Using Deploy Hooks

A deploy hook allows you to trigger a deployment using a unique URL.

The following are some of the example use cases for Deploy Hooks:

- Rebuilding your site to reflect changes in a Headless CMS
- Scheduling deployments with Cron Jobs

This document guides you through the process of creating, triggering, renaming, and deleting a deploy hook in Launch.

Prerequisites

1. Contentstack account
2. Access to Launch for your organization
3. A project already deployed in Launch

Create a Deploy Hook

Follow the steps given below to create a deploy hook in your project:

1. From the Launch landing page, click the project card to open your project.
2. In the Environments screen, hover over the environment where you created the project and then click the Environment Settings icon to configure the deploy hook. For this tutorial, we will be adding a deploy hook to the Default environment.

3. In Environments under Settings, click Deploy Hooks.

4. Click the + New Deploy Hook button.
5. Enter a name for your deploy hook.
   Note: The deploy hook name is for your reference. This name will display in your list of deploy hooks.

6. Click the Create Deploy Hook button.
   Launch provides you with a unique URL for this deploy hook.

Trigger a Deploy Hook

Follow the steps given below to trigger a deploy hook:

1. Click the Environments icon in the left panel.
2. Hover over the Default environment and click the Environment Settings icon to go to the Settings page.

3. In Environments under Settings, click Deploy Hooks.
4. Click the ellipses under Actions and then click Copy URL to copy the deploy hook URL.

5. Send a POST request to this URL to trigger the deploy hook you created.
   Example Request
   curl -X POST
   https://cs-domain.com/Launch-path/manage/deploy/deploy-hook-id

   Example Response
   { "message": "Deployment triggered successfully" }
6. Click the Environments icon in the left panel and then go to your environment.

You will see that it has triggered a deployment successfully.

If you used a git repository to import your project, then by default, the deploy hook uses the latest commit from the git branch for the deployment.

Parameter used in a Deploy Hook

Deploy hooks accept the following optional URL query parameter:

- commit: Parameter that specifies which git commit to deploy on triggering the deploy hook.

Here's an example where a deploy hook URL uses the commit parameter:

```
curl -X POST
https://cs-domain.com/Launch-path/manage/deploy/h83y7hreohuuererh7343?commit=rbg734
```

The commit can be the full SHA or just the first seven characters of your full SHA.

Note: When you import a project by uploading a file, you should not pass the commit parameter.

Rename a Deploy Hook

Follow the steps given below to rename a deploy hook:

1. Click the Environments icon in the left panel.
2. Hover over the Default environment and click the Environment Settings icon to go to the Settings page.

3. In Environments under Settings, click Deploy Hooks.

4. Click the ellipses under Actions and then click Edit to edit the deploy hook URL.

Delete a Deploy Hook

Follow the steps given below to delete a deploy hook:

1. Click the Environments icon in the left panel.
2. Hover over the Default environment and click the Environment Settings icon to go to the Settings page.

3. In Environments under Settings, click Deploy Hooks.

4. Click the ellipses under Actions and then click Delete to delete the deploy hook URL.

Redeploy Automatically when Content is Published on CMS

Launch allows you to set up a site to redeploy automatically when its content is modified in the CMS.

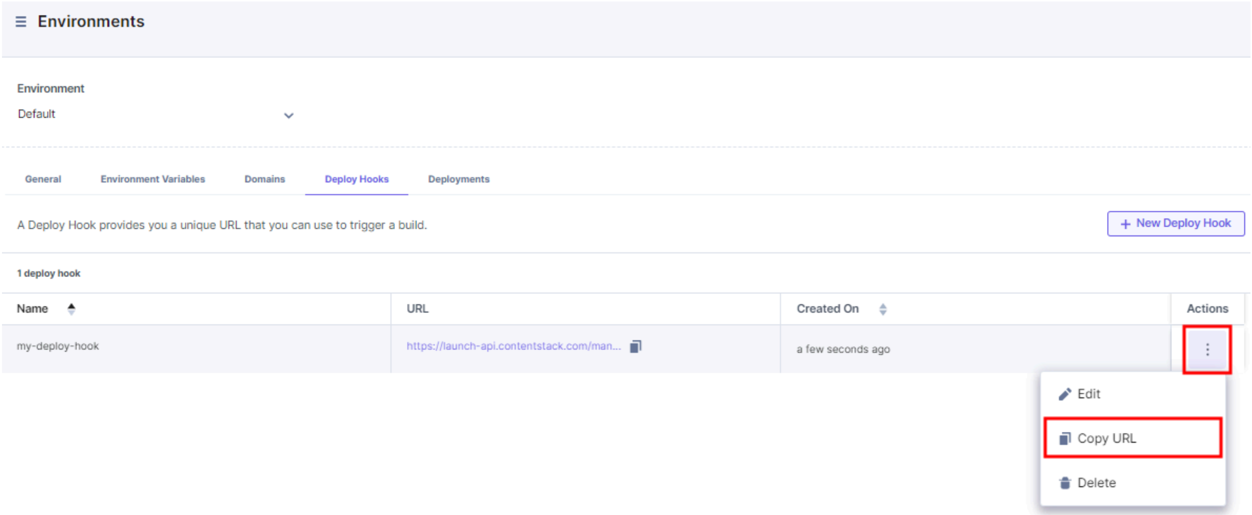
Prerequisites

1. Contentstack account
2. Access to Launch for your organization
3. A site on Launch that fetches data from the CMS

Steps for execution

Follow the steps to redeploy automatically when content is published.

1. Create a deploy hook in Launch.
2. Follow the steps to add the deploy hook as a Webhook to Contentstack CMS.



Environment
Default

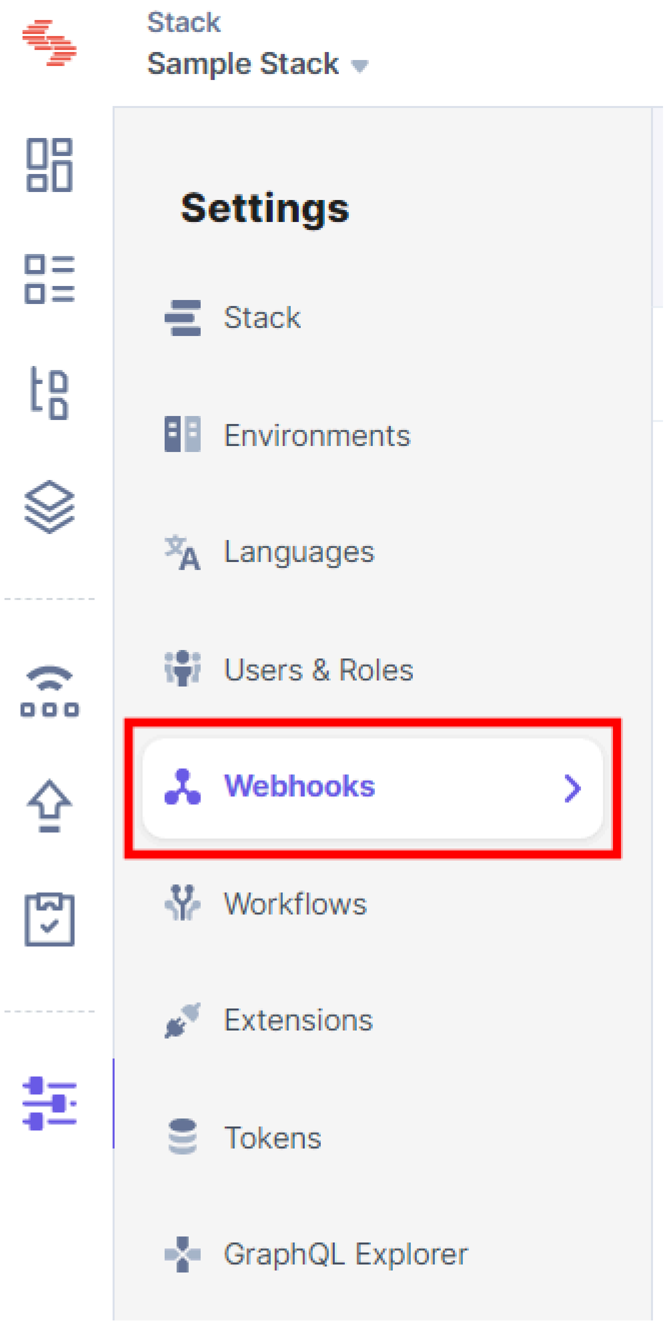
A Deploy Hook provides you a unique URL that you can use to trigger a build.

1 deploy hook

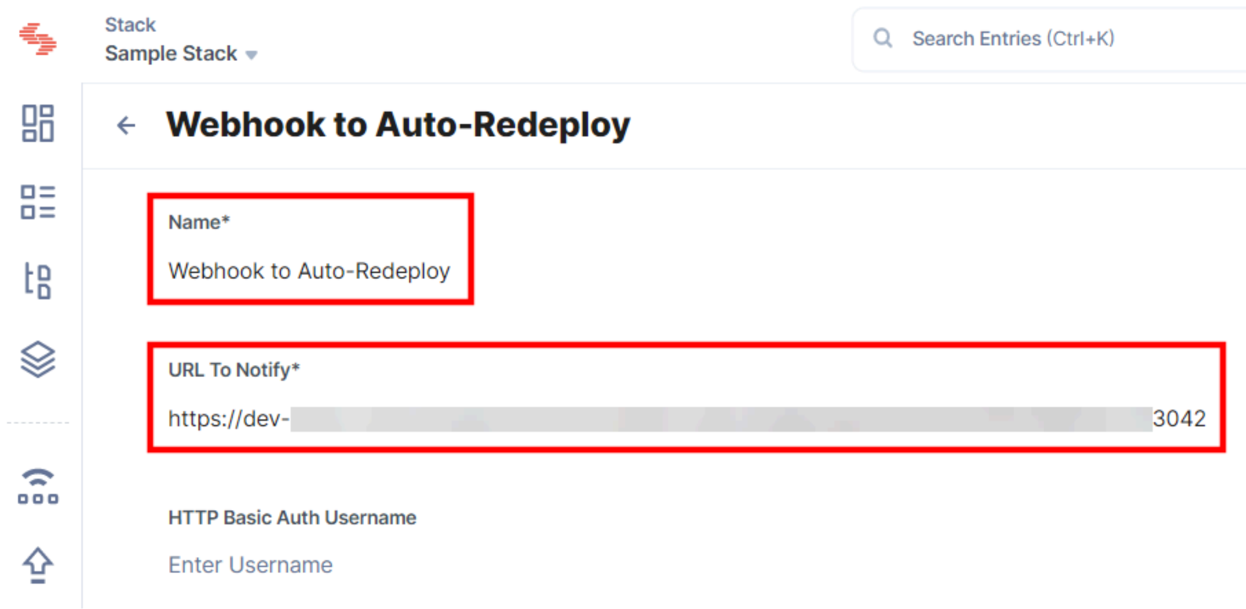
Name	URL	Created On	Actions
my-deploy-hook	https://launch-api.contentstack.com/man...	a few seconds ago	⋮

   1. Copy the deploy hook URL.

2. Go to your stack and then go to Settings > Webhooks.



3. Click the New Webhook button.
4. In the Name field, enter a suitable name for the Webhook.
5. In the URL To Notify field, paste the deploy hook URL.



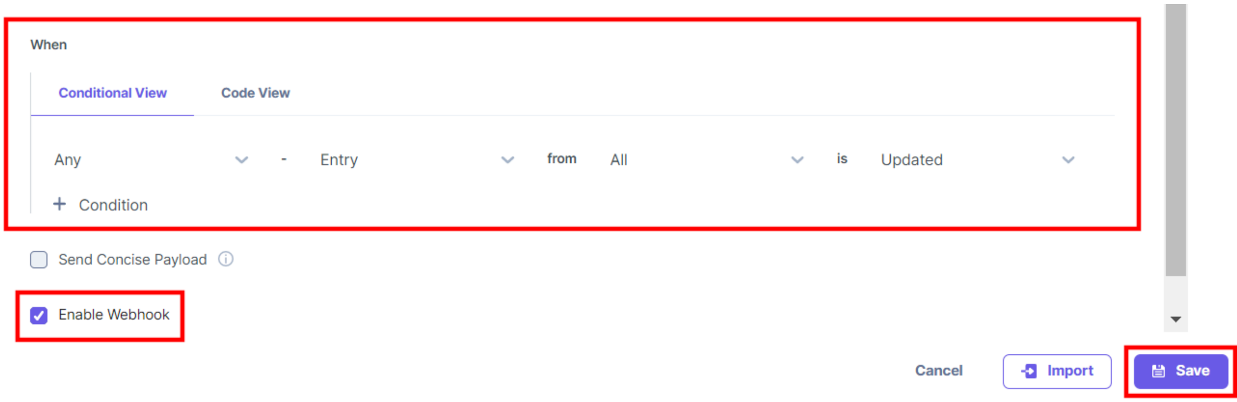
Webhook to Auto-Redeploy

Name*: Webhook to Auto-Redeploy

URL To Notify*: https://dev-[redacted]3042

HTTP Basic Auth Username: Enter Username

6. In the When field, enter the conditions for the deploy hook to trigger.
7. Mark the Enable Webhook checkbox.
8. Click the Save button.



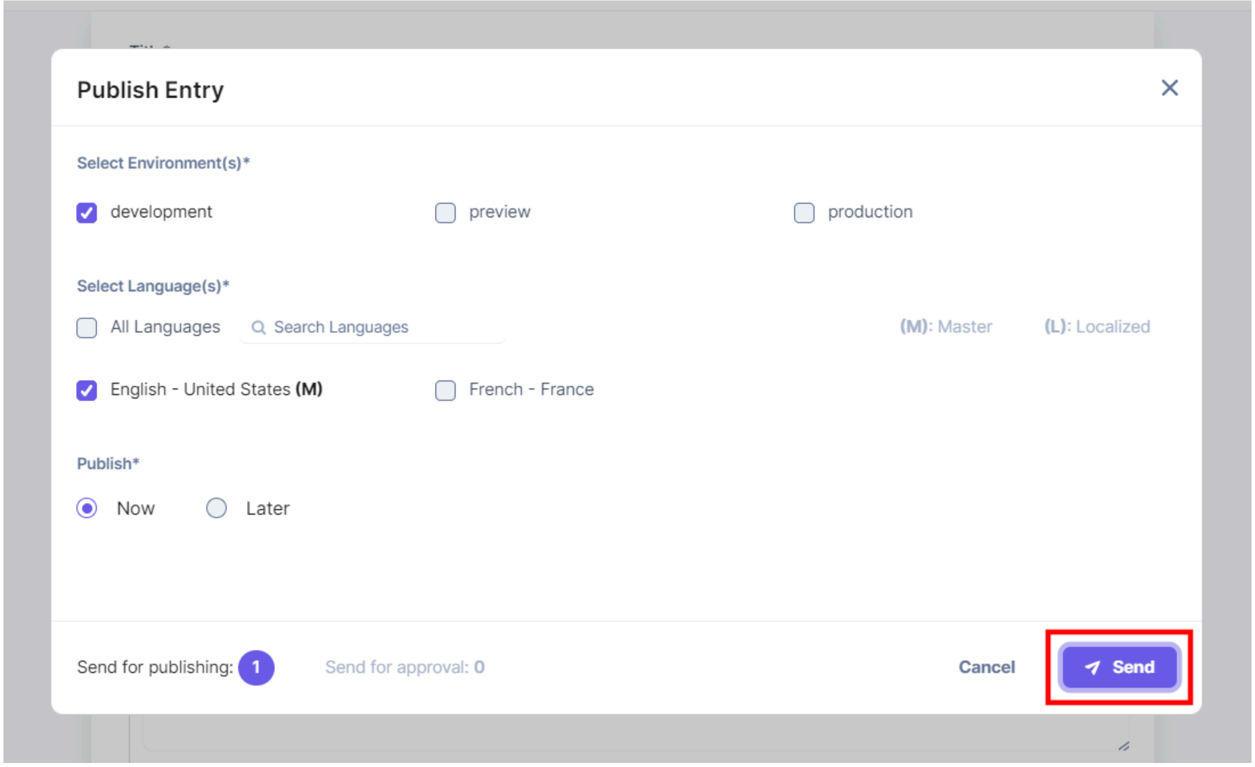
When: Any - Entry from All is Updated

□ Send Concise Payload

☑ Enable Webhook

3. You will see the conditions updated on the deploy hook page in Launch.

4. Go to the entry where you want to make the update.
5. Make the necessary updates, and click the Save button.
6. Click the Publish button, select Environment(s) and Language(s), and then click the Send button.



Publish Entry

Select Environment(s)*

☑ development ☐ preview ☐ production

Select Language(s)*

☐ All Languages

(M): Master (L): Localized

☑ English - United States **(M)** ☐ French - France

Publish*

● Now ○ Later

Send for publishing: 1

Send for approval: 0

On clicking Publish, the Webhook gets triggered and the project gets automatically redeployed in Launch.