



Site Search & Enhanced CMS Workspaces (Beta)

Salesforce, Summer '22



@salesforcedocs

Last updated: Jun 30, 2022



Important: This feature is a Beta Service. Customer may opt to try such Beta Service in its sole discretion. Any use of the Beta Service is subject to the applicable Beta Services Terms provided at [Agreements and Terms](#).

You can provide feedback and suggestions for the beta features in the [Lightning Web Runtime group in the Trailblazer Community](#).

CONTENTS

Overview	4
Before You Begin	4
Beta Limitations	4
How to Set Up the Beta Features	5
1. Enable the Beta Features	6
2. Create an LWR Site	6
Enable Guest Users to View CMS Content and Public APIs	7
3. Configure Your Enhanced CMS Workspaces	8
Create an Enhanced CMS Workspace	8
Create and Share Content in Your Enhanced CMS Workspace	9
Use Sidebar Extensions in your Content Editor	9
4. Add Content to Your Site	10
5. Configure Site Search	12
Rebuild the Search Index	12
Set Up the Search Bar Component	13
Set Up the Search Results Component	13
6. Track User Engagement on Your Site	15
Appendix A: experience/cmsEditorApi Wire Adaptors and Functions	17
getContext	17
getContent	17
updateContent	18
Appendix B: Connect REST APIs and Connect in Apex Methods	19

Overview

With improved site search, visitors can look for content contained in components in Build Your Own (LWR) sites. And with enhanced Salesforce CMS workspaces, you can create, manage, and publish content more easily.

Behind the scenes, a completely re-architected platform powers these beta features. After you enable the beta, the Build Your Own (LWR) sites and enhanced CMS workspaces that you create are hosted together on our flexible, unified backend.

Previously, site visitors used the Global Search component to find CRM, Knowledge, and CMS content records. But content contained in components on the page wasn't included in the search results.

Now, improved site search lets visitors look through text contained within page components, such as the Rich Content Editor and HTML Editor components, along with page title metadata. And because everything runs together on our unified backend, site search can also include content that you share from an enhanced CMS workspace in the site search results.

Before You Begin

To participate in this beta, you need:

- A Developer Edition org or a sandbox org in Enterprise, Performance, or Unlimited editions with Digital Experiences enabled
- Some experience in:
 - Building sites with Experience Builder
 - Creating content with CMS workspaces

Beta Limitations

Before you begin using the beta features, keep the current limitations in mind.

- The beta features aren't available in production orgs, nor can you deploy to production.
- After you enable the beta features, you can access improved site search and share content from enhanced workspaces only with Build Your Own (LWR) sites created after you enable the beta features
- Multilingual support is unavailable in new LWR sites created after you enable the beta features. If you create a new LWR site when the beta is disabled, multilingual features are available.
- Static page content that's indexed by site search is limited to the Rich Content Editor and HTML Editor components, as well as the page title metadata.
- Enhanced CMS workspaces don't yet support multilingual content, version control, history, scheduling, or rollback capabilities.
- If an Enhanced CMS workspace contains published content, and you subsequently add a Build Your Own (LWR) site as a new channel, you must republish the content for it to appear in the site.

How to Set Up the Beta Features

To try out the new beta features, follow these key steps. These steps cover only the areas required to set up and test the features successfully, but you can add other elements as needed.

1. [Enable the beta features](#) in your org.
2. [Create a new LWR site](#) using the Build Your Own (LWR) template.
3. [Create an enhanced CMS workspace](#), and create and share CMS content.
4. [Add content to your site](#).
5. [Configure site search](#).
6. Optionally, [track user engagement](#) in your site.

1. Enable the Beta Features

To enable the beta feature in your Developer Edition or sandbox org:

1. From Setup, in the Quick Find box, enter *Digital Experiences*, and then click **Settings**.
2. Select **Turn on beta features**, and save your changes.
3. By default, System Administrators can configure the features after they're enabled. For other users, ensure that they have the following user permissions:

USER PERMISSIONS

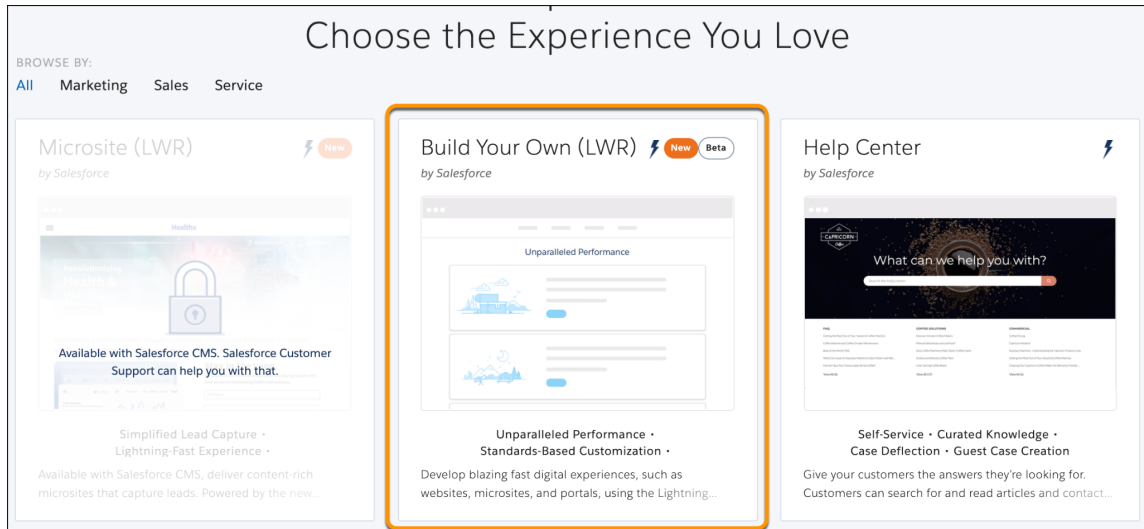
To create an Experience Cloud site:	"Create and Set Up Experiences"
To create workspaces and channels in the Salesforce CMS app:	"Modify All Data" OR "Create CMS Workspaces and Channels"
To manage content, contributors, and channels within a Salesforce CMS workspace:	<ul style="list-style-type: none">• Assigned a content admin role in that CMS workspace• AND, if adding Experience Cloud site channels, "Create and Set Up Experiences"
To rebuild the site search index:	"Create CMS Workspaces and Channels"

2. Create an LWR Site

You can access improved site search and share content from enhanced workspaces only with Build Your Own (LWR) sites created after you enable the beta features. When you create a new site, it's hosted on the new backend platform.

- To create a site, either:
 - In Setup, under **Digital Experiences | All Sites**, click **New**.
 - In the App Launcher, select the **Digital Experiences** app ([formerly called the Salesforce CMS app](#)). On the Digital Experiences Home tab, in the All Sites section, click **New**.

- Select the **Build Your Own (LWR)** template and click **Get Started**.



Tip:

- The Beta icon beside the template indicates that the beta features are enabled.
 - Consider including “Beta” in the name of your site to make it easier to identify while you test out the new features.
- Enter a name and base URL value.
The base URL value is appended to the domain that you created when you enabled Digital Experiences. For example, if your domain name is UniversalTelco.my.site.com and you’re creating a partner site, you can enter *partners* to create a unique URL UniversalTelco.my.site.com/partners.
- Choose an authentication option. Select:
 - **Unauthenticated** to build a public site that’s open to anyone on the web, which doesn’t support login or authentication. The link doesn’t include /s at the end of the base URL.
 - **Authenticated** to build an authenticated site, where users can log in and access user-specific data. The link includes /s at the end of the base URL.
- Click **Create**.

After you create your site, the Experience Workspaces area appears.

Enable Guest Users to View CMS Content and Public APIs

If you create an unauthenticated site or make pages in an authenticated site public, make sure that you enable the following settings for guest users.

In the Experience Workspace for your site, navigate to **Administration | Preferences**, select the checkboxes, and save your changes.

- **Allow guest users to access public APIs**
- **Let guest users view asset files and CMS content available to the site**

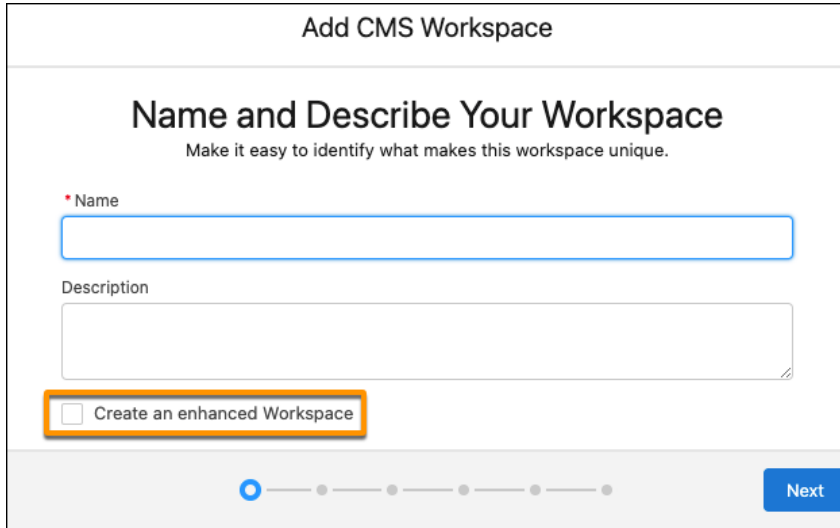
3. Configure Your Enhanced CMS Workspaces

Enhanced CMS workspaces are completely re-architected from the ground up on a powerful, unified backend. A CMS workspace is where you create, manage, and publish content that you can then share across channels—in this case, your new Build Your Own (LWR) site.

Create an Enhanced CMS Workspace

To create an enhanced workspace:

1. Open Digital Experiences from the App Launcher.
2. On the Digital Experiences Home tab, click **Add Workspace** and provide a name and description.
3. Select **Create an enhanced Workspace** and click **Next**.



4. Add your newly created LWR site as a channel to the workspace.
5. Select contributors from available users and public groups, click **Next**, and then assign them a role.

Content admin

These contributors have access to all content in the CMS workspace and can manage contributors and content sharing.

Content manager

These contributors have full access to all content in the CMS workspace.

6. Click **Next** and then **Done**.



Note: Enhanced CMS workspaces are noted with “Enhanced Workspace” in Summer ’22.

Digital Experiences Home Summer 2022 Campaign Enhanced Workspace		
Description	Created By Test User	Created Date 5/11/2022, 3:19 PM

Create and Share Content in Your Enhanced CMS Workspace

Content is unique to each enhanced CMS workspace and is published to the channels in that workspace—in this case, the new Build Your Own (LWR) site that you created.

In the content editor, you can use rich text options to apply paragraph heading styles, format text, and add CMS images, video URLs, and hyperlinks to the content body, and other rich text fields. And you can use the <iframe> tag and all formatting tags supported by [standard Salesforce rich text editors](#), along with several new attributes: allow, allowfullscreen, frameborder, and scrolling.


To create a news content type:

1. In your new enhanced CMS workspace, click **Add Content**.
2. Select the **News** content type and click **Create**.
The LWR site that you created contains a default News Detail page, which connects to this content type.
3. In the content editor, enter your content and then click **Save**, which saves your content as a draft.
4. When you're ready to share your finished content with the workspace's channels, click **Publish**.
Until you publish the content, it's unavailable to your site channel.

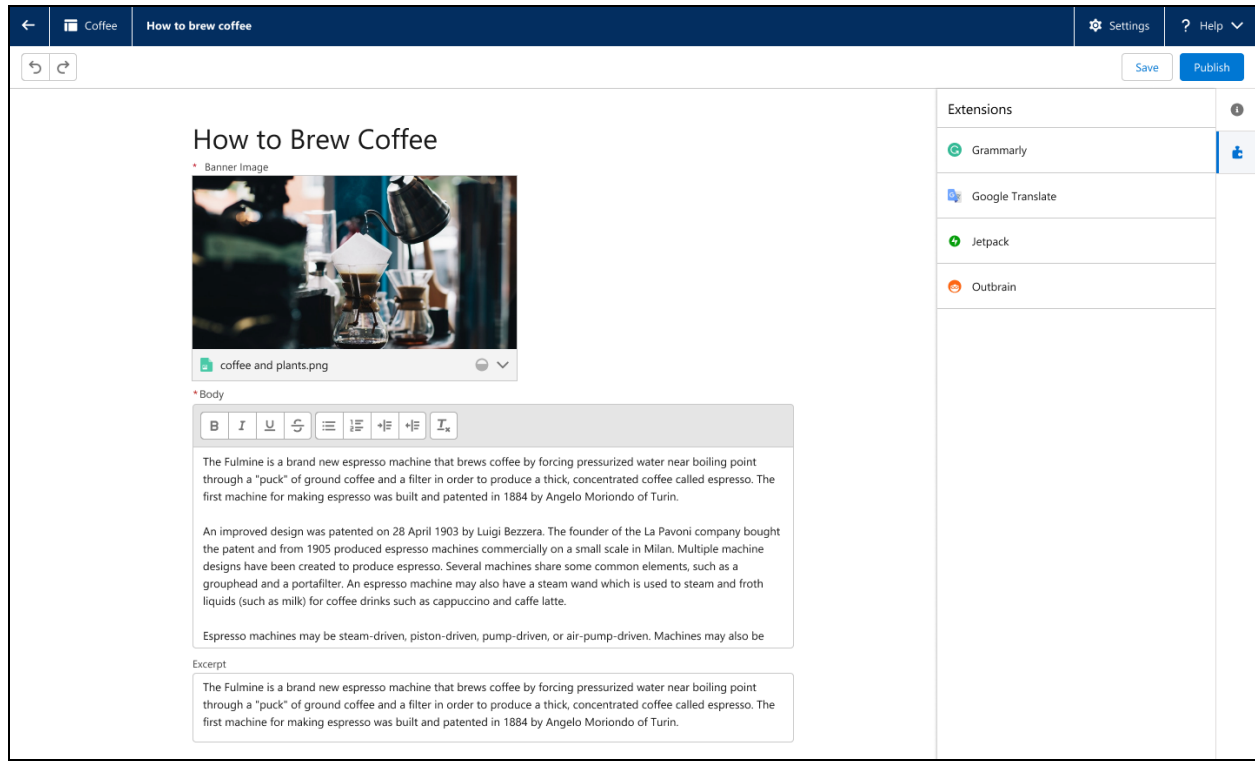
Use Sidebar Extensions in your Content Editor

Put productivity and authoring tools right into the content editor. Use content editor extensions to implement features like spell checkers, grammar and tone editors, content recommendation apps, and translation services..

The extensions panel is visible on the right side of the content editor after you've added at least one extension. You can add as many extensions as your team can use. Content authors can expand and collapse the panel itself, and the extensions within the panel.

1. Open Salesforce CMS from the Salesforce App Launcher, then open the CMS Enhanced Workspace where the content was created.
2. Click the row-level action dropdown menu of the content and select Edit. The content editor opens.
3. Click the extensions icon  on the right side of the CMS content editor.

4. Click Extension to view the list of available extensions.
5. Click the name of an extension to open it in the panel.



Editor sidebar extensions have read and write access to content item properties. Developers who can create and deploy lightning web components can write custom content editor extensions and load them directly into their CMS. You can also search AppExchange for extensions created by Salesforce partners.

4. Add Content to Your Site

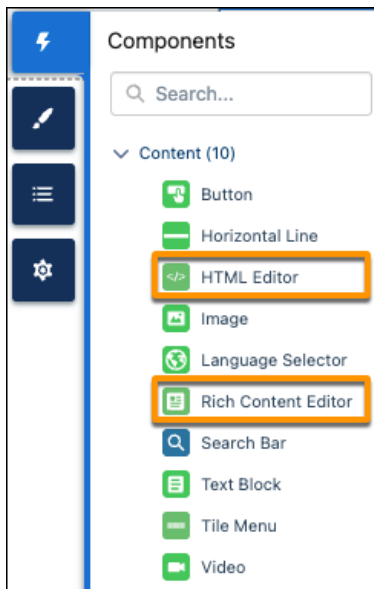
To test out the improved site search capabilities, it's time to add content to your site! Currently, site search returns results for text contained in the following items on the Home page or any standard page that you create.


- Rich Content Editor component
- HTML Editor component
- Page title metadata

To add searchable text to your site:

1. In Experience Builder, open the Pages menu on the top toolbar and then click **New Page** at the bottom of the menu.
2. Select the **Standard Page** type.
3. Click **New Blank Page**.

4. Enter a name for the page, and click **Create**.
5. Drag a Rich Content Editor or HTML Editor component onto the Content section of the page, and add some text.

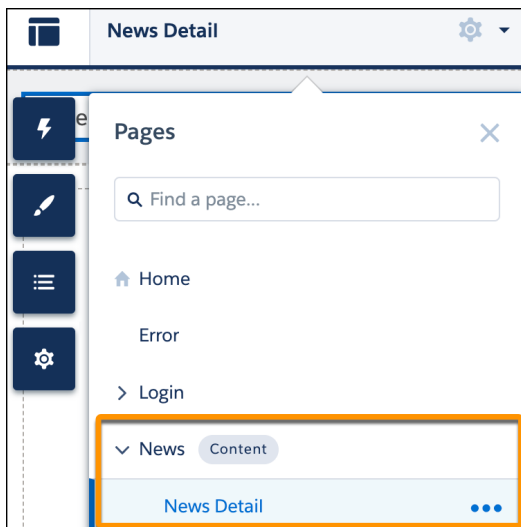


 **Tip:** Content in the Header or Footer sections isn't currently included in search results.

6. Publish your site updates.
Only published content can be included in the search index and returned in search results.

Site search can also return results from the enhanced CMS workspace that's connected to your site for published content items—in this case, the News item that you added.

By default, your new LWR site includes a blank News Detail page that connects to the News content type.



You can add components to the News Detail page to display fields from the News content item. However, for site search, the contents of page components aren't included in any results. Instead, site search returns results directly from the published content items in the enhanced CMS workspace.

If you add other CMS content types to your enhanced CMS workspace, remember to create a corresponding CMS Content page in your site. For example, if you add an Image content type in the CMS workspace, then go to Experience Builder, click **Pages | New Page**, select **CMS Content Page**, and select **Image**.



Tip: You can also add CMS content to your site by binding it to components, such as the Card and Overlap components. Although content contained in these components isn't yet included in site search results, the components highlight how to dynamically include CMS content in your site. Check out the [Did You Know: Content Binding Components video](#) to learn more.

5. Configure Site Search

The new site search capability includes two new components available in the Build Your Own (LWR) template.

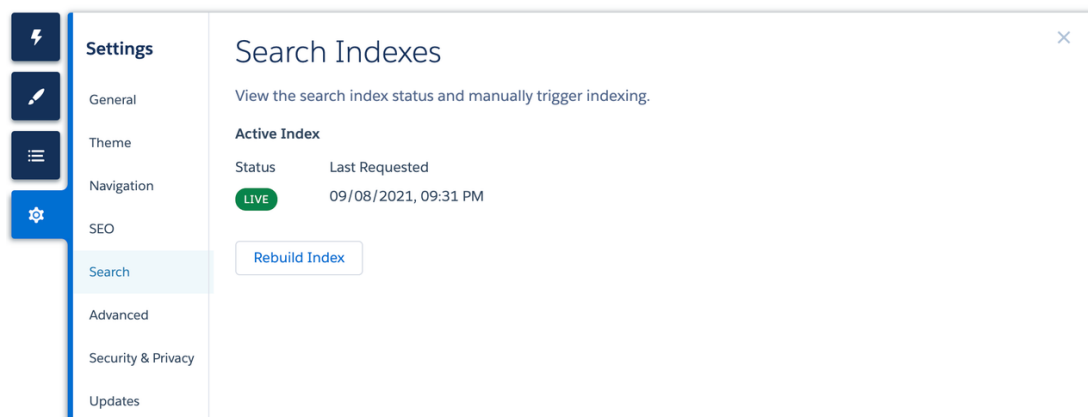
- The **Search Bar** component lets visitors enter search terms and can be added to any page or region.
- The **Results Layout** component displays search results. It's available only for standard Search pages.

Additionally, site search provides the ability to rebuild the search index whenever updates are published to content or to pages on your site.

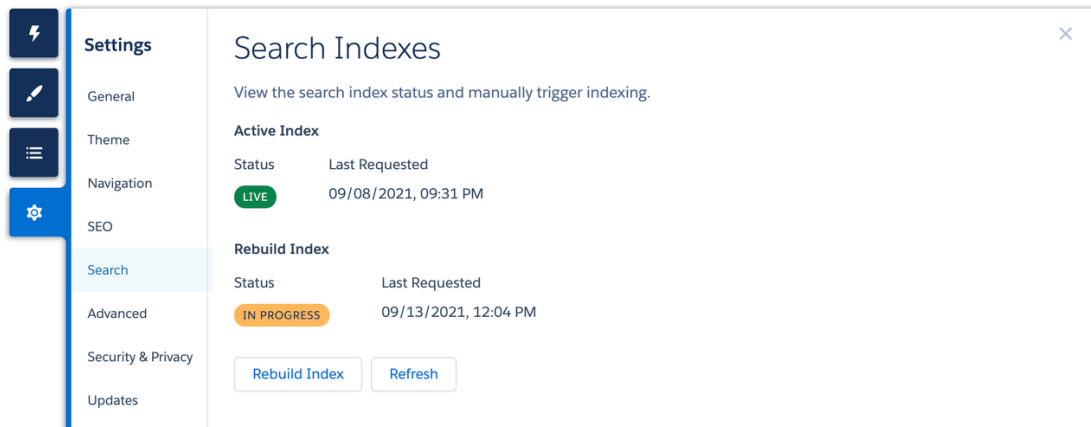
Rebuild the Search Index

Site visitors can search only indexed content, so remember to rebuild the index after you publish fresh content in your site or in the connected CMS workspace. Otherwise, the new content can't appear in the search results.

1. In Experience Builder, navigate to **Settings | Search**.



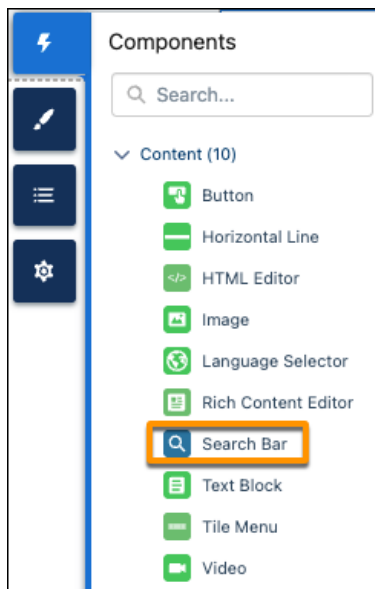
2. Click **Rebuild Index**.



Alternatively, you can trigger search indexing or get search indexes for a channel with [Connect REST API](#).

Set Up the Search Bar Component

1. In Experience Builder, open any page.
2. Drag the **Search Bar** component from the Components panel onto the header section of your site. Adding the component to the header section makes site search available across all site pages.

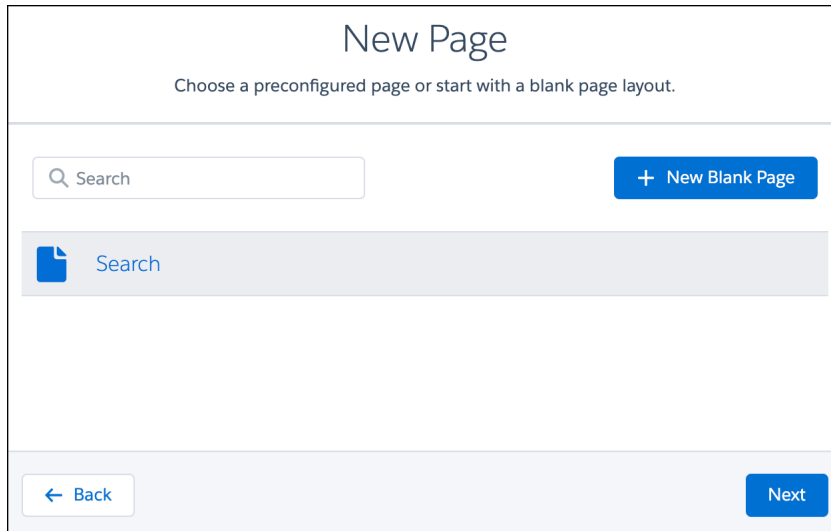


Set Up the Search Results Component

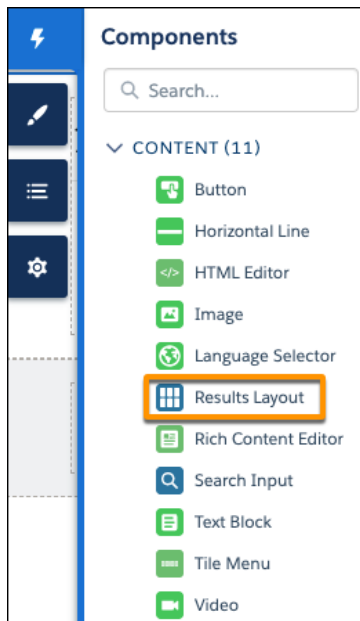
Search results can only be shown on a standard search page, so to use the Results Layout component, you must first create the search page.


1. In Experience Builder, open the Pages menu on the top toolbar and then click **New Page** at the bottom of the menu.

2. Select the **Standard Page** type.
3. Select **Search** and click **Next**.

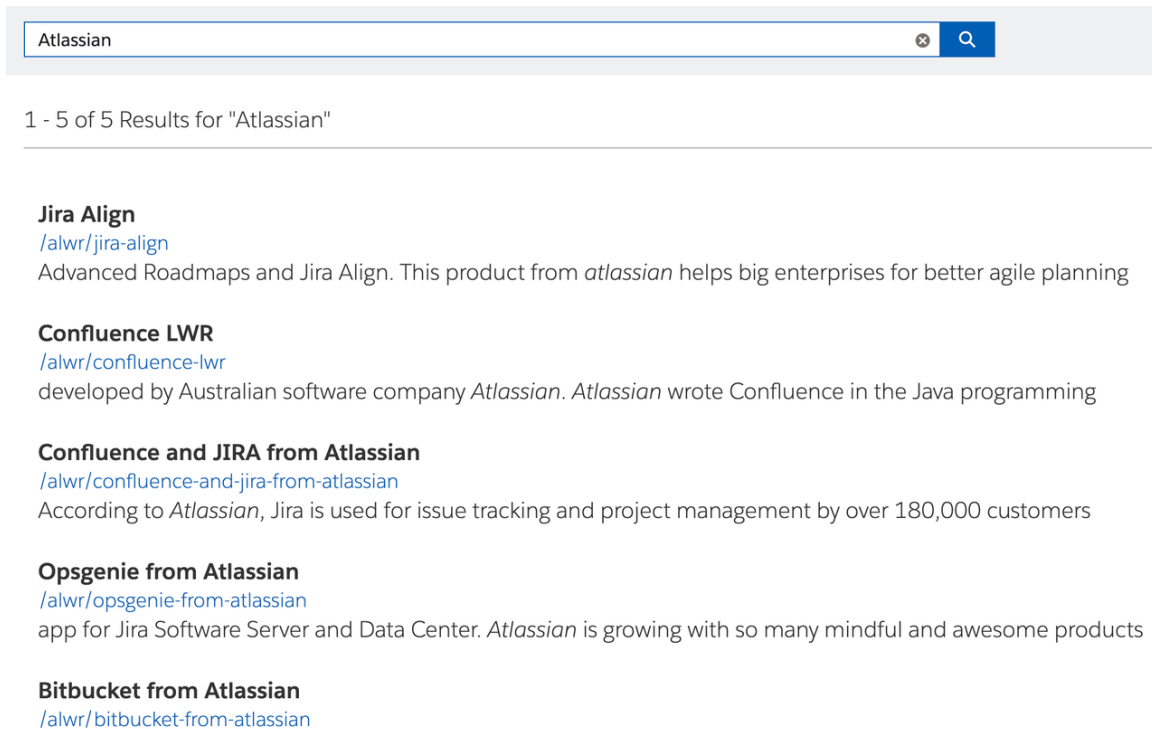


4. Enter a name for the page, such as *Search Results*, and click **Create**.
5. Drag the **Results Layout** component from the Components panel onto the new Search Results page.



6. To test out the site search:
 - a. Click **Preview**, then click , and preview the site as an authenticated or guest user.
 - b. Enter some appropriate text in the Search Bar field and then revel in the returned results!
7. Publish your site updates.

After you set up the site search components and then rebuild the search index, site visitors can search for published content on your site using the Search Bar component in your site's header, and this takes them to the search results page as shown below.



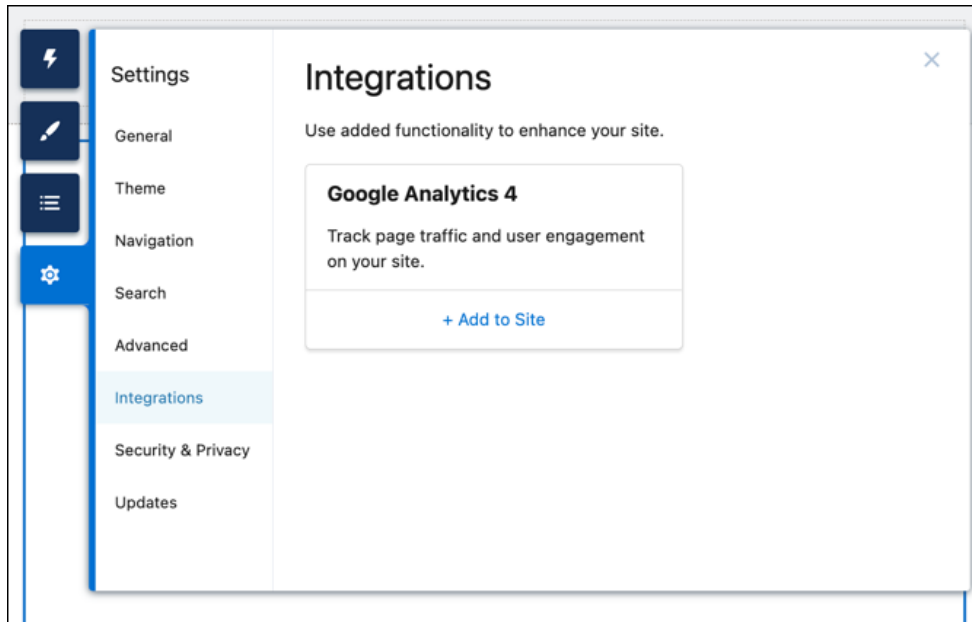
Alternatively, you can search your site with [Connect REST API](#) or [Connect in Apex](#).

6. Track User Engagement on Your Site

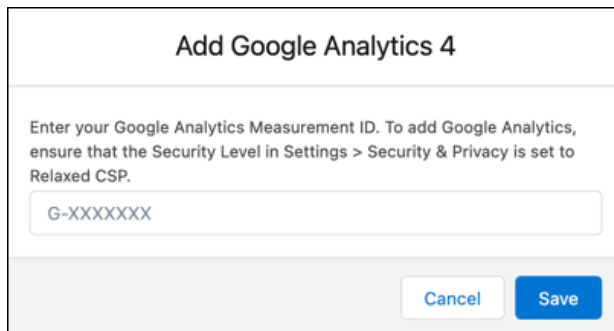
For insights into page visits and user engagement, add Google Analytics 4 to your site from Experience Builder. This feature is available only in orgs created in Summer '22 or later.

If you don't have a Google Analytics account, refer to the [GA4 documentation](#) from Google to set one up.

1. In Experience Builder, go to **Settings | Integrations**.



2. For Google Analytics 4, click **Add to Site**.
3. Enter your Google Analytics [Measurement ID](#), and save the change.



3. Publish the site. Google Analytics begins tracking user engagement after you publish.

To stop Google Analytics from tracking user engagement, you can remove it from the site. Existing data for the site is still available in your Google Analytics account.

You also can change the Google Analytics Measurement ID. On the Integrations page, click **Edit** for Google Analytics 4 and enter a different Measurement ID. Then publish the site.



Note: For testing purposes, you can publish the site from a sandbox or Developer Edition org. However, we recommend that you wait until you can deploy your site to production before you make it available to the public.

Appendix A: experience/cmsEditorApi Wire Adaptors and Functions

Use the `experience/cmsEditorApi` module to interact with the CMS content editor to get and set content. The wire adapters get the current state of the editor context and content, and a JavaScript function sets the content in the content editor.

getContext

Retrieves metadata that provides contextual information about the content item in the CMS content editor.

Syntax

```
import { LightningElement, wire } from 'lwc';
import { getContext } from experience/cmsEditorApi;
export default class Example extends LightningElement {
  @wire(getContext, {})
  context;
}
```

Returns

- `data`
 - `variantId`—ID of the content variant.
 - `contentSpaceId`—ID of the workspace that contains the content item.
 - `managedContentId`—ID of the managed content item.
 - `contentKey`—Unique identifier that tracks a content item through version updates and across environments. Assigned automatically when content is added to a CMS workspace.
 - `schema`—Information layout based on the content type.
- `error`

getContent

Retrieves the `title`, `urlName`, and `contentBody` of the content item based on the content type—image, news, document, or custom content.

Syntax

```
import { LightningElement, wire } from 'lwc';
import { getContent } from experience/cmsEditorApi;
export default class Example extends LightningElement {
  @wire(getContent, {})
  content;
}
```

Returns

- data
 - title—The title of the content item.
 - urlName—The content slug.
 - contentBody—The JSON representation of the content body.
- error

updateContent

Updates the `title` and `contentBody` of the content item in the editor. You can use `this.context.data.schema.schema.properties` to return the list of `contentBody` fields for the content type. Updates persisted only after the user clicks **Save** in the editor.

Syntax

In this example, `name`, `description`, and `excerpt` are fields of the `contentBody` for the content type.

```
import { LightningElement, wire } from 'lwc';
import { updateContent } from experience/cmsEditorApi;
class Example extends LightningElement{
    // change the content in the editor
    updateContent({
        "title": "new title",
        "contentBody": {
            "name": "New name",
            "description": "New description",
            "excerpt": "new-excerpt"
        }
    }).then(() => {
        //handle success
    })
}
```

Parameters

- title—(Optional) The new title of the content item in the editor.
- contentBody—(Optional) The JSON representation of the content body.

Returns

- A Promise object that resolves with the `updateDAF` response.
- error

Appendix B: Connect REST APIs and Connect in Apex Methods

Task	Supported in...
Get search indexes or trigger a search index for a site.	Connect REST API
Search a site.	Connect REST API Connect in Apex
Get a channel.	Connect REST API Connect in Apex
Get a collection of published contents for a channel.	Connect REST API Connect in Apex
Get a collection of published contents for an Experience Cloud site.	Connect REST API Connect in Apex
Get a piece of published content for a channel.	Connect REST API Connect in Apex
Get a piece of published content for a site.	Connect REST API Connect in Apex
Create managed content.	Connect REST API
Publish content.	Connect REST API
Unpublish content.	Connect REST API
Get or update a managed content variant.	Connect REST API
Get a managed content space.	Connect REST API Connect in Apex