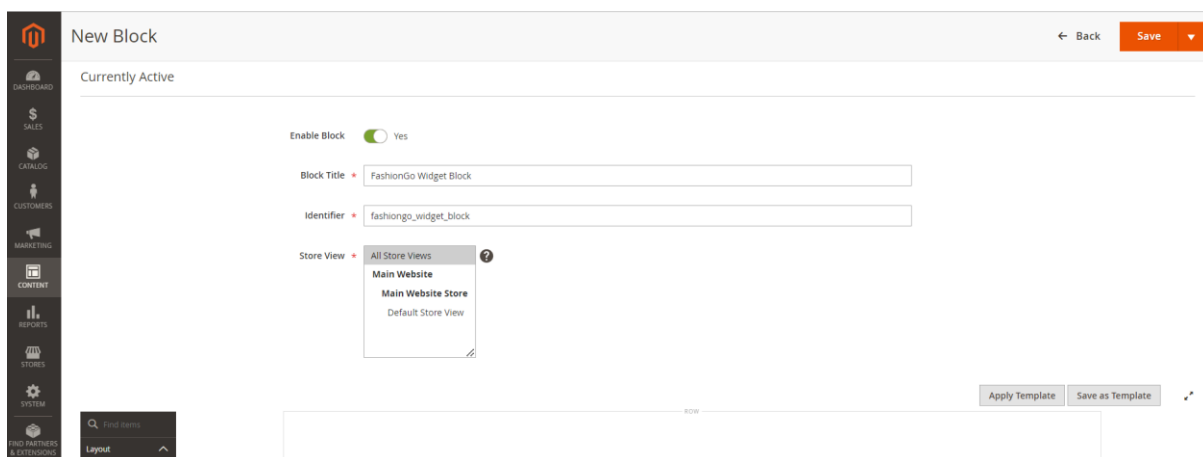
- Navigate to Content > Blocks.
- Click Add New Block.



The screenshot shows the Magento 2 admin interface. On the left, the 'Content' sidebar is expanded, and 'Blocks' is highlighted with a red rectangle. The main area displays a table of existing static blocks. A yellow banner at the top right says 'System Messages: 1'. A red 'Add New Block' button is visible in the top right corner of the main area.

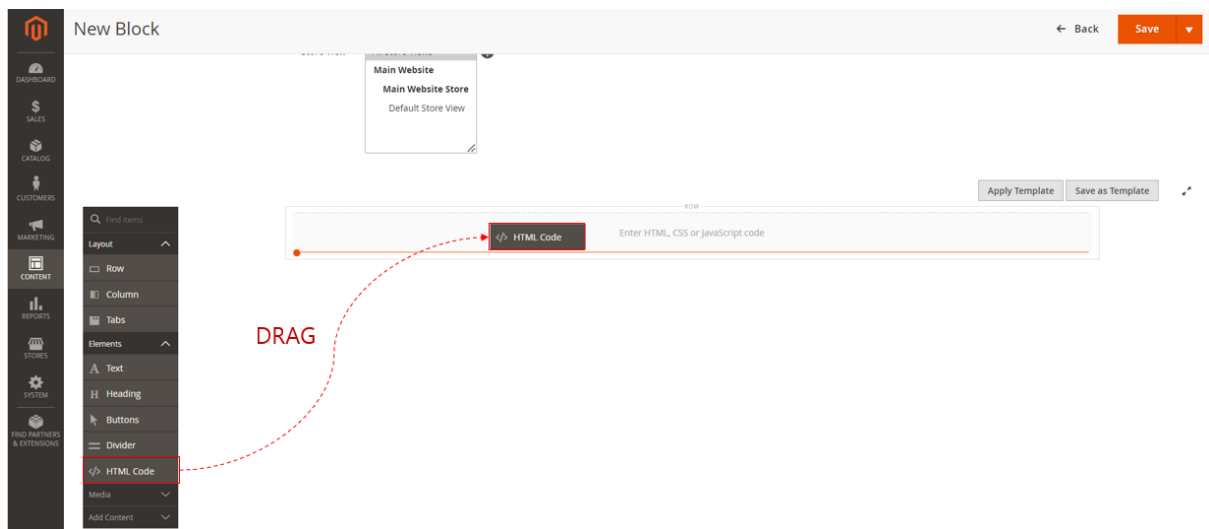
	Store View	Status	Created	Modified	Action
Block 1	All Store Views	Enabled	Jan 14, 2019 9:40:50 AM	Jan 14, 2019 9:40:50 AM	Select
Block 2	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
Block 3	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
Block 4	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
Block 5	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
Block 6	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select
Block 7	All Store Views	Enabled	Jan 14, 2019 9:42:50 AM	Jan 14, 2019 9:42:50 AM	Select

- Name the block (e.g., "FashionGo Widget Block") and provide an identifier (e.g., fashiongo_widget_block).



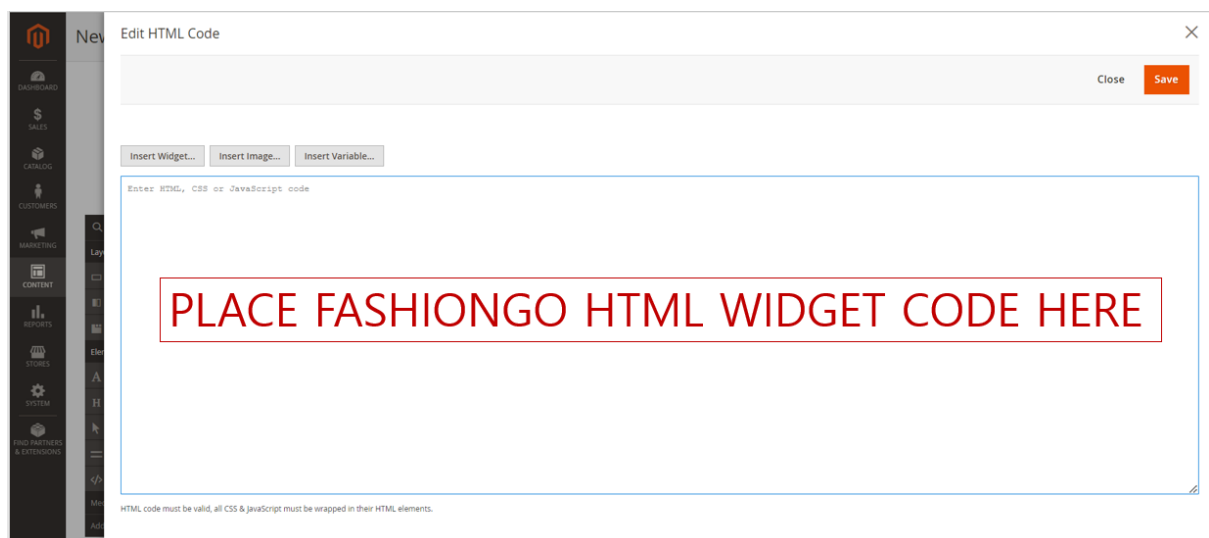
The screenshot shows the 'New Block' form in the Magento 2 admin interface. The 'Block Title' is 'FashionGo Widget Block' and the 'Identifier' is 'fashiongo_widget_block'. The 'Store View' dropdown is open, showing 'All Store Views', 'Main Website', 'Main Website Store', and 'Default Store View'. The 'Enable Block' toggle is set to 'Yes'. At the bottom, there are buttons for 'Apply Template', 'Save as Template', and a 'Layout' button.

- In the Content section, click and drag </> HTML Code button and click edit to switch to HTML mode.



Step 2: Place the Widget Code in the CMS Block:

- Paste the FashionGo widget code between the content div tags or sections where you want the widget to appear.
- Ensure the widget code is placed logically within the block to match your design intent.

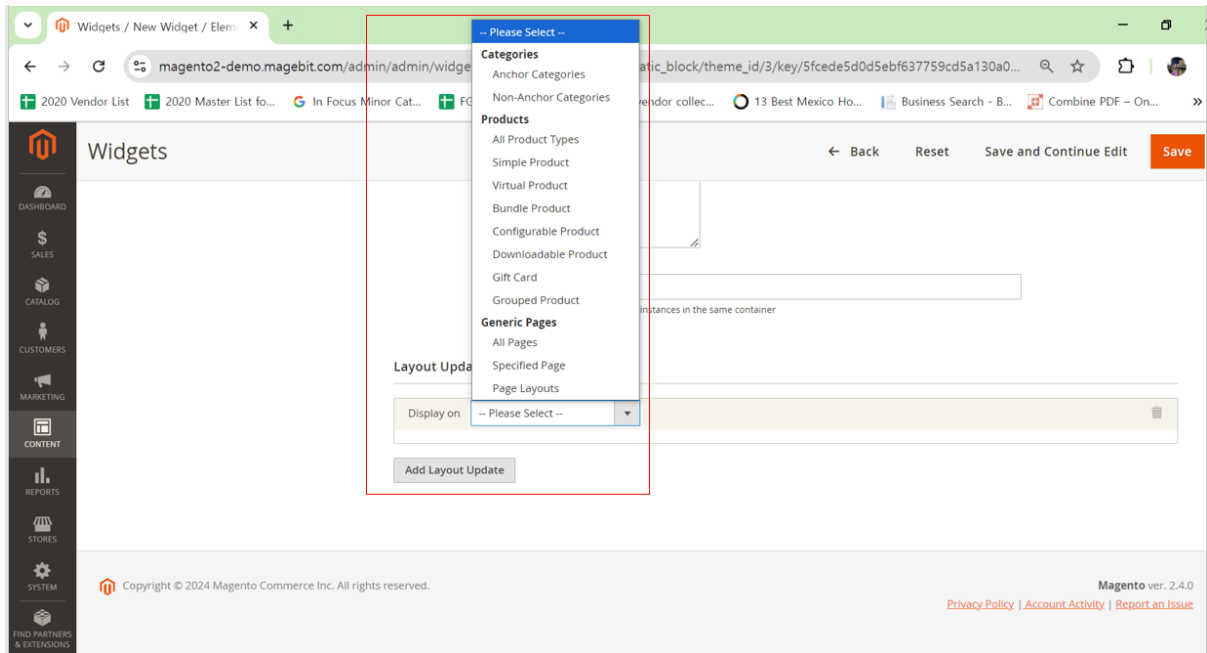


Step 3: Save the CMS Block:

- Once the code is pasted, click Save.

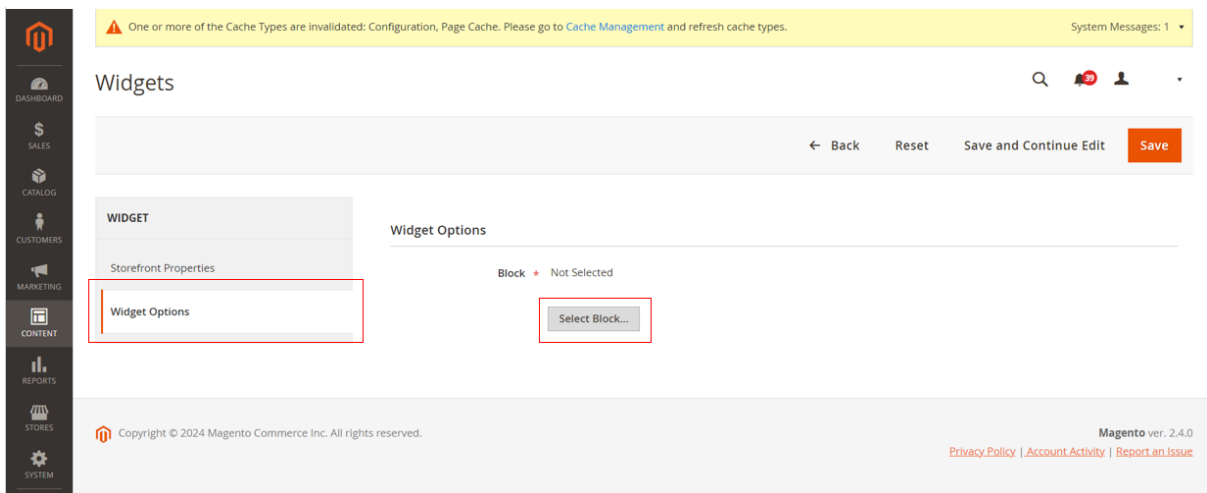
Step 4: Assign the Block to a Widget:

- Navigate to Content > Widgets.
- Click Add Widget.



Step 6: Link the CMS Block to the Widget:

- In the Widget Options tab, select the CMS Static Block you created earlier.



Step 7: Save the Widget:

- Click Save in the top-right corner of the page.

Step 8: Flush the Magento Cache:

- Go to System > Cache Management and flush all caches to ensure the widget appears on the frontend.

The screenshot displays the Magento 2 System Cache Management interface. On the left, the 'SYSTEM' menu is visible, with 'Cache Management' highlighted. The main content area shows a 'Flush Cache Storage' button and a table of cache tags. A red box highlights the 'Flush Magento Cache' button.

	Tags	Status
Invalidated	CONFIG	INVALIDATED
	LAYOUT_GENERAL_CACHE_TAG	ENABLED
	BLOCK_HTML	ENABLED
	COLLECTION_DATA	ENABLED
	REFLECTION	ENABLED
	DB_DDL	ENABLED
	COMPILED_CONFIG	ENABLED
	EAV	ENABLED

Step 9: Test:

- Visit the pages or sections where you added the widget (e.g., sidebar or footer) and verify that the widget is displayed correctly.

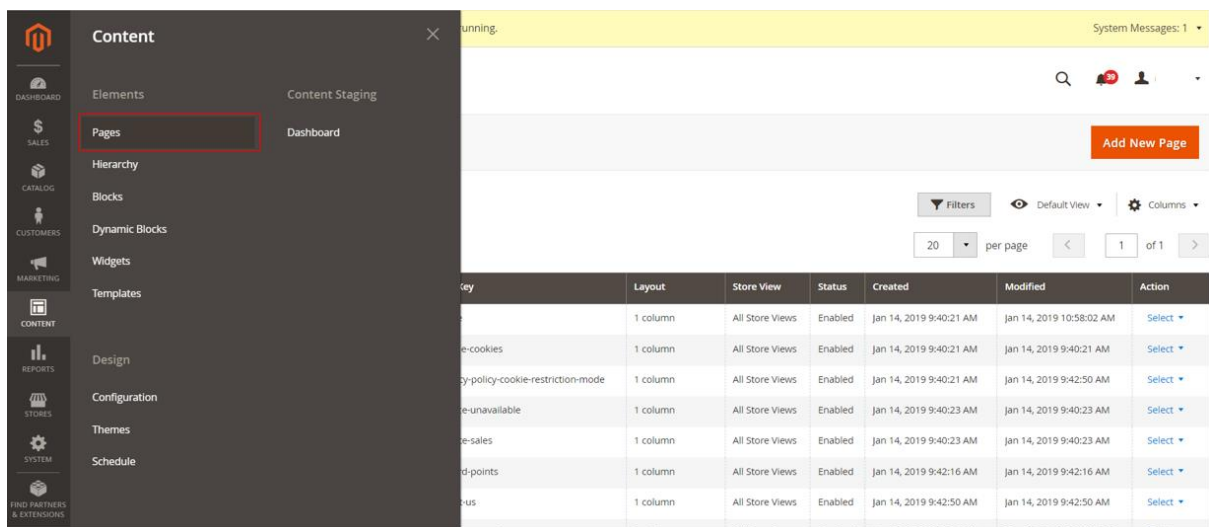
Method 3: Integrating the FashionGo Widget as a Pop-up

This method explains how to integrate the FashionGo widget into your Magento store as a pop-up. The pop-up can be triggered after a delay, when users scroll, or based on other actions.

Step-by-Step Guide:

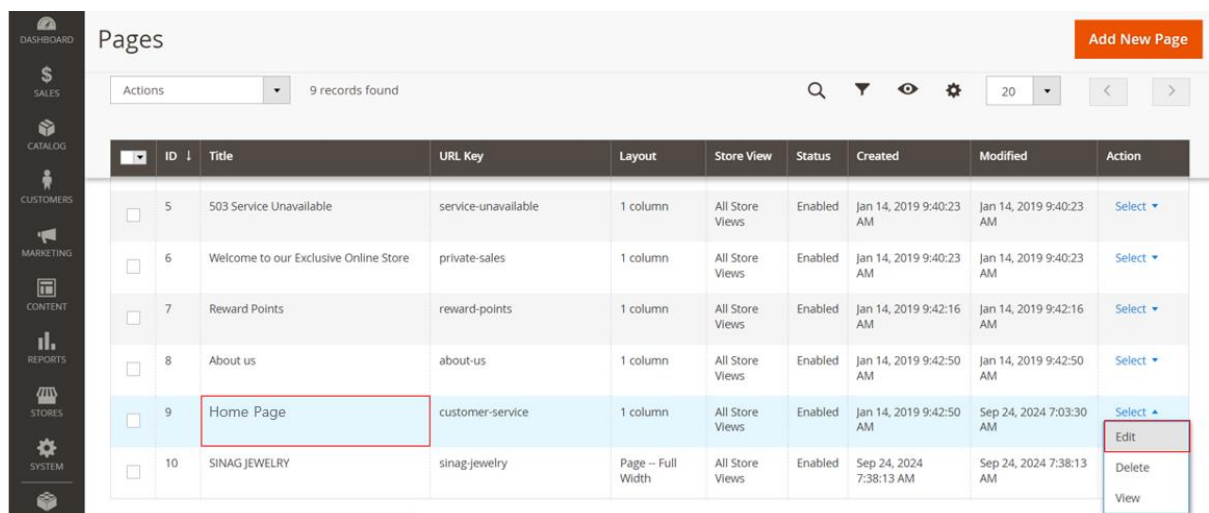
Step 1: Log in to your Magento admin panel.

Step 2: Navigate to Content > Pages.



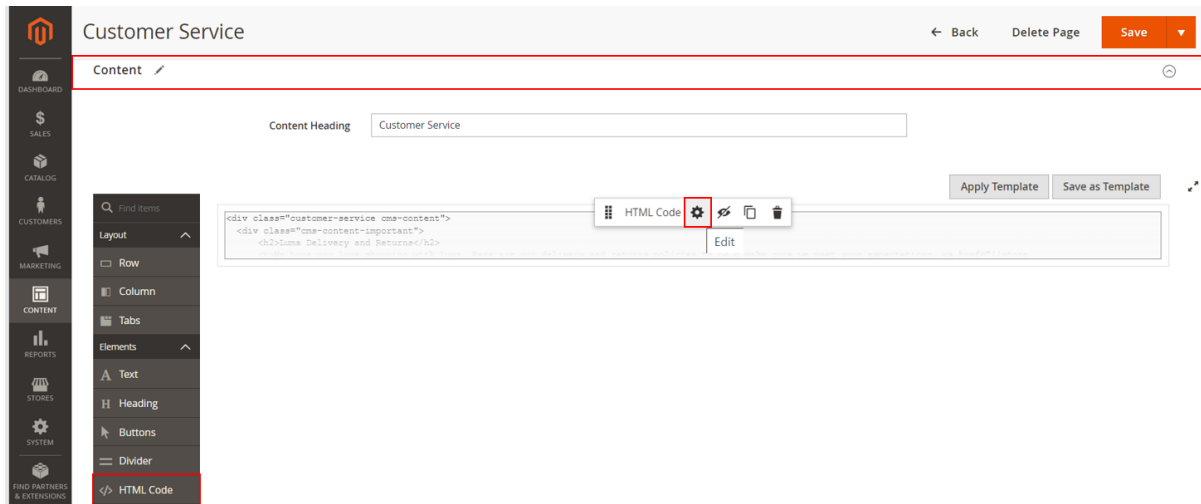
Step 3: Edit the Page:

- Find the page where you want the pop-up to appear (e.g., the homepage) and click Edit.



Step 4: Access the HTML View:

- In the Page > Content section, locate Element in the left sidebar and click on `</> HTML Code`.
- Press Edit to switch to HTML mode, where you'll insert the HTML, CSS, and JavaScript for the pop-up.



Step 5: Insert the Pop-up HTML Structure:

Place the following HTML code at the bottom of the page's content, just before the closing `</body>` tag. This will create the structure for the pop-up window:

```
<!-- Pop-up Structure -->
<div id="fashiongo-popup" class="popup-overlay" style="display:none;">
  <div class="popup-content">
    <span class="popup-close" id="popup-close">&times;</span>
    <!-- Place your FashionGo Widget code here -->
    <widget-embed id="widget_id" ...></widget-embed>
  </div>
</div>
```

- The **widget code** should be placed inside the pop-up content div.

Step 6: Add CSS for Pop-up Styling:

In the Page > Content section, add the following CSS in the `<head>` section to style the pop-up:

```
<style>
.popup-overlay {
  position: fixed;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background: rgba(0, 0, 0, 0.5);
  display: flex;
  justify-content: center;
  align-items: center;
  z-index: 1000;
}
```

```

}

.popup-content {
  background: white;
  padding: 20px;
  border-radius: 8px;
  width: 80%;
  max-width: 500px;
  position: relative;
}

.popup-close {
  position: absolute;
  top: 10px;
  right: 10px;
  font-size: 20px;
  cursor: pointer;
}
</style>

```

Step 7: Add JavaScript to Control the Pop-up Behavior:

Below the pop-up HTML (before the closing `</body>` tag), add this JavaScript to control when the pop-up appears:

```

<script>
  // Show the pop-up after 5 seconds
  setTimeout(function() {
    document.getElementById('fashiongo-popup').style.display = 'flex';
  }, 5000); // 5000 milliseconds = 5 seconds delay

  // Close the pop-up when the close button is clicked
  document.getElementById('popup-close').onclick = function() {
    document.getElementById('fashiongo-popup').style.display = 'none';
  };
</script>

```

- This script triggers the pop-up to appear after 5 seconds and allows users to close it by clicking the close button.

Step 8: Save the Page:

Once you've inserted the HTML, CSS, and JavaScript, click Save in the top-right corner of the page editor.

Step 9: Flush the Magento Cache:

- Go to System > Cache Management and flush all caches to ensure the widget appears on the frontend.

The screenshot shows the Magento 2 System Cache Management interface. The left sidebar contains a menu with the following items: DASHBOARD, SALES, CATALOG, CUSTOMERS, MARKETING, CONTENT, REPORTS, STORES, and SYSTEM. The SYSTEM menu item is highlighted. The main content area displays a table of cache types with the following columns: Name, Tags, and Status. The table contains the following data:

Name	Tags	Status
Invalidated	CONFIG	INVALIDATED
	LAYOUT_GENERAL_CACHE_TAG	ENABLED
	BLOCK_HTML	ENABLED
	COLLECTION_DATA	ENABLED
	REFLECTION	ENABLED
	DB_DDL	ENABLED
	COMPILED_CONFIG	ENABLED
	EAV	ENABLED

At the top of the main content area, there is a yellow banner with the text "Refresh cache types." and a "System Messages: 1" dropdown. Below the banner, there is a "Flush Cache Storage" button and a "Flush Magento Cache" button, which is highlighted in a red box.

Step 10: Test:

- Visit the page you just edited (e.g., the homepage) and verify that the FashionGo widget appears as a pop-up after the 5-second delay.