

[Gatsby](#) is an open-source React-based framework that provides a hybrid static/server-side rendered model. Using Gatsby data fetching, you can render your content differently based on your application's use case and build APIs using Gatsby Functions.

Launch supports the following features of Gatsby out of the box:

- [Gatsby Functions](#)
- [DSG](#)
- [SSR](#)

The fastest way to get started with Launch is to clone and deploy a [sample Gatsby Starter project](#). Launch allows you to create a project by importing the website code from GitHub.

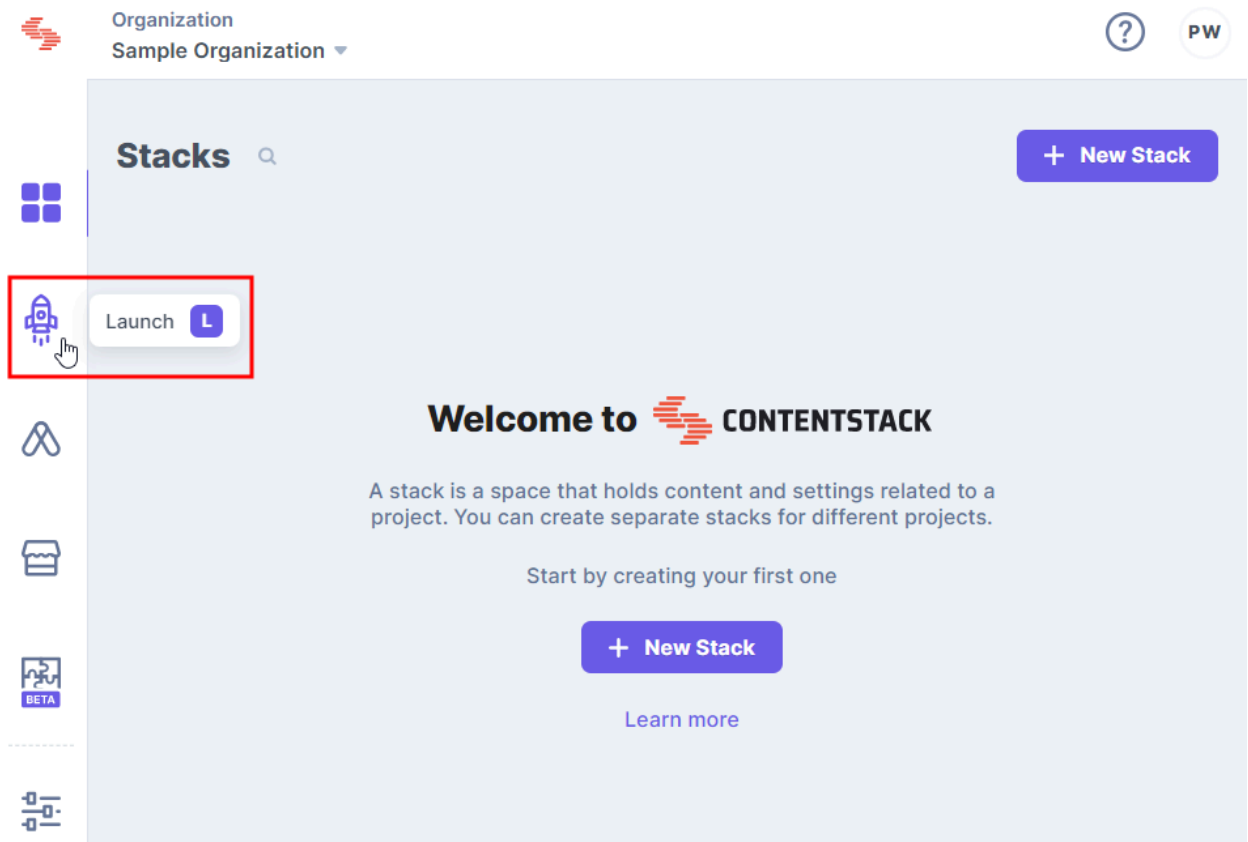
Prerequisites

1. [Contentstack account](#)
2. Access to Launch for your organization (Note: Only the Organization Admin/Owner has permissions to create Launch projects)
3. [GitHub account](#)

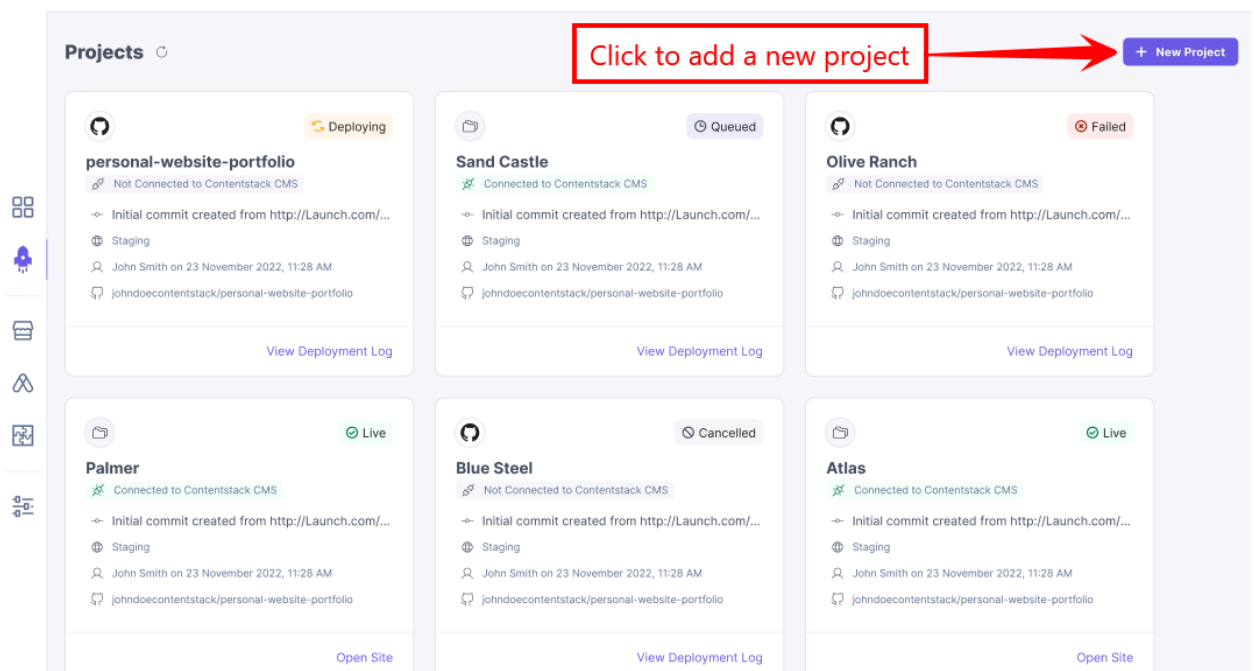
Steps for Execution

Follow the steps given below to deploy your first Launch project using GitHub:

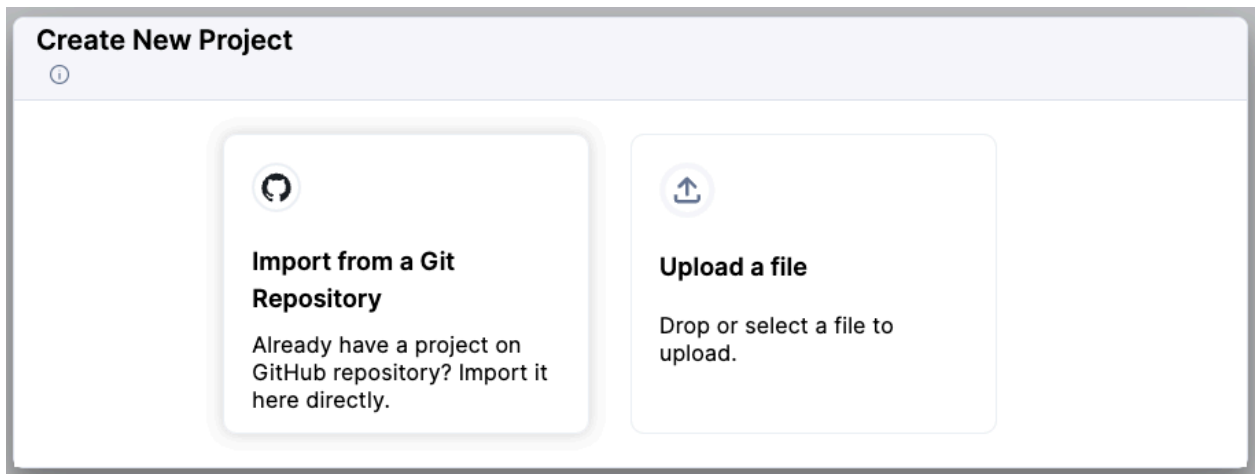
1. Log into Contentstack and click the Launch icon from the left navigation panel, as shown below:



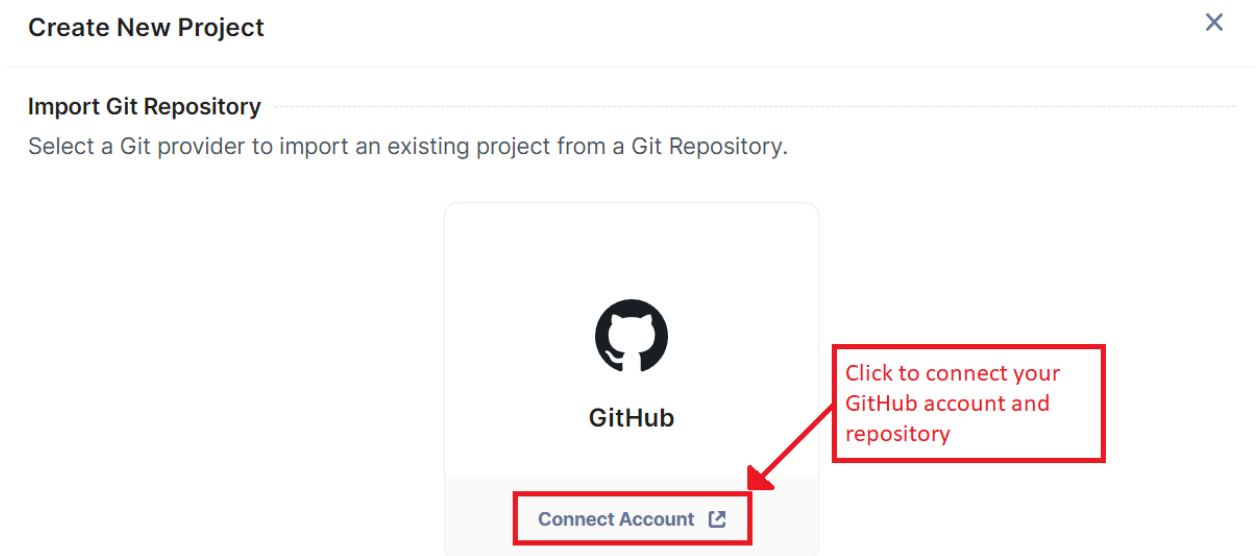
2. Click the + New Project button:



3. In the Create New Project modal, click Import from a Git Repository. (Note: You can also choose to [upload a .zip file to create your project.](#))



4. Click GitHub:



This will open a modal window to authenticate with your GitHub account.

5. Enter your credentials and log in:



Sign in to **GitHub**
to continue to **Contentstack Launch**

Username or email address

Password

[Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

6. In the Repository Access section, select All repositories.
Note: If you want to choose specific repositories, pick the Only select repositories option and choose the required repository.

Repository access

☒ **All repositories**

This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

☐ **Only select repositories**

Select at least one repository.
Also includes public repositories (read-only).


Save

Cancel

7. Click the Save button
Note: If you are accessing GitHub through Contentstack for the first time, you must authorize Contentstack by clicking the "Install & Authorize" button after selecting All repositories.



Install & Authorize Contentstack Launch

Install & Authorize on your personal account 

☒ **All repositories**

This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

☐ **Only select repositories**

Select at least one repository.
Also includes public repositories (read-only).

with these permissions:

- ✓ **Read** access to metadata and pull requests
- ✓ **Read** and **write** access to administration, code, commit statuses, repository hooks, and workflows

Click to install and authorize

Install & Authorize

Cancel

Next: you'll be redirected to <https://app.contentstack.com/connected-apps/callback>

Once you've connected your GitHub account, you'll be redirected back to Launch.

8. In the Create New Project modal, add the following details:

Repository (Mandatory): Select your Git repository.

Git Branch (Mandatory): By default, master or main is selected as the branch. You can choose another branch from the dropdown.

Project Name (Mandatory): This is auto-populated when you select the repository, but you can edit it as needed.

Environment Name (Mandatory): The environment name is automatically populated as Default, but can be changed. Note that this is specific to Launch and not directly related to any environments in your stack.

Build and Output Settings (Mandatory): The fields in this section get auto-populated based on the detected framework. (If no framework is detected, you can specify your own Build Command and Output Directory.)

Environment Variables (Optional): Enter the key and value of one or more environment variables. You can also use the "Connect and Import Variables" option to automatically import API-related variables from a specific stack and delivery token.

← Create New Project

×

Repository

RohitKini/LaunchGatsbyDemo

▼

Git Branch*

master

▼

Detected framework: Gatsby

Project Name*

LaunchGatsbyDemo

Environment Name*

Default

Build and Output Settings

Framework Preset*

Gatsby

▼

Build Command

npm run build

Output Directory*

./public

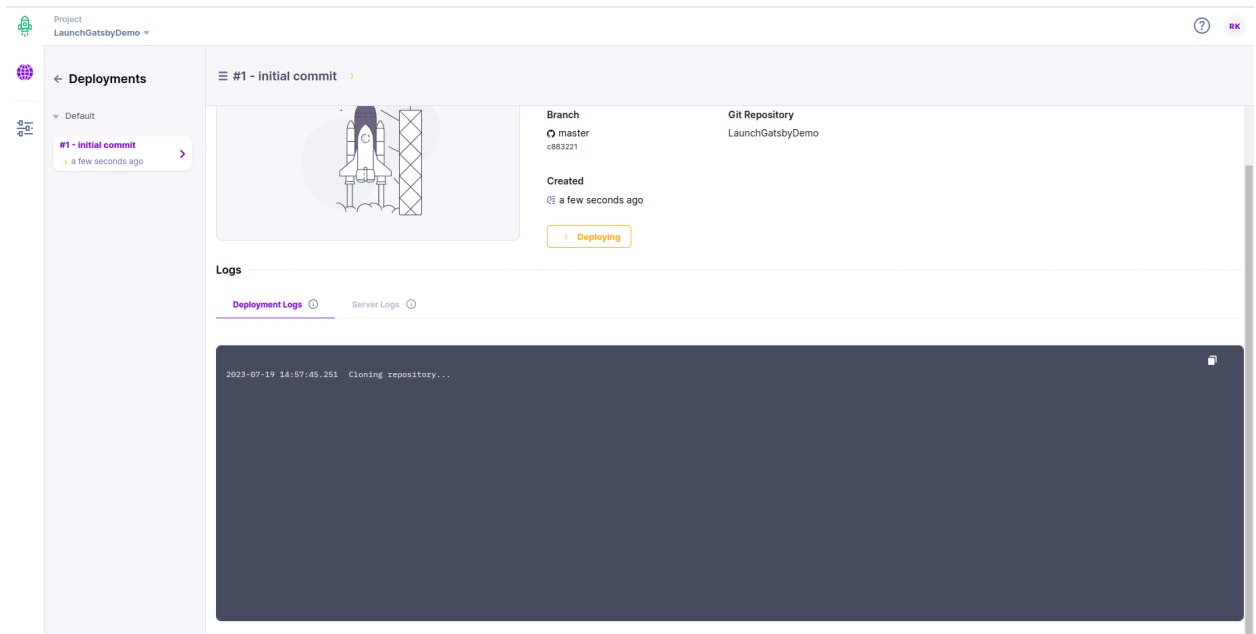
Cancel

← Back

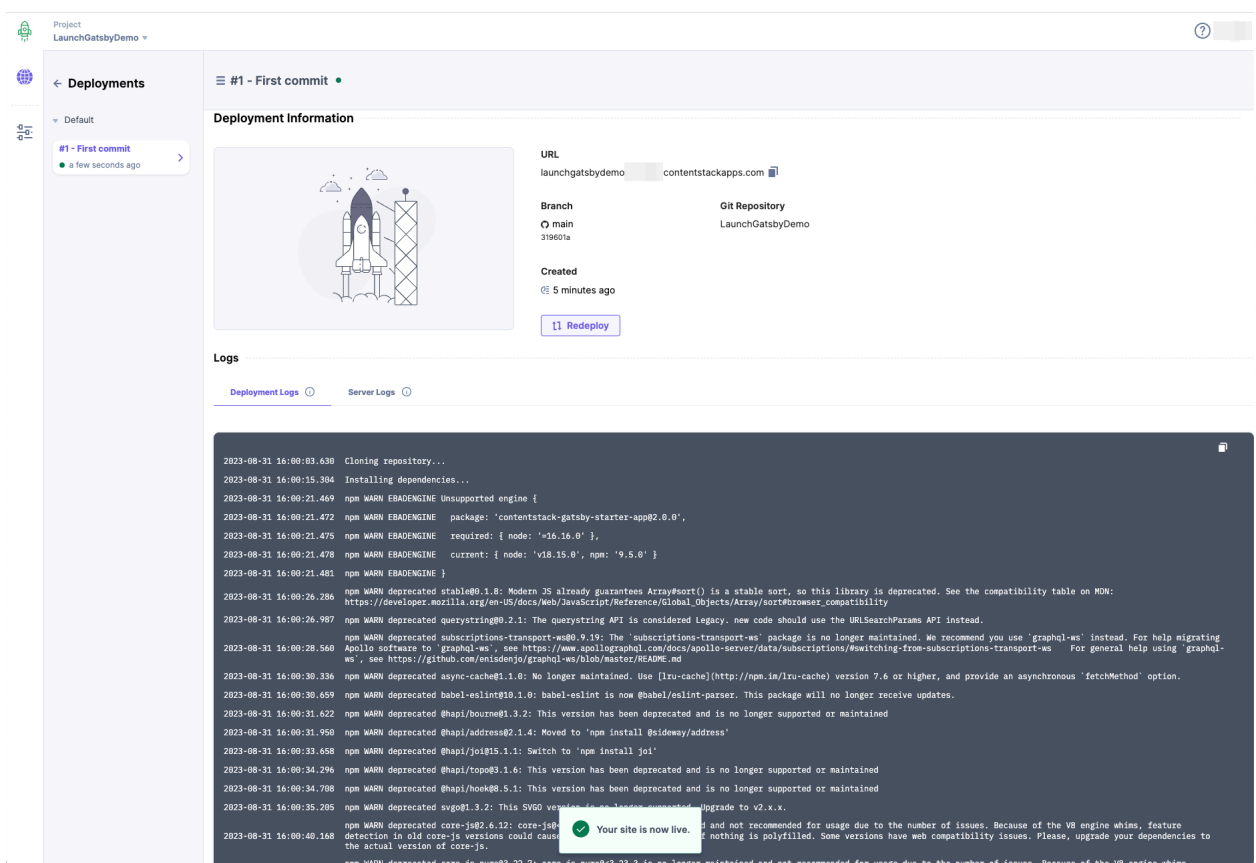
🚀 Deploy

Note: Click Back to revert the changes and re-enter details.

9. Once ready, click Deploy to save and deploy the project.
You'll see the project's status update in real time, from Queued to Deploying to Deployed. Note: depending on the size and complexity of your specific application, it may take several minutes to build and deploy your site.



Upon successful deployment, you will see the following screen:



You can preview the deployed website by copying the URL from the URL

section under Deployment Information and opening it in a browser.

If you want to deploy the changes from any other commits, you can click the Redeploy button and choose the required commit.

The screenshot shows the Vercel deployment interface for a project named 'LaunchGatsbyDemo'. On the left, a sidebar contains a 'Deployments' section with a list of deployments, including '#1 - initial commit' which is 5 days old. The main area is titled '#1 - initial commit' and contains a 'Deployment Information' section. This section includes a preview of the deployed application, the URL 'launchgatsbydemo.contentstackapps.com', the branch 'master' (commit c883221), the Git Repository 'LaunchGatsbyDemo', and the creation time '5 days ago'. A 'Redeploy' button is visible. Below the information is a 'Logs' section with tabs for 'Deployment Logs' and 'Server Logs'. The 'Deployment Logs' tab is active, showing a detailed log of the deployment process, including cloning the repository, installing dependencies, and resolving peer dependencies.

Project
LaunchGatsbyDemo

Deployments

Default

#1 - initial commit
5 days ago

Deployment Information

URL
launchgatsbydemo.contentstackapps.com

Branch
master
c883221

Git Repository
LaunchGatsbyDemo

Created
5 days ago

Redeploy

Logs

Deployment Logs Server Logs

```
2023-07-19 14:57:45.251 Cloning repository...
2023-07-19 14:57:53.085 Installing dependencies...
2023-07-19 14:58:00.081 npm WARN ERESOLVE overriding peer dependency
2023-07-19 14:58:00.085 npm WARN While resolving: react-server-dom-webpack@0.0.0-experimental-c8b778b7f-20220825
2023-07-19 14:58:00.088 npm WARN Found: react@18.2.0
2023-07-19 14:58:00.091 npm WARN node_modules/react
2023-07-19 14:58:00.094 npm WARN   react@"18.2.0" from the root project
2023-07-19 14:58:00.099 npm WARN   7 more (@gatsbyjs/reach-router, gatsby, gatsby-link, ...)
2023-07-19 14:58:00.102 npm WARN
2023-07-19 14:58:00.104 npm WARN Could not resolve dependency:
2023-07-19 14:58:00.107 npm WARN peer react@"9.0.0-experimental-c8b778b7f-20220825" from react-server-dom-webpack@0.0.0-experimental-c8b778b7f-20220825
2023-07-19 14:58:00.110 npm WARN node_modules/react-server-dom-webpack
2023-07-19 14:58:00.111 npm WARN   react-server-dom-webpack@0.0.0-experimental-c8b778b7f-20220825 from gatsby@5.7.0
```

Note: The log details of the current deployment can be found under the Logs section.