

[Home](#) / [Guides](#) / [Ruby](#) / Configure CI/CD

# Configure CI/CD for your Ruby on Rails application

Table of contents

[Prerequisites](#)

[Overview](#)

[Step one: Create the repository](#)

[Step two: Set up the workflow](#)



[Step three: Run the workflow](#)

[Summary](#)

[Next steps](#)

---

## Prerequisites

Complete all the previous sections of this guide, starting with [Containerize a Ruby on Rails application](#). You must have a [GitHub](#)  account and a [Docker](#)  account to complete this section.

## Overview


In this section, you'll learn how to set up and use GitHub Actions to build and test your Docker image as well as push it to Docker Hub. You will complete the following steps:

1. Create a new repository on GitHub.
2. Define the GitHub Actions workflow.
3. Run the workflow.

[Give feedback](#)

# Step one: Create the repository

Create a GitHub repository, configure the Docker Hub credentials, and push your source code.

1. [Create a new repository](#)  on GitHub.
2. Open the repository **Settings**, and go to **Secrets and variables > Actions**.
3. Create a new **Repository variable** named `DOCKER_USERNAME` and your Docker ID as value.
4. Create a new [Personal Access Token \(PAT\)](#) for Docker Hub. You can name this token `docker-tutorial`. Make sure access permissions include Read and Write.
5. Add the PAT as a **Repository secret** in your GitHub repository, with the name `DOCKERHUB_TOKEN`.
6. In your local repository on your machine, run the following command to change the origin to the repository you just created. Make sure you change `your-username` to your GitHub username and `your-repository` to the name of the repository you created.

```
$ git remote set-url origin https://github.com/your-username/your-reposit
```

7. Run the following commands to stage, commit, and push your local repository to GitHub.

```
$ git add -A
$ git commit -m "my commit"
$ git push -u origin main
```

# Step two: Set up the workflow

Set up your GitHub Actions workflow for building, testing, and pushing the image to Docker Hub.

1. Go to your repository on GitHub and then select the **Actions** tab.
2. Select **set up a workflow yourself**.

This takes you to a page for creating a new GitHub actions workflow file in your repository, under `.github/workflows/main.yml` by default.

3. In the editor window, copy and paste the following YAML configuration.

```
name: ci

on:
  push:
    branches:
      - main

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Login to Docker Hub
        uses: docker/login-action@v3
        with:
          username: ${ vars.DOCKER_USERNAME }
          password: ${ secrets.DOCKERHUB_TOKEN }

      - name: Set up Docker Buildx
        uses: docker/setup-buildx-action@v3


      - name: Build and push
        uses: docker/build-push-action@v6
        with:
          push: true
          tags: ${ vars.DOCKER_USERNAME }/${ github.event.repository.name }
```

For more information about the YAML syntax for `docker/build-push-action`, refer to the [GitHub Action README](#).

## Step three: Run the workflow

Save the workflow file and run the job.


1. Select **Commit changes...** and push the changes to the `main` branch.  
After pushing the commit, the workflow starts automatically.
2. Go to the **Actions** tab. It displays the workflow.  
Selecting the workflow shows you the breakdown of all the steps.

3. When the workflow is complete, go to your [repositories on Docker Hub](#) .  
If you see the new repository in that list, it means the GitHub Actions successfully pushed the image to Docker Hub.

## Summary

In this section, you learned how to set up a GitHub Actions workflow for your Ruby on Rails application.

Related information:

- [Introduction to GitHub Actions](#)
- [Docker Build GitHub Actions](#)
- [Workflow syntax for GitHub Actions](#) 

## Next steps

Next, learn how you can locally test and debug your workloads on Kubernetes before deploying.

[Test your Ruby on Rails deployment »](#)

Product offerings

Pricing

About us

Support

Contribute

---

Copyright © 2013-2025 Docker Inc. All rights reserved.



[Cookies Settings](#)

Theme: [Light](#)