GitHub Actions

GitHub Actions

Anonymous · 09/09/2026

GitHub Actions

Derafu Foundation includes pre-configured GitHub Actions workflows for continuous integration (CI) and continuous deployment (CD). This document explains how these workflows work and how to customize them.

Overview

The `.github/workflows/` directory contains two main workflow files:

- `ci.yml`: Continuous Integration workflow.
- `cd.yml`: Continuous Deployment workflow.

Continuous Integration (CI)

The CI workflow runs automatically on each push and pull request to validate code quality and ensure tests pass.

What It Does

The CI workflow:

1. Sets up multiple PHP versions for testing.
2. Installs dependencies using Composer.
3. Validates Composer configuration.
4. Runs PHP CS Fixer to check code style.
5. Runs PHPStan for static analysis.
6. Runs PHPUnit tests.
7. Generates test coverage reports.

Customization

To customize the CI workflow, edit the `.github/workflows/ci.yml` file. Common customizations include:

- Changing PHP versions to test against.
- Adding or removing validation steps.
- Modifying test coverage thresholds.
- Changing notification settings.

Example of adding a new PHP version to test against:

```
# In .github/workflows/ci.yml
jobs:
  tests:
    strategy:
      matrix:
        php-version: ['8.3', '8.4']  # Add or remove versions as needed.
```

Continuous Deployment (CD)

The CD workflow automatically builds and deploys your project documentation to GitHub Pages when changes are pushed to the main branch.

Note: Deployment is disabled by default, just remove the `false &&` in the file to enable it.

What It Does

The CD workflow:

1. Checks out the repository.
2. Sets up Node.js.
3. Installs frontend dependencies.
4. Builds assets using Vite.
5. Sets up PHP.
6. Installs Composer dependencies.
7. Generates a static website from your project documentation.
8. Deploys the static website to GitHub Pages.

GitHub Pages Setup

To use the CD workflow, you need to:

1. Enable GitHub Pages in your repository settings.
2. Set the source branch to `gh-pages` and the directory to `/` (root).
3. Ensure the workflow has permission to write to the repository.

Customization

To customize the CD workflow, edit the `.github/workflows/cd.yml` file. Common customizations

include:

- Changing the deployment branch.
- Adding custom build steps.
- Configuring environment variables.
- Adding additional deployment targets.

Example of changing the deployment branch:

```
# In .github/workflows/cd.yml
- name: Deploy to GitHub Pages
  uses: peaceiris/actions-gh-pages@v3
  with:
    github_token: ${{ secrets.GITHUB_TOKEN }}
    publish_dir: ./public
    publish_branch: documentation  # Change from gh-pages to another branch name.
```

Workflow Badges

You can add workflow status badges to your README.md to show the current status of your workflows:

```
![GitHub last commit](https://img.shields.io/github/last-commit/derafu/project/main)
![CI
Workflow](https://github.com/derafu/project/actions/workflows/ci.yml/badge.svg?branch=
main&event=push)
![GitHub code size in
bytes](https://img.shields.io/github/languages/code-size/derafu/project)
![GitHub Issues](https://img.shields.io/github/issues-raw/derafu/project)
![Total Downloads](https://poser.pugx.org/derafu/project/downloads)
![Monthly Downloads](https://poser.pugx.org/derafu/project/d/monthly)
```

Replace `derafu/project` with your actual GitHub username and repository name.

Best Practices

1. **Keep workflows fast**: Optimize workflows to complete quickly.
2. **Use caching**: Cache dependencies to speed up builds.
3. **Secure secrets**: Use GitHub Secrets for sensitive information.
4. **Test workflow changes**: Test workflow changes on a feature branch before merging to main.
5. **Monitor workflow runs**: Regularly check workflow runs for issues.

Troubleshooting

If a workflow fails:

1. Check the workflow run logs in the GitHub Actions tab.
2. Verify that all dependencies are correctly installed.
3. Ensure environment variables are correctly set.
4. Check if tests are failing locally as well.
5. Look for GitHub Actions service status issues.

If needed, you can run the workflow manually from the Actions tab by selecting the workflow and clicking “Run workflow”.

Last updated on 09/09/2026