

Self-hosted Backstage vs Managed Backstage

A guide for making the best choice for your team

Presented by



Introduction

Backstage is one of the fastest-growing CNCF incubating projects, with 1000+ public adopters, including large-scale enterprises like HP and many scale-ups like SumUp, Contentful, and Snyk.

Backstage is the most flexible and powerful option for setting up an Internal Developer Portal, thanks to its OSS nature and plugin-based architecture. However, it relies on a stack not always familiar to the Platform and DevOps teams setting it up (ie. Node, React, TypeScript). Backstage also requires some manual work for maintenance and needs you to keep up to date with internal APIs due to its rapid evolution.

Gartner and others have reported that setting up and maintaining Backstage yourself can be challenging, but the value of doing so has overwhelming benefits for many companies. However, you do not necessarily have to set up Backstage from scratch and self-host it. Instead, you can adopt a managed Backstage instance via [Roadie.io](https://roadie.io), dramatically reducing your setup and maintenance costs.

In this whitepaper, we'll overview considerations to help you decide if self-hosting Backstage is your best option or if managed Backstage can free your time for other tasks. We'll cover aspects regarding:

1. [Initial setup and deployment](#)
2. [Authentication and Security](#)
3. [Upgrades and maintenance](#)
4. [Availability and support](#)
5. [Customizability](#)
6. [Working with Plugins](#)

Initial setup and deployment

After shopping around for Internal Developer Portals (IDPs), it's time to try out Backstage. Naturally, you'll first install it locally and play around with it, but it won't be long until you have to set up a production-grade deployment for your IDP. Here you have two options: figure out the deployment yourself, or request a hosted Backstage trial from [Roadie.io](https://roadie.io).

Below are some considerations for each of the alternatives from what we have observed from a technical standpoint and after talking to dozens of Backstage adopters.

Managed Backstage (Roadie.io)

2-4 hours, 1 engineer

Get a production-ready Backstage instance from Roadie. All instances are hosted on AWS.

Use Roadie's Admin UI to set up at least one integration, for example, GitHub, through a few clicks. To simplify this process, we offer a GitHub App that can also check your catalog-info files before they're merged to prevent corrupt entries in your Catalog.

Our team of Backstage experts is ready to support you if you run into any trouble. Roadie is the second largest contributor to the OSS project.

Self-hosted Backstage

2-3 weeks, 2 engineers

Work with TypeScript, JSON, and YAML files to configure your Backstage instance, using Node for the backend and React for the frontend.

Implement a secrets management strategy that lets you inject secrets into the container that will be running Backstage. You'll use this for most integrations.

Implement at least one integration like GitHub to try out your setup. To do that, you'll need to install the corresponding packages through npm and set them up using React, Node, and JSON files. Then create a GitHub app or token with correct permissions (and keep that up to date). Hook the integration into your secrets management and auth solution.

Set up a deployment pipeline. Backstage recommends deploying as you usually would with a similar monorepo application, whether that is using Kubernetes, Docker, Heroku, or something else.

Authentication and Security

Your Internal Developer Portal will become a central hub of every developer in your organization, integrating third-party services (GitHub, ArgoCD, PagerDuty, etc) and your internal clusters. Furthermore, through your IDP, users can request infrastructure thanks to its self-service or golden paths capabilities. Therefore, you need to make sure that your Backstage instance is secure.

Managed Backstage (Roadie.io)

1-2 hours, 1 engineer

Enable Single Sign On (SSO) through Okta, GitHub, and other providers with a few clicks in Roadie's Admin UI. Both frontend and backend authentication is set up out of the box.

Roadie uses ephemeral environments on each Scaffolder template run, which minimizes the scope and permissions of each template.

Roadie offers a battle-tested Catalog API with authenticated access so you can do more with Backstage.

Roadie applies security patches from Backstage as soon as possible.

Roadie is SOC2 Type 2 compliant under rigorous industry standards continuously assessed by a third party.

Roadie gets regular pen-testing assessments through external security firms.

Self-hosted Backstage

2-3 weeks, 2 engineers

Install an existing authentication integration through npm, and hook it up into your app through Node and React configs to make sure the right sign-in page shows up, and that authorization headers are working correctly across your instance.

Implement an authorization strategy for the frontend and backend. Backstage comes with a permissions framework that you have to learn about and leverage.

To minimize risks, you may want to isolate scaffolder tasks. Backstage doesn't come with this out of the box so you must implement a solution on your own. Even if your instance runs in a trusted environment, isolating the Scaffolder tasks is important not to drown the backend.

Stay tuned to vulnerability patches from Backstage and apply the upgrades as soon as possible.

Find a way to secure the API access to you can use Backstage in other parts of your DX

Upgrades and maintenance

The Backstage team merges an average of 300 PRs per month from dozens of contributors, resulting in minor version releases on the second week of every month. As you can imagine from this figure, the framework gets a constant stream of new features and bug fixes. The upgrade process for Backstage right now involves some manual work, and the complexity of applying each release varies significantly.

Managed Backstage (Roadie.io)

0 hours

Roadie manages all upgrades for you, which means you get all the new features automatically. Upgrades are all different, and even after having managed Backstage full-time for a while, it can still take a few days—sometimes, weeks—to bring all the new features to your instance after having tested them.

Roadie is ready for you to scale, which includes a proprietary mitigation to the GitHub rate limits issue, a common pain point for Backstage adopters as they onboard more components into their Catalog.

Self-hosted Backstage

1-3 weeks per month, 1 engineer

Upgrade the dozens of core plugins, as well as associated dependencies. Backstage provides a version bump utility, but keep in mind that you'll likely still have significant manual work left to make the upgrade work.

You need to stay tuned to the Backstage project changes to be able to keep up with the upgrades. Backstage interacts with many external services, which means slight changes in an API can disrupt your IDP because you missed changing a setting in a vendor.

Usually, new features are provided as new APIs and new UI components, but you still need to wire the ones you're interested in into your instance using Node and React.

As you onboard more components and developers into Backstage, you'll start running into problems with rate limits, especially from source control systems like GitHub. Backstage offers OSS libraries you can leverage to implement a webhooks-based solution to mitigate this issue, but you need to integrate them into your IDP yourself.

Availability and support

Ensuring that your Internal Developer Portal is available will grow in importance as Backstage becomes central to your organization's development workflow. Otherwise, developers might be frustrated when they can't access documentation, request infrastructure, or discover resources because their IDP is down. Additionally, developers may need help to be productive with Backstage so you need to have a way to assist them.

Managed Backstage (Roadie.io)

0 hours

Roadie runs Backstage on a modern, highly-available, Cloud. That's why Roadie provides an [average uptime](#) of 99.997%.

Roadie has an automated backup process for customer data.

Roadie has on-call rotations in case an incident occurs, even at odd times, to get your Backstage instance back up as soon as possible.

Roadie offers a dedicated Slack channel on the Growth plan so your developers can ask questions to Backstage experts.

Roadie offers an Amplitude analytics dashboard that shows how your Backstage instance is being used.

Self-hosted Backstage

4-8 hours per month, 1 engineer

You need to implement a monitoring and alerting solution for your IDP, and a strategy to keep your Backstage instance up and recover it when it goes down.

Recovering a Backstage instance fully after an incident requires specialized knowledge of the platform, which relatively few people in your team are exposed to. Figuring out exactly what is causing issues in your instance can be challenging, especially after an upgrade as you need to be aware of all the changes and how they impact the integrated services APIs.

Once you set up your Backstage IDP, you'll want to share it with every dev team in your organization. They will most likely need support from you to onboard their components into the Catalog or to write Scaffolder templates.

To understand how your IDP is being used and improve it, you need to leverage the Analytics API provided by Backstage.

Customizability

An Internal Developer Portal must be adapted to the need of your developers, and that includes the visual aspect too.

Managed Backstage (Roadie.io)

<1 hour

Set up a color scheme for Backstage's theme and add your company logo through a UI

Customize your Backstage's layouts through a drag-and-drop UI.

Self-hosted Backstage

Depends on your ambition

An amazing aspect of Backstage is that you can customize it entirely. Some teams, like HP, have even replaced the entire UI and rebuilt it using their design system. This comes at a high cost—as you'll miss out on the OSS contributions to the UI—but it can make a more cohesive UX across all your internal tools.

For a minimum theme, you can change a few CSS variables and set your company logo using React.

To customize layouts, you'll also rely on React and styles to put blocks where you want them to be.

Working with Plugins

Backstage's architecture is based on plugins to let you build an Internal Developer Portal with everything your teams need. There are dozens of Backstage plugins available in the OSS community, typically integrating with services that your developers already use, such as PagerDuty or Snyk. The idea is to make a Single Pane of Glass that reduces context-switching for your developers.

Managed Backstage (Roadie.io)

5-10 minutes per plugin

Roadie offers [47 vetted plugins](#) out of the box, which you can set up through the Admin UI, no coding required.

Roadie keeps all the plugins up to date with new releases.

You can add custom plugins to Roadie by sharing your plugin code using a secure pipeline. Once you do, it'll become available in your instance. However, the custom plugins in Roadie have to be designed to rely on existing APIs or third-party endpoints through the Backstage proxy.

Self-hosted Backstage

1-2 hours per plugin + upgrades

Use npm or yarn to install the plugin packages, usually 2 or 3 (frontend, backend, common) into your instance.

Add the configuration details to the instance config file, add the backend plugin to the express app, and add the frontend plugin to the UI using React.

Re-deploy your instance.

To add a custom plugin to your instance, you can deploy it as part of your monorepo app. You have to wire the custom plugin into your instance just as you would do with an OSS plugin.

Try Roadie Backstage

If you want to find out more about Roadie's managed Backstage instances, [book a demo](#) today and figure out if it's the best option for you.

