Disaster recovery for GitHub Codespaces

In this article

This article describes guidance for a disaster recovery scenario, when a whole region experiences an outage due to major natural disaster or widespread service interruption.

We work hard to make sure that GitHub Codespaces is always available to you. However, forces beyond our control sometimes impact the service in ways that can cause unplanned service disruptions.

Although disaster recovery scenarios are rare occurrences, we recommend that you prepare for the possibility that there is an outage of an entire region. If an entire region experiences a service disruption, the locally redundant copies of your data would be temporarily unavailable.

The following guidance provides options on how to handle service disruption to the entire region where your codespace is deployed.

> **Note:** You can reduce the potential impact of service-wide outages by pushing to remote repositories frequently.

Option 1: Create a new codespace in another region

In the case of a regional outage, we suggest you recreate your codespace in an unaffected region to continue working. This new codespace will have all of the changes as of your last push to GitHub. For information on manually setting another region, see "Setting your default region for GitHub Codespaces."

You can optimize recovery time by configuring a `devcontainer.json` in the project's repository, which allows you to define the tools, runtimes, frameworks, editor settings, extensions, and other configuration necessary to restore the development environment automatically. For more information, see "Introduction to dev containers."

Option 2: Wait for recovery

In this case, no action on your part is required. Know that we are working diligently to restore service availability.

You can check the current service status on the Status Dashboard.

Option 3: Clone the repository locally or edit in the browser

While GitHub Codespaces provides the benefit of a pre-configured developer environmnent, your source code should always be accessible through the repository hosted on GitHub.com. In the event of a GitHub Codespaces outage, you can still clone the repository locally or edit files in the GitHub browser editor. For more information, see "Editing files."

While this option does not configure a development environment for you, it will allow you to make changes to your source code as needed while you wait for the service disruption to resolve.

Option 4: Use the Dev Containers extension and Docker for a local containerized environment

If your repository has a `devcontainer.json`, consider using the Dev Containers extension in Visual Studio Code to build and attach to a local development container for your repository. The setup time for this option will vary depending on your local specifications and the complexity of your dev container setup. For more information, see "Developing inside a container" in the VS Code documentation.

> **Note:** Be sure your local setup meets the minimum requirements before attempting this option.